ТЕХНИЧЕСКИ УНИВЕРСИТЕТ – СОФИЯ
Факултет Компютърни системи и технологии
Катедра Информационни технологии в индустрията

Маг. инж. Георги Руменов Цочев

ИЗСЛЕДВАНЕ НА МЕТОДИ ОТ ИЗКУСТВЕНИЯ ИНТЕЛЕКТ ЗА ПРИЛОЖЕНИЕ В КОМПЮТЪРНАТА МРЕЖОВА СИГУРНОСТ

А В Т О Р Е Ф Е Р А Т

на дисертация за придобиване на образователна и научна степен
"ДОКТОР"

Област: 5. Технически науки

Професионално направление: 5.3 Комуникационна и компютърна техника

Научна специалност: Системи с изкуствен интелект

Научени ръководители: Проф. д-р Румен Трифонов
Доц. д-р Георги Попов

СОФИЯ, 2018 г.

Дисертационният труд е обсъден и насочен за защита от Катедрения съвет на катедра „Информационни технологии в индустрията“ към Факултет Компютърни системи и технологии на ТУ-София на редовно заседание, проведено на 18.06.2018 г.

Публичната защита на дисертационния труд ще се състои на 11.10.2018 г. от 13,00 часа в Конферентната зала на БИЦ на Технически университет – София на открито заседание на научното жури, определено със заповед № ОЖ-232 / 04.07.2018 г. на Ректора на ТУ-София в състав:

1. проф д-р Огнян Наков – председател
2. проф. д-р Румен Трифонов – научен секретар
3. акад. д.т.н. Васил Сгурев
4. проф. д-р Ангел Смрикаров
5. доц. д-р Пламен Вачков

Рецензенти:

1. акад. д.т.н. Васил Сгурев
2. доц. д-р Пламен Вачков

Материалите по защитата са на разположение на интересуващите се в канцеларията на Факултет Компютърни системи и технологии на ТУ-София, блок №1, кабинет № 1443а.

Дисертантът е задочен докторант към катедра „Информационни технологии в индустрията“ на факултет Компютърни системи и технологии. Изследванията по дисертационната разработка са направени от автора, като някои от тях са подкрепени от научноизследователски проекти.

Автор: маг. инж. Георги Цочев
Заглавие: Изследване на методи от изкуствения интелект за приложение в компютърната мрежова сигурност
Тираж: 30 броя
Отпечатано в ИПК на Технически университет – София

I. ОБЩА ХАРАКТЕРИСТИКА НА ДИСЕРТАЦИОННИЯ ТРУД

Актуалност на проблема

В днешно време Интернет е навсякъде и разчитаме изключително много на тази услуга, не зависимо дали я използваме за работа, забавление или само за достъп до определена информация. Почти няма бизнес, които да не разчита на интернет и отдалечени комуникации.

Много важно за една организация е да изработи механизми за сигурност, които да предотвратят неоторизиран достъп до системните ресурси и поверителни данни на компанията. Има много начини да се защити мрежова инфраструктура на едно предприятие, в това число се включват и системите за откриване и превенция на прониквания. През последните две десетилетия развитието на мрежите от компютри е поле за иновационни подобрения. На пазара има много разновидности от тип „Next Generation“, които предоставят сравнително добър набор от инструменти (системи) за взаимодействие и предотвратяване от мрежови атаки.

Физическите устройства като сензори и датчици не са достатъчни за наблюдение, анализ и защита на мрежовата инфраструктура. Необходими са по-сложни информационни технологии, които могат да моделират нормалното поведение и определят анормалното. Тези системи за кибер защита трябва да бъдат гъвкави, адаптивни и ясни, но и в същото време в състояние да открият голямото разнообразие от заплахи и да направят интелигентния избор. Факт е, че повечето мрежови атаки се извършват от интелигенти агенти, като например компютърните червеи и вируси. Затова борбата с тях може да се стана с интелигентни полу-автономни или изцяло автономни агенти, които могат да открият, оценят и отговорят със съответното действие за защита. Тези интелигентни методи ще трябва да бъдат в състояние да управляват целият процес в отговор на една атака, т.е да се анализира и установи какъв тип атака се случва, какво се цели и какво е подходящото противодействие, и не на последно място как да се даде приоритет и предотвратяване на вторични атаки. Именно при тези тежки ситуации се нуждаем от иновативни подходи, като прилагането на методи от изкуствения интелект.

Цел на дисертационния труд, основни задачи и методи за изследване

В тази дисертация са изследвани методи от изкуствения интелект за приложение в компютърната мрежова сигурност с цел повишаване на надеждността и способността на превантивно детектиране и откриване на атаки към мрежовата компютърна сигурност.

За да може успешно да се реализира целта са използвани размита логика, обучение с утвърждаване и мулти-агентни технологии от изкуствения интелект за създаване на система за детектиране и откриване на атаки.

Във връзка с основната цел са формулирани следните задачи:

1) Изграждане на две софтуерни рамки: host based monitoring system и network gateway monitoring system (частично базирана на правила)

2) Използване на мулти-агент технология като парадигма за дизайн на софтуерните рамки на системата, както и прилагане на стратегията за опазване на защита в дълбочина вътре в нашите защитни механизми

3) Интегриране на различни технологии IPS в една платформа, за да се използват предимствата на различните интегрирани технологии

4) Прилагане на поведенчески анализ и техники за откриване, които се фокусират върху поведението на охранявания обект, а не поведението на заплахата или на хакер

5) Системата включва три основни концепции за управление на откриванията: обучение за размито укрепване, управление на знания (KM) и управление на множество агенти (MA) в основния архитектурен проект

Практическа приложимост

Разработени са тестови сценарии за оценка на ефективността на мулти-агент-базирана система за откриване на проникване и превенция, които ясно и еднозначно потвърждават постигнатата ефективност на предложения модел. Част от изследванията са направени в реална среда.

Разработени са различни алгоритми, които симулират ефективността на хибриден модел. Изградени са различни сценарии за симулация.

По отношение на разработените алгоритми, те могат също да намерят приложение в реална работеща мрежа.

Апробация

Резултатите от дисертационното изследване са апробирани в четири пленарни доклада на международни конференции, две статии в международни индексирани списания и една статия в национално списание.

Публикации

Основни постижения и резултати от дисертационния труд са публикувани в национални и международни издания. Общият брой на статиите е седем, от които има една самостоятелна. Две от статиите са публикувана в международни списания и една в национално, като едната е самостоятелна. Четири от статиите са публикувани в материали на международни конференции.

Структура и обем на дисертационния труд

Дисертационният труд е в обем от 196 страници, като включва увод, шест глави за решаване на формулираните основни задачи, списък на основните приноси, списък на публикациите по дисертацията и използвана литература. Цитирани са общо 116 литературни източници, като 105 са на латиница и 3 на кирилица, а останалите са интернет адреси. Работата включва общо 80 фигури и 21 таблици. Номерата на фигурите и таблиците в автореферата съответстват на тези в дисертационния труд.

II. СЪДЪРЖАНИЕ НА ДИСЕРТАЦИОННИЯ ТРУД

ГЛАВА 1. ВЪВЕДЕНИЕ

В глава 1 е направен преглед на методите от изкуствения интелект, които са приложими в мрежова и информационна сигурност, по специално в системи за откриване и предотвратяване на проникванията. Направен е сравнителен анализ с техните предимства и недостатъци и е избран подходящ и перспективен метод за извършване на изследванията. Бяха разработени набор от критерии за оценка на ефективността на откритието и нивото на противодействие. Изключително важно е да се постигне правилният баланс между фалшивите положителни резултати и фалшивите негативи. Неправилните положителни резултати (така наречените фалшиви аларми) могат да бъдат не по-малко вредни от фалшивите отрицателни резултати. Установено е, че методологията на мулти-агентни системи изтъква най-традиционните системи, базирани на изкуствен интелект при откриването на атаки, особено от неизвестен характер. Ефективността на откриването на опасности в мулти-агентните системи също надхвърля традиционните системи. Най-важните аспекти на системите за разпознаване и предотвратяване на проникване на различни агенти (IDPS) са висока точност, самообучение и устойчивост. Описаните в източниците практики показват, че резултатите от правилното откриване на заплахи, използващи системи, базирани на множество агенти, постоянно се увеличават, тъй като процентът на фалшивите аларми драстично намалява. Несъмнено подходите, основаващи се на няколко агента, потенциално биха могли да постигнат по-голяма гъвкавост, което ще ги направи още по-популярни в близко бъдеще.

Допълнително в тази глава са посочени мотивацията, дефиниране на проблема, целта и задачите в дисертацията, а именно:

1. Мотивация - Непрекъснатото нарастване на компютърните престъпления и огромните финансови загуби бизнеса. Това предизвиква необходимостта от нови подобрения в областта на сигурността и решения, който да имат свойството да бъде напред в новите технологии и една стъпка преди креативността на хакерите. Това ще доведе бизнес организациите до едно по-надеждно и сигурно ниво. От друга страна, най-бързите сфери за развитие на изкуствения интелект са могли да бъдат по-ефективно използвани като се има предвид, че интелигентните агенти могат да бъдат използвани за изпълнение на такива решения за сигурност, за да се постигне благоприятно действие.
2. Дефиниране на проблема - За изграждането на такава система за сигурност, която има способността да защитава мрежата (сървъри и хостове) срещу атаки и зловреден софтуер, и същевременно с това да бъде една стъпка пред креативността на хакерите и откриването на заплахи атаки от тип zero-day (открити технологични пропуски, но все още не коригирани от производителите със съответни софтуерни обновявания), има трябва да се разглежда редица предизвикателства за изпълнение, както следва :
    a. откриване на заплахи тип zero-day, без предварително познаване на техните сигнатури или характеристики, с ниско ниво на фалшиви сигнали.
    b. Системата трябва да работи в унисон с никаква възможност за провал, защото това може сериозно да повлияе на производителността на мрежата, както и някои опити за заплахи могат да останат неоткрити.
    c. Активно блокиране на зловредния трафик в мрежата и на системни обекти, като същевременно позволява на нормалния трафик и обекти да преминава свободно, и да отслаби настъпването на неверни положителни и фалшиво отрицателни резултати.
    d. Висока производителност и защита в реално време, без претоварване.

e. Избягване на огромното количество сигнатури и модели на атаки, използвани за идентифициране на злонамерен трафик и софтуер, както и идентифициране на заплахи тип zero-day.
f. Тези предизвикателства за изпълнение са критичен и комплексен проблем в днешно време, и се нуждаят от задълбочени проучвания и работа. Осигуряването на подобна система за сигурност, която има способността да се справи с тези предизвикателства може да се счита за голямо постижение в областта на сигурността на компютърните мрежи

3. Цел и задачи –
   a. Изграждане на две софтуерни рамки: host based monitoring system и network gateway monitoring system (частично базирана на правила)
   b. Използване на мулти-агент технология като парадигма за дизайн на софтуерните рамки на системата, както и прилагане на стратегията за опазване на защита в дълбочина вътре в нашите защитни механизми
   c. Интегриране на различни технологии IPS в една платформа, за да се използват предимствата на различните интегрирани технологии
   d. Прилагане на поведенчески анализ и техники за откриване, които се фокусират върху поведението на охранявания обект, а не поведението на заплахата или на хакер
   e. Системата включва три основни концепции за управление на откриванията: обучение за размито укрепване, управление на знания (KM) и управление на множество агенти (MA) в основния архитектурен проект

ГЛАВА 2. МУЛТИ-АГЕНТНИ СИСТЕМИ

Във втора глава са разглеждани мулти-агент-базирани стандарти, технологии и софтуерни инструменти. Направени са изводи, кои софтуерни платформи ще се използват в дисертационния труд.

В литературата съществуват много и различни определения за софтуерни агенти.

(1) Агентите са самостоятелни програмни единици, действащи автономно, за постигане на поставена от потребителя цел. Характерните особености на агентите са: автономност, адаптируемост, сътрудничество и мобилност.
(2) Агентът е единица, която може да се разглежда като възприемане на средата чрез своите сензори и да действа при тази среда чрез манипулатори.
(3) Автономните агенти са изчислителни системи, които съществуват в някаква сложна динамична среда, действат самостоятелно в тази среда и по този начин реализират набор от цели и задачи, за които те са проектирани.
(4) Интелигентните агенти изпълняват непрекъснато три основни функции: възприемане на динамичните условия на средата; извършване на действия, за да изпълнят условията в средата, както и мотиви за интерпретиране на възприятията, решаване на проблеми, изготвяне на изводи и действия.
(5) Агентът е изчислителен процес, който се изпълнява самостоятелно и комуникира функционално с друго приложение.

Агентите притежават следните основни характеристики: автономност, социалност, реактивност и про-активност.

- Автономност - възможността на агент да действа самостоятелно без пряката намеса на хора или други агенти и да има контрол върху собствените си действия и вътрешни състояния.

- Социалност - взаимодействат помежду си като използват език за комуникация.
- Реактивност - възприемат заобикалящата ги среда и реагират своевременно на промените, които се случват в нея.
- Про-активност - не просто действат в отговор на тяхната околна среда, а са в състояние да проявяват целево поведение (проява на инициативност).

Допълнителни характеристики на агентите са:

- Комуникативност – интерфейс/и към потребители и други агенти, по-специално съвместни (collaborative) агенти; агентите взаимодействат помежду си, като използват език за комуникация;
- Непреходност (continuity) – продължително и многократно изпълнение на функциите, не се отнася за всички типове агенти;
- Мобилност – миграция между възли (може и при ниска мобилност на кода), не за всички – например агенти за извличане на информация от разпределени документни системи;
- Адаптивност – еволюция на реакциите при еднакви параметри на средата, не се отнася за всички агенти;
- Самообучение – промяна на поведението, базирано на предишен опит;
- Целева ориентация – не действа просто в отговор на околната среда, а за постигане на дадена цел;
- Гъвкавост – действията не са по сценарий.

Не всички агенти притежават всички характеристики. В агент-базираните системи е възможно да се реализират част от тях, свързани със спецификата на решаваните задачи.
Интелигентните агенти имат пет основни типа:

1. Рефлекторни
2. Рефлекторни агенти, основани на модели на света
3. Целеви
4. Агенти, основани на полезност
5. Обучаващи се агенти

Агент-ориентирани технологии се разглежда като подобрение и разширение на Обектно-ориентирани технологии. Агент-мислене вместо Обектно-мислене помага в разработването на разпределени приложения с висока скорост на изпълнение, автономия и мащабируемост поради функцията на мобилни агенти. Таблица 2-1 показва просто сравнение между агенти и предмети.

Таблица 2-1 Сравнение между агенти и предмети

	АГЕНТ	ОБЕКТ
ХАРАКТЕРИСТИКИ	Агенти имат автономно поведение, което не може да бъде директно управляемо от външната среда.	Обектите са контролирани от външната среда
	Агентите имат собствена нишка на контрол	Има една нишка на контрол върху прилагането
	Агентите са обикновено самостоятелни изчислителни единици, които не трябва да бъдат използвани заедно с други компоненти, за да се реализират услугите, които те предоставят.	Обектите не са автономни и нямат съответното понятие за реактивно, проактивно, или социално поведение.

	С мобилни агенти, вместо разделяне код на две части, една от страната на сървъра втора от страната на клиента, целият код е в един агент и агента се движи, както му е необходимо. Няма клиент-сървър връзка.	Трябва да съществуват сървър и клиент. Данните се изпращат към сървъра и резултатите се връщат на клиента. Така обектът не се движи или пътуват.
МЕТОДИ ЗА КОМУНИКАЦИЯ	Агентите поддържат високо ниво на взаимодействие помежду си. Използват езици за комуникация на агенти (ACL).	Обекти използват различни техники за съобщения, като Simple Object Access Protocol (SOAP).
МОБИЛНОСТ И КОНТРОЛ	При мобилните агенти действителната имплементация се движи. Това ги прави по-лесни при вмъкването на по-нови обекти, тъй като те не трябва да бъдат предварително разположени и монтирани преди използването им.	Съществуващ обект се контролира дистанционно. С други думи - обекта съществува само на сървъра, но се контролира от клиента.

Една мулти-агентна система се състои от множество автономни модули - агенти, които работят заедно за решаване на проблеми, които са извън техните индивидуални възможности:

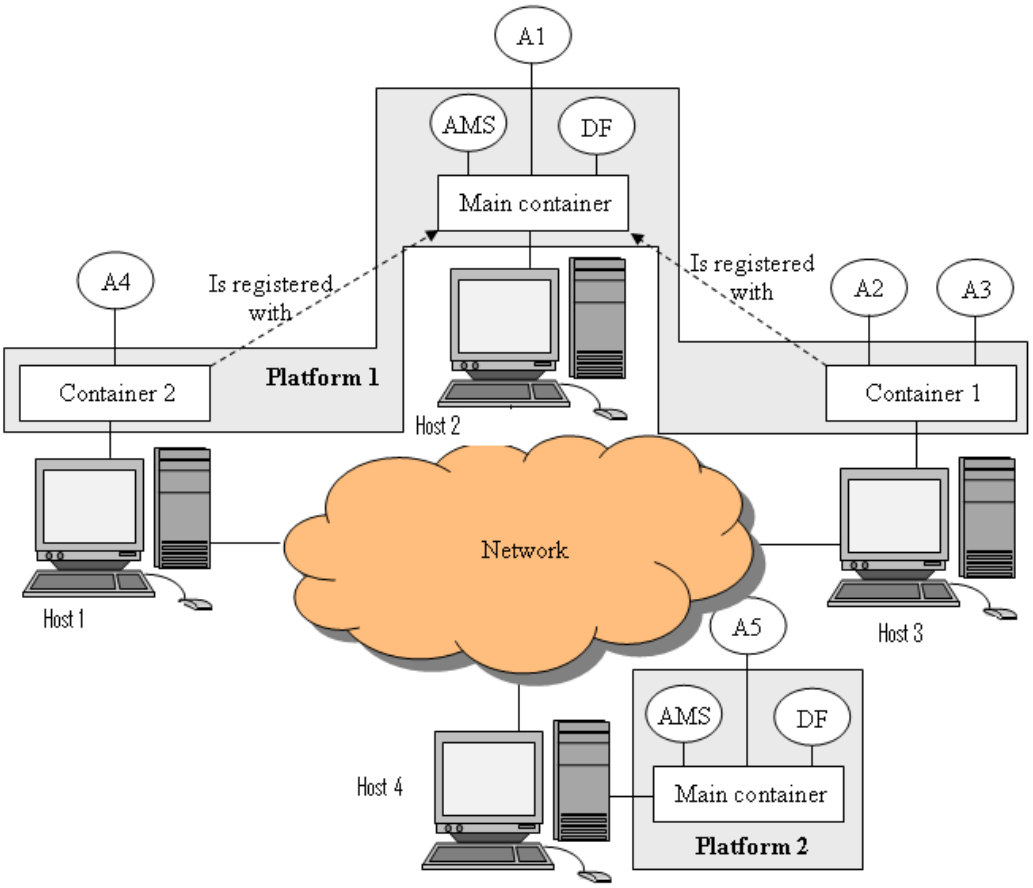
- Решаване на проблеми, които могат да бъдат твърде големи за един агент;
- Осигуряване на по-добра скорост и надеждност;
- Толериране на несигурни данни и знания.

Агентите в мулти-агентните системи се считат за автономни (т.е. независими, неконтролируеми), реактивни (т.е., в отговор на събития), проактивни (т.е. започване на действия по свое желание) и социални (т.е. комуникативни). Всеки агент може да има своя собствена задача и/или роля. Агенти и мулти-агентни системи се използват като метафора за моделиране на сложни разпределени процеси. Основните характеристики на мулти-агентните системи са:

- Изпълняват се в определена среда, с която взаимодействат;
- Състоят се от множество от агенти, които представляват активни компоненти в системата;
- Използват множество връзки, които свързват обектите и агентите;
- Агентите възприемат, създават, разрушават и модифицират множество от обекти в средата;
- Въздействието на агентите върху обектите се осъществява чрез множество от операции.

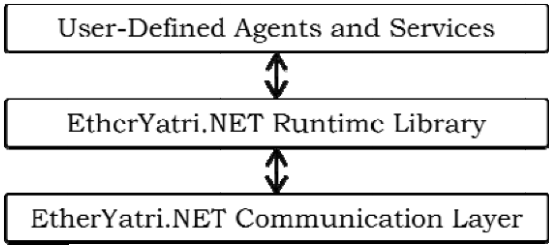
Изследвания в мулти-агентните системи наскоро доведе до развитието на практически програмни езици и инструменти, които са подходящи за осъществяването на такива системи. Повечето агент базирани програмни езици имат базова платформа, която изпълнява своите семантични техники за съответните аспекти като комуникация и координация между агенти. Най-новите езици ще бъдат придружени от интегрирана среда

за разработка (Integrated Development Environment IDE), предназначена за повишаване на производителността на програмистите чрез автоматизиране. Обикновено те ще предоставят функционалности, като например управление на проекти, създаване и редактиране на файлове, рефакториране, процеси по изграждане и тестване.



Фиг. 2-1 JADE архитектура

Агент-базираните програмни езици и платформи са широко имплементирани с помощта на Java. Примерни за такива са Aglets, MASON и Jade. Наскоро някои проучвания са насочени за създаване на платформи за агента, използващи Microsoft .NET технологии като EtherYatri.Net. .NET има много функции, които улесняват развиващите се мулти-агентни системи: като много по-интуитивна обектно-ориентирана софтуерна рамка за създаване и прилагане на мобилни агенти. .NET Framework има вградена услуга, която улеснява мобилността на код.



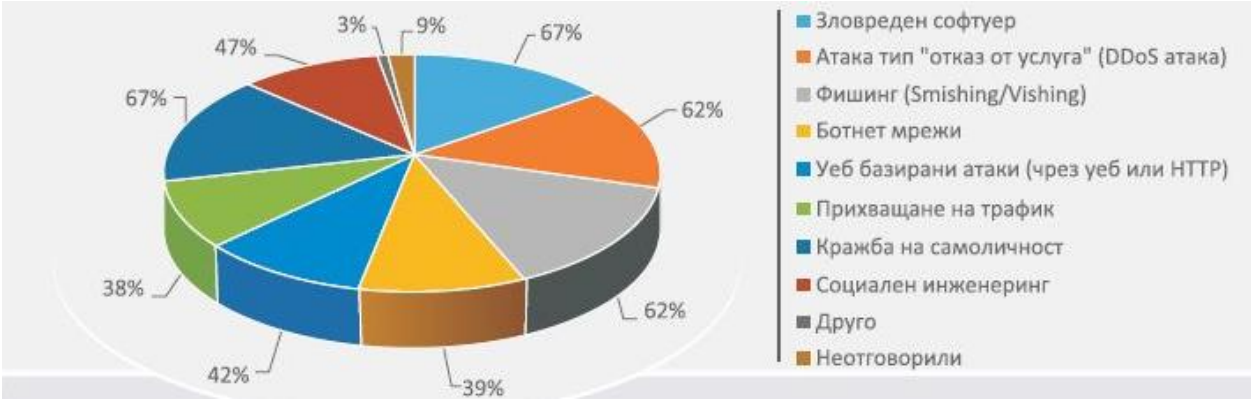
Фиг. 2-2 Архитектура на EtherYatri.Net

С прегледа на различните софтуерни рамки за агенти като ZEUS, JACK, Grasshopper-2, EtherYatri.Net и JADE изпълнението на системите е по-просто и улеснено за работа. Също така с интегрираните функционалности времето за разработка на приложения и цената е намалена. Агентите са много приспособими към едновременните връзки с други агенти.

ГЛАВА 3. СИСТЕМИ ЗА ОТКРИВАНЕ НА ПРОНИКВАНИЯ

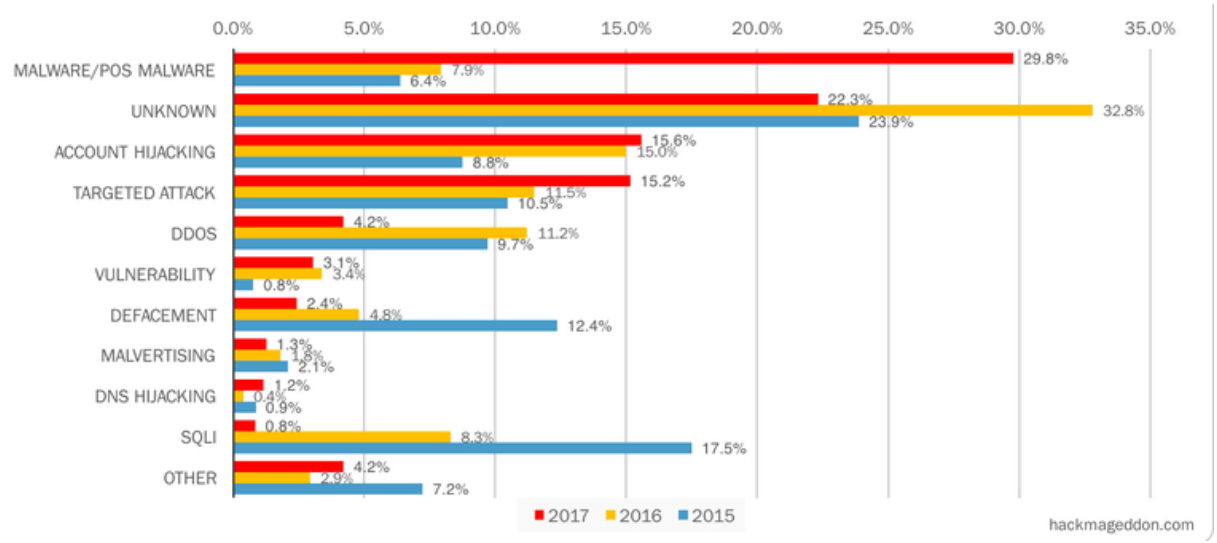
В трета глава се дефинират понятията и критериите за мрежова сигурност – надеждност, цялостност и поверителност. Направен е обзор на видовете мрежово базирани атаки – злонамерен код, DoS, DDoS, отвличане на сесия. Описани за различните системи за откриване и предотвратяване на проникванията, като е направен и анализ (класификация и сравнение) на различните подходи и е представено графично съотношение на традиционните системи

Според различни социални изследвания, петте най-разпространени киберзаплахи през 2014г. са били (Фиг. 3-2): зловредният код, кражбата на самоличност, атаката тип „отказ от услуга" или позната като DDoS атака (62%), фишингът (62%) и социалният инженеринг - 47%. Тези, наред с уеб-базираните атаки, са и най-сериозните заплахи за дейността на организациите.



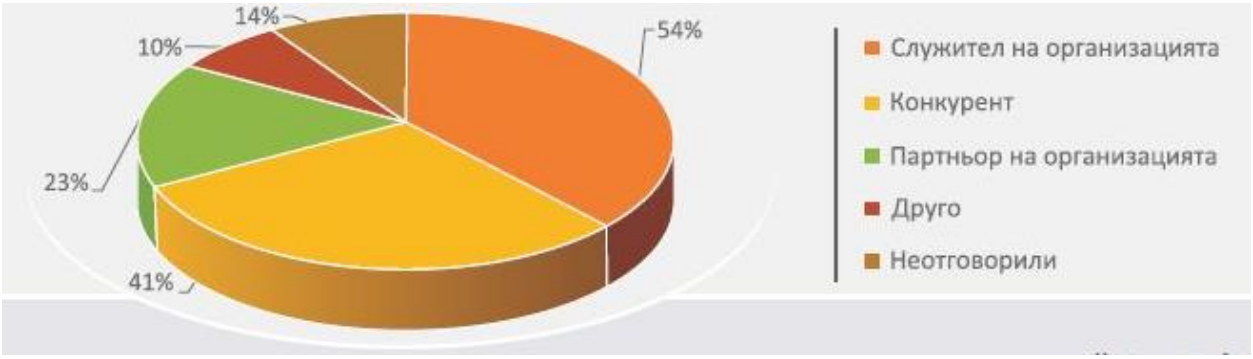
Фиг. 3-3 Кибератаките през 2014

През следващите години процентът на зловредният софтуер в статистика се увеличава. Според проучвания на HACKMAGEDON киберзаплахите през последните 3 години са се промени по следният начин:



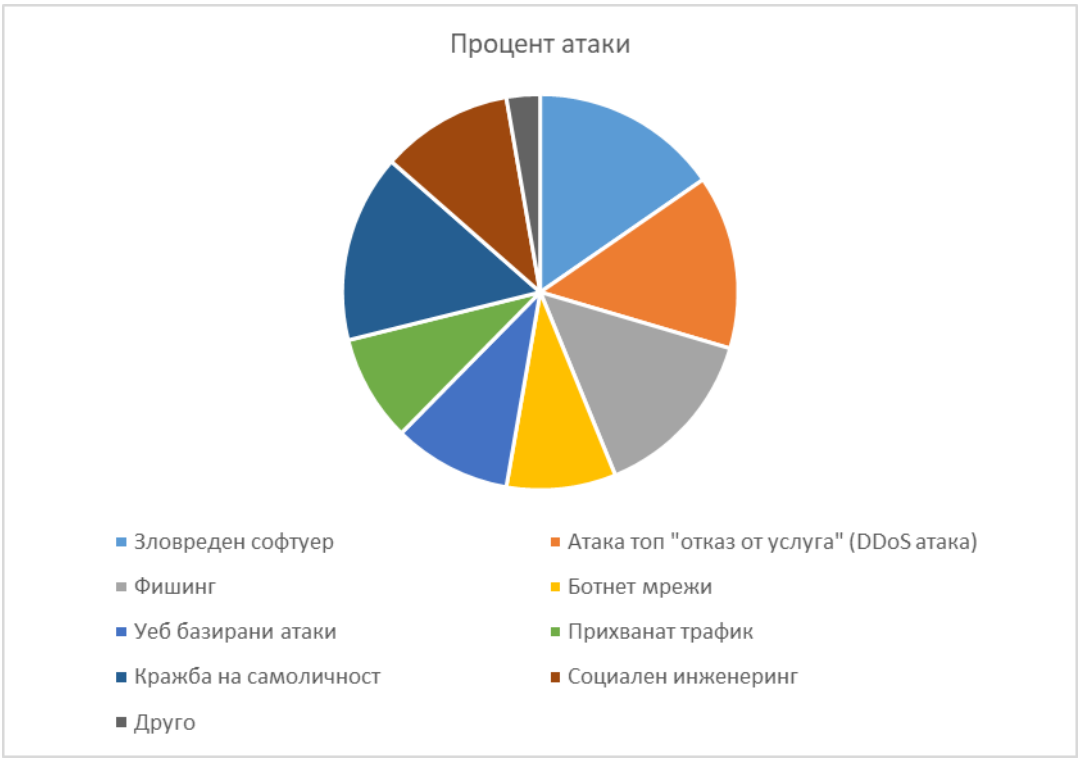
Фиг. 3-4 Топ 10 на заплахите през 2015,2016,2017 година

Служителите на организациите са основните източници (около 54%) на киберзаплахи, според проучване на Български киберцентър по компетентност за обучение и изследвания (фиг.3-4). При възникнали атаки организациите най-често се обръщат към организацията/компанията „майка". Не малък процент – 33% не докладват за инциденти поради липса на доверие в резултатите, които ще бъдат постигнати.



Фиг. 3-5 Източници на атаки

Броят на компютърните атаки в последните няколко години нарасна значително с развитието на Интернет. Факт е, че повечето мрежови атаки се извършват от интелигенти агенти, като например компютърните червеи и вируси (фиг. 3-7).



Фиг. 3-6 Атаките през 2015

В общия случай, заплахите могат да се групират условно на три надграждащи се нива:

- „известни известни“ – известни слабости, заплахи и пробиви, свързани с основната „триада“ на информационната сигурност (КИД – конфиденциалност, интегритет, достъпност);
- „известни неизвестни“ - комбинирани заплахи, свързани с информационната сигурност, разнообразие от APT, атаки срещу репутацията на организации и личности, кампании за дезинформация, пробиви в КИД в особено големи мащаби

(национални, регионални и световни), изискващи разширено и системно прилагане на КИД за всички активи в дигиталната екосистема;

– „неизвестни неизвестни“ – непредсказуеми, неочаквани заплахи в киберпространството, динамично променящи се рискове и комплексни въздействия с непредсказуеми последствия, които изискват гъвкавост и устойчивост на системите, организацията и процесите, и съответни изисквания при разработването и внедряването им - основните характеристики на състоянието киберустойчивост.

Фиг. 3-7 Класификация на атаките

Заплахите за комуникацията могат да бъдат условно разделени на няколко основни групи:

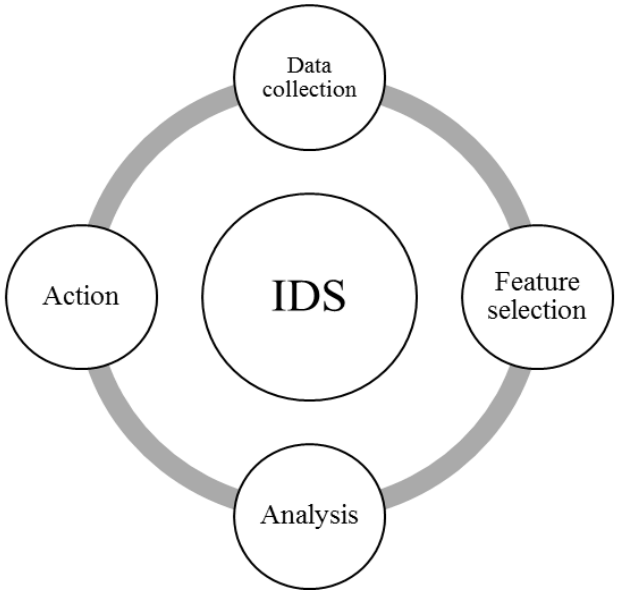
- Злонамерен код (malicious code)
- Атаки с цел невъзможност за предоставяне на услуга - denial-of-service (DoS) или разпределени DoS (DDoS)
- Сканиране
- Атаки, включващи отвличане на сесия/ достъп до ресурси
- Други (spam, spoofing, phishing email и т.н.)

Важно е да се разбере, че всеки може да извършва тези атаки на всеки и без да е компютърен хакер, необходимо е само да има малки познания по компютърни мрежи и съответният софтуер. Следователно, належащо е специалиста по мрежовата сигурност да разбира напълно начина на провеждане на атаката, за да може да й се противопостави.

DoS атаките се използват, за да се наруши нормалната операция на дадена система или мрежа. Цели се чрез DoS атаката да се претовари или спре достъпа до системата или дадения ресурс на мрежата. Такава атака води до 100% натоварване на мрежовите устройства или сървъри и те не могат да обработват подадените им пакети, в следствие на което ги отказват (drop). Целта на атакуващия е да се откаже (спре) достъпа на легитимни потребители до различните услуги. Този вид атаки често се използват наред с други атаки, като целта им е да „осакатят“ системите за сигурност преди реалната атака. Най-често при DoS атака се цели да се наруши мрежовата свързаност, като се правят опити да се отворят множество фалшиви TCP или UDP връзки. Устройството, към което е насочена атаката, се опитва да обработи всички заявки за връзки и по този начин се утилизират всички възможни ресурси. Съществуват три DoS атаки, които целят нарушаване на мрежовата свързаност.

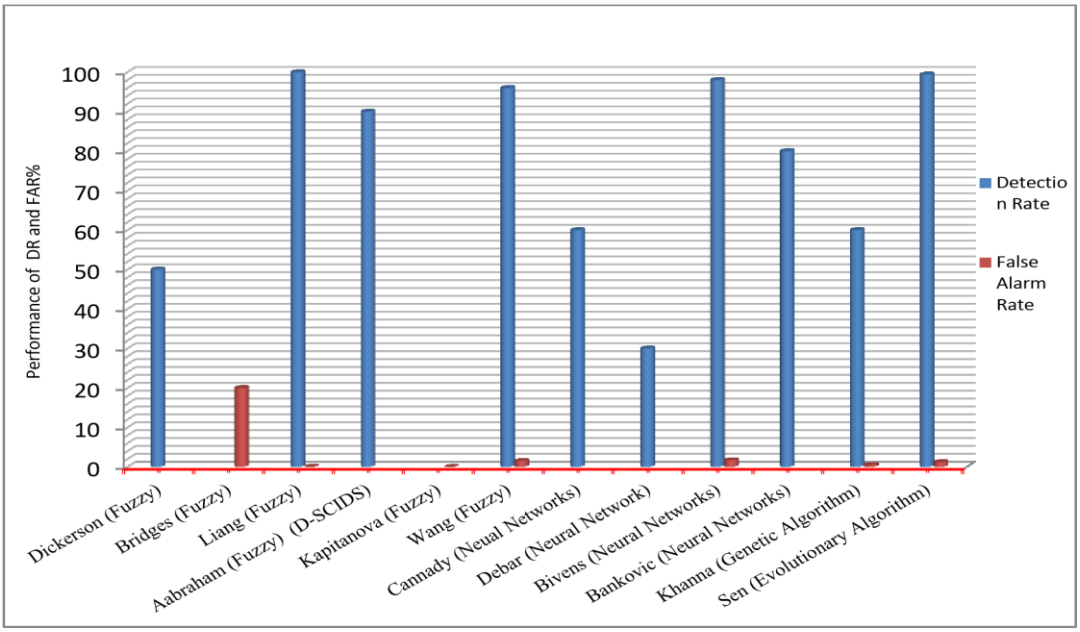
Една атака се състои от три фази - подготовка, изпълнение и след атака. В етапа на подготовка, нападателят събира информация, необходима за самата атаката. Действителната атаката се случва във фазата на изпълнение. В периода след атака, желаните ефекти (включително странични ефекти) на атаката са видими.

По този начин за откриване на проникване може да се определи като технология, която наблюдава компютърни дейности за предотвратяване на атака още в първа фаза. Откриването на проникване е процес на идентифициране и реагиране при злонамерена активност, насочена към компютър и/или мрежа.
Терминът IDS има по-общо значение и може да се отнася не само за мрежовите технологии, но и за анализ на други софтуерни процеси или опити за използване на системни ресурси без необходимите права.
Реализирането на IDS е софтуерно и изисква специално разработени алгоритми, които най-често използват сигнатури за откриване на атаките. Сигнатурите съдържат необходимите описания на атаката и могат да бъдат активирани или деактивирани от конфигурацията на IDS системата. Аналог на тези модули има и при антивирусните продукти – дефинициите на зловредния код. Типичните компоненти в една IDPS са сензор или агент, сървър за управление, сървър за бази данни и конзола.
IDS се състоят от четири основни елемента (Фигура 3-20) - за събиране на данни, функции за селекции, анализ и действие.

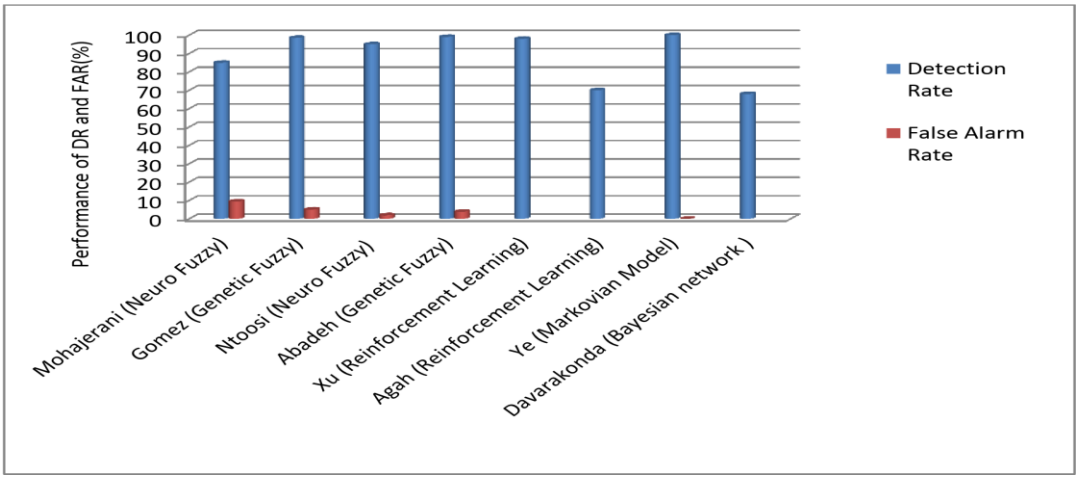


Фиг. 3-8 Функционалност на IDS

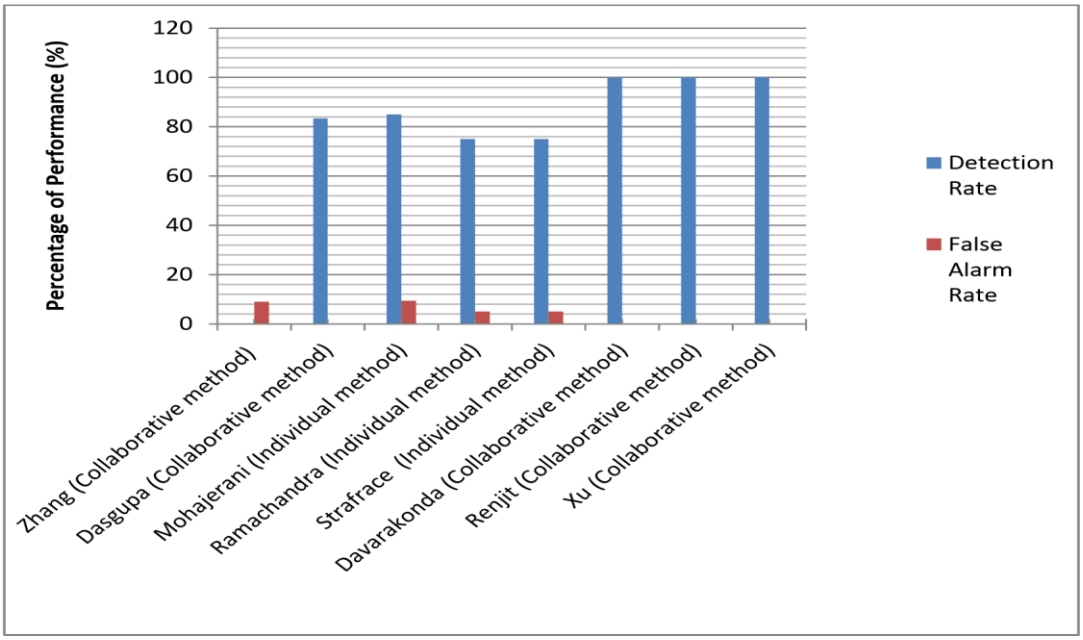
Събраните данни обикновено се записват във файл, от които трябва да бъдат анализирани. В IDS, базирани на правила, анализът се извършва чрез проверка на данните, като се сравняват с подпис или модел. Друг метод се основава на аномалия. Действието определя атаката и реакцията на системата.
Традиционно, подходите за откриване на прониквания и превенция са изследвани от два основни възгледа, а именно аномалия и откриване злоупотреба, макар че не съществува значителна разлика в характеристиките между тях. Поради липсата на по-подробен оглед на системите за установяване и предотвратяване, използващи мулти-агент базирани системи, тази теза представя класификация на три подкласа с перспектива в дълбочина от характеристиките: традиционен изкуствен интелект, computational intelligence и мулти-агент базирани. Съответно, внимателно са прегледани съществуващите подходи за откриване на проникване.
В съответствие с напредъка на мулти агент-базираните системи множество интелигентни системи за откриване на прониквания ги прилагат. Фигура 3-23 показва, че резултатите при правилното откриването с използването на мулти агент-базираните системи, постоянно се увеличава, като процента на лъжливите аларми драстично намалява. Без съмнение, мулти агент-базираните подходи могат потенциално да достигнат повишена гъвкавост, което ги прави още по-популярен в близко бъдеще.



Фиг. 3-9 Традиционни методи от изкуственият интелект



Фиг. 3-10 Computational Intelligence

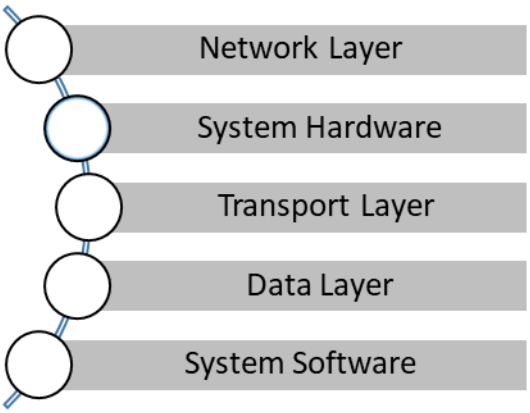


Фиг. 3-11 Приложения с мулти-агент системи

В тази глава, първо, бяха представени дефинициите за мрежова и информационна сигурност. Те трябва да осигурят надеждност, цялостност и поверителност на информацията.
На второ място бяха представени основните типове заплахи за компютърните системи и мрежи. Бяха представени и разгледани атаките, които нарушават сигурността на системата. Представено е и проучване на тенденциите в мрежовата и информационна сигурност.
На трето място - концепцията за система за откриване на проникване и превенция е анализирана в детайли, което свидетелства за значението на тази парадигма, за да се подобри ефективността на система и намаляване на оперативните разходи. В допълнение, в рамките на това широко понятие, основните системи за откриване и превенция на прониквания бяха оценени и категоризирани в три тенденции: традиционен изкуствен интелект, computational intelligence и мулти- агент-базирани.
На четвърто място, тази глава показва възможността от комбинирането на мулти-агент базираните методи със системите за откриване и превенция на прониквания. Изводът, който може да се направи е, че са нужно още много усилия за намирането на ефективни решения, от гледна точка на адаптивни техники за оптимизиране кооперативните подходи.

ГЛАВА 4. ПЛАНИРАНЕ НА ЕКСПЕРИМЕНТИТЕ

В глава четири са поставени изискванията, целите и техниките на експериментите. Направено е предложение на система за откриване на прониквания, като са описание нейните компоненти, функции и възможности.
Преди изграждането на система за мрежова сигурност, експерти по сигурността трябва да определят стратегията за отбрана, целите на системата, както и използваните техники и технологии за разработване на системата. Тези спецификации ще дадат възможност за създаването на една система, която е в състояние да постигне целите си, с висока степен на ефективност и съвместимост.
За проектиране и изпълнение на предложената система е използвана мулти-агент-базирани технологии, поради факта, че те се развиват бързо в областта на мрежовите приложения, както е обсъдено в глави 2 и 3. Също така са използвани поведенчески анализ и техника за откриване при анализа и модулите за откриване в предложената система. Поведенческите техниките служат за сравнение на поведението на обекти с предварително определен нормален профил на поведение. След това откриват отклонения от нормалното и го считат като злонамерено поведение.
Изграждане на нормален профил изисква много тренировки за обучение на системата, за да се получи чиста „снимка“ на един обект. Предложената система е инсталирана в чиста система, която е напълно свободна от всякакъв вид заплахи. Това означава, че са подготвени машини, които нямат достъп до интернет в началният стадий, за да може предложената система да си изгради профилите на обектите на закрила, като взема истинска „снимка“ на атрибутите на обекта. След като са изградени нормалните профили системата е свързана с достъп до интернет.
Предложеният модел се състои от две основни софтуерни рамки базирани на мулти-агенти - host based monitoring system (HBMS) и network gateway monitoring system (NGMS). Те работят на различни слоеве. Работата на предложената системата е разделена на пет слоя (фиг. 4-1) – network, system hardware, transport, data, system software.



Фиг. 4-12 Слоеве на предложената система

Слоевете може да бъдат обединени според тяхното предназначение. Първите три network, system hardware, transport се групират в областта на TCP/IP стека, а transport, data, system software на операционни ниво (Операционна система). Както може да се забележи транспортното ниво участва и в двете групи, поради факта, че то има важен двойствен характер.

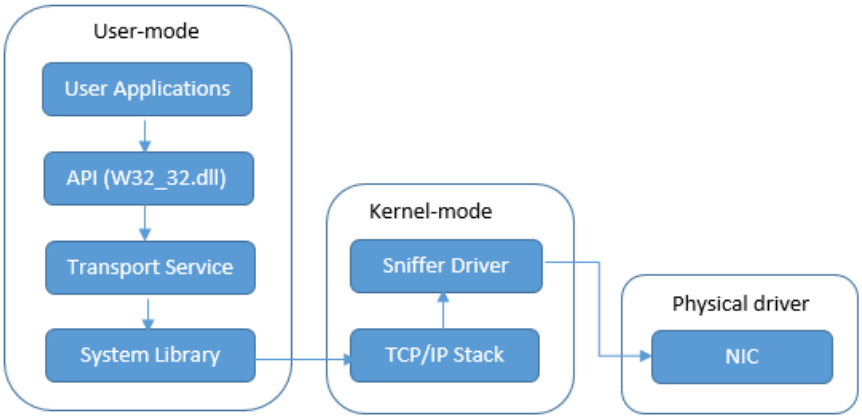
Network gateway monitoring system (NGMS) е мулти-агент-базирана софтуерна рамка, която се състои от две части – Network Prevention (NP) и Host Prevention (HP). Точката за достъп до интернет се осъществява чрез сървър, който има две мрежови карти. Едната за входящият интернет и една за изходящият. Налага се две карти за по-лесно следене и анализиране на заглавните части на пакетите.

NP компонентът работи на транспортния слой на модела TCP/IP, и проверява мрежовия трафик на втория мрежов интерфейс за откриване и предотвратяване на злонамерени пакети и следи от проникванията. Компонентът HP работи на приложния слой на модела TCP/IP и на слоя System Software на операционната система, и инспектира нейната дейност (потребителска дейност) и дейността на ядрото за откриване и предотвратяване на зловреден код.

Host based monitoring system (HBMS) е мулти-агент-базирана софтуерна рамка, която се състои от един компонент Host Prevention (HP). Този компонент е същият като при NGMS, но с леко разширение (описано по-долу).

Фигура 4-2 показва дизайна на NGMS и взаимодействията на агентите. Таблица 4-1 показва спецификациите на използваните агенти в NP компонента на NGMS, и таблица 4-2 показва спецификациите на използваните агенти в HP компонента.

NP компонентът функционира на мрежовия интерфейс, който е външен за сървъра. Работата му започва със следене на трафика от Sniffing Agent, който си комуникира с драйвера на ядрото отговарящ за връзката със TCP/IP стека (tcpip.sys). На фиг. 4-3 е представена съответната схема.



Фиг. 4-3 NP компонент

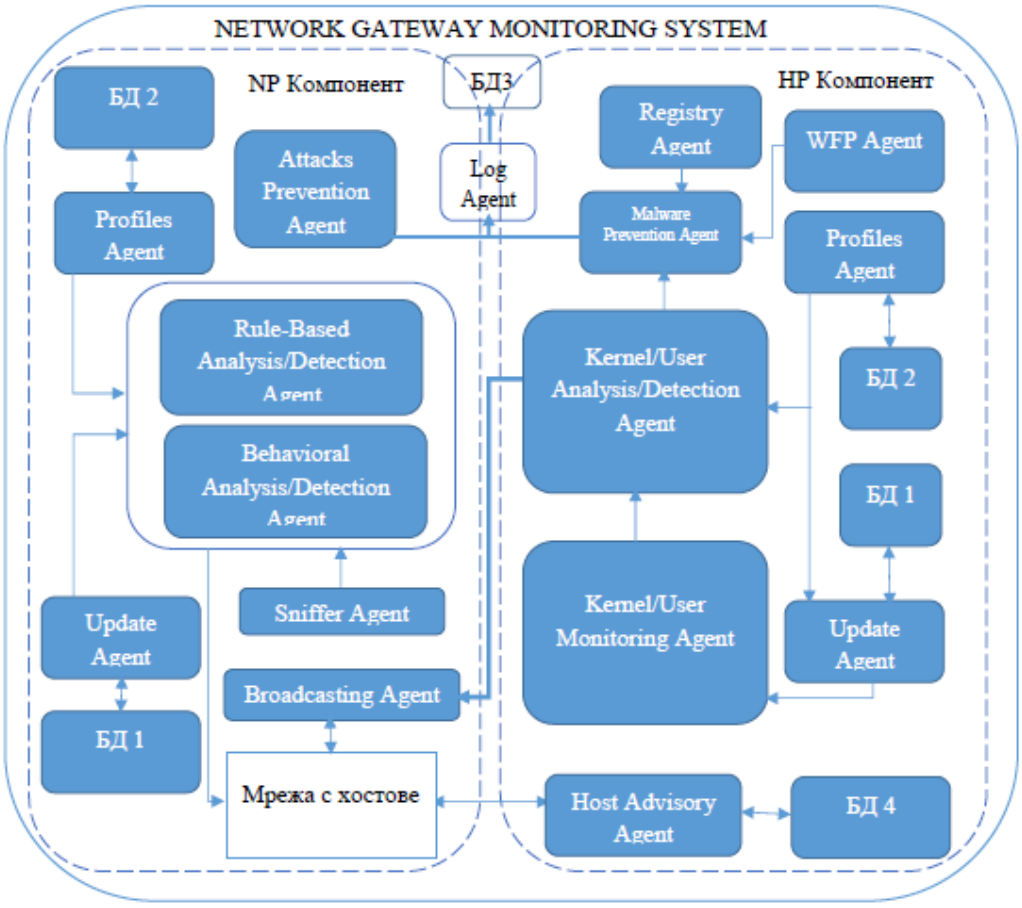
Таблица 4-1 спецификациите на използваните агенти в NP компонента на NGM

Агент	Вид на агента	Възприема информация от	Предавана информация	Действие на агента	Цел на агента
Sniffing Agent	Рефлекторен	TCP/IP Стек	Пакети	Записва и филтрира прихванатия трафик	Идентифицира мрежовия трафик
Behavior-Based Analysis/Detection Agent	Целеви	Sniffing Agent	Филтрира TCP/IP пакети	Анализира филтрираните TCP/IP пакети на базата на нормалния профил на системата	Открива атаки, базирано е на статистически нормалния профил, описващ нормалното мрежово поведение
Rule-Based Analysis/Detection Agent	Целеви	Sniffer Agent	Филтрира TCP/IP пакети	Анализира филтрираните TCP/IP пакети на базата на дефинирани правила	Открива мрежови атаки, базирано е на правила
Attacks Prevention Agent	Агент, основани на полезност	Analysis/Behavior и Rule-Based Detection Agents	Открити атаки	Предотвратява атака, чрез подходящи действия	Предотвратява атаки, чрез предприемане на подходящи действия
Logging Agent	Рефлекторен	Rules and behavioral Analysis/Detection и Malwares Prevention Agent	Анализи, открити и предотвратени резултати	Записва в логовете в База данни БД3	Записва резултатите, които после може да бъдат анализирани
Update Agent	Рефлекторен	БД1	Ъпдейти	Обновява информацията от анализите в двата вида Analysis/Detection Agents	Обновява информацията на фреймурка

Таблица 4-2 спецификациите на използваните агенти в компонента HP компонента

Агент	Вид на агента	Възприема информация от	Предавана информация	Действие на агента	Цел на агента
Kernel Monitoring Agent	Рефлекторен, основан	Ядрото на ОС	Дейности на ядрото на ОС	Следи някои от дейностите на ядрото на ОС	Изпраща следените дейности на Analysis Agents

	на модел				
User Monitoring Agent	Рефлекторен, основан на модел	WIN32 Subsystem	Дейности на потребителя	Следи някои от дейностите на потребителя	Изпраща следените дейности на Analysis Agents
Kernel Analysis/ Detection Agent	Целеви	Kernel Monitoring Agent	Дейности на ядрото на ОС	Анализира дейностите на ядрото и открива аномалии от нормалния профил	Открива зловреден код в ядрото на ОС
User Analysis/ Detection Agent	Целеви	User Monitoring Agent	Дейности на потребителя	Анализира дейностите на потребителя и открива аномалии от нормалния профил	Открива зловреден код при потребителя
WFP Agent	Целеви	WFP Service	WFP параметри и DLL	Следи WFP параметрите за да се избегне атака по тази услуга	Защитава изпълнимите файлове от потребителя
Registry Agent	Целеви	Windows registry	Важни ключове от регистрите	Предпазва регистрите от промяна на стойностите.	Защитава от неправомерни дейности на потребителя
Malwares Prevention Agent	Агент, основани на полезност	User& Kernel Analysis/ Detection Agents	Резултати от открит зловреден код	Вземане на решение за защита и запис на лог в БД3	Защитава от зловреден код
Logging Agent	Рефлекторен	User& Kernel Analysis/ Detection и Malwares Prevention Agent	Резултати от анализите, откритите и предотвратени атаки	Записва в логовете в База данни БД3	Записва резултатите, които после може да бъдат анализирани
Update Agent	Рефлекторен, основан на модел	БД1	Ъпдейти	Обновява информацията от анализите в Analysis/Detection Agents	Обновява информацията на фреймурка
Profiles Agent	Рефлекторен, основан на модел	БД2	Анализиран нормалният профил на обект	Изпраща профил на агентите User и Kernel Analysis /Detection Agents	Предоставя на User& Kernel Analysis/Detection Agents необходимата информация



Фиг. 4-2 Network Gateway Monitoring System

ГЛАВА 5. ЕКСПЕРИМЕНТАЛНА ПОСТАНОВКА И АНАЛИЗ НА РЕЗУЛТАТИТЕ

В глава пет е направена експериментална постановка. Описание са техническите изисквания към системата и лабораторната обстановка. Направени са експерименти на различни нива в системата и анализ на резултатите.

Предложената система е имплементирана и тестване в средата на Microsoft Windows и неговото семейство от операционни системи. За да може системата да отговаря на поставените цели и изисквания са използвани следните средства:

1) .NET технологии (C#)
2) C
3) Pyton
4) EtherYatri.Net – набор от библиотеки за разработване на мобилни агенти в средата на Microsoft .Net Platform. На фигура 5-1 е показана архитектурата на EtherYatri.Net
5) Microsoft Driver Development Kit (DDK) – е набор от програми за разработване на софтуерни и хардуерни драйвери за Windows
6) Kali OS - Предшественик на Kali Linux е операционната система BackTrack. Тази ОС оглавява списъка с най-добрите операционни системи за хакване. Kali Linux се базира на Debian и се разпространява с над 600 предварително инсталирани софтуерни инструмента за откриване на слабости в информационната сигурност. Софтуерът редовно се актуализира и се предлагат дистрибуции и за ARM процесори, както и дискови образи за VMware. Kali Linux често се използва в съдебномедицинската работа и има възможност за стартиране от флаш-стик. Kali Linux създава перфектна среда за разкриване на уязвимости.

За оценяването на IDS, за всяка възможна тестова стойност, има два вида грешки: фалшиво положително (FP) и фалшиво отрицателно (FN). FP възниква, когато дадено

събитие е предвидено като натрапчиво, но всъщност е нормално, докато FN се случва, когато се случи наистина натрапчиво събитие, без да бъде разпознато като едно. От друга страна, истинската положителна (TP) измерва дела на действителните положителни резултати, които са правилно идентифицирани като такива, докато истинската отрицателна (TN) измерва дела на негативите, които са правилно идентифицирани като такива.
Ефективността на всеки класификатор може да се определи количествено, като се използват мерките за измерване на степента на откриване (DR) и общата точност (OA). DR показва процента на истинските пробиви, които са успешно открити:

$$DR=\frac{TP}{TP+FN}\times 100\%$$

OA се изчислява като общият брой на правилно класифицираните прониквания, разделен на общия брой наблюдения:

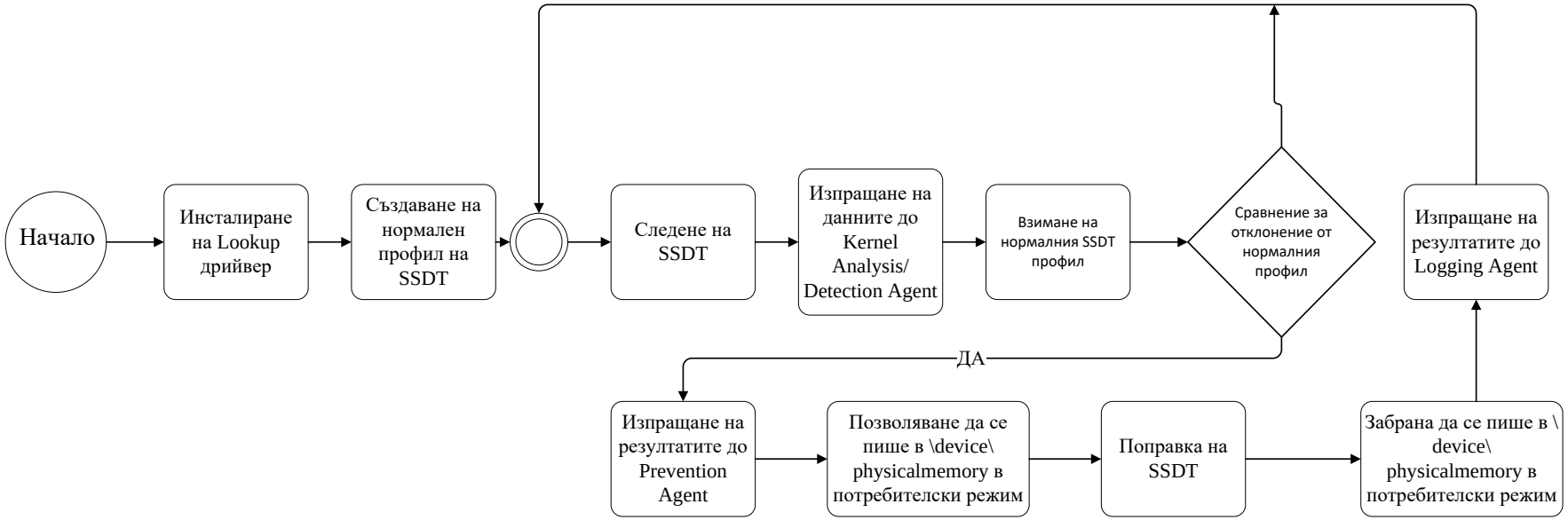
$$OA=\frac{TP+TN}{TP+TN+FP+FN}\times 100\,\%$$

Ефективната IDS изисква висока степен на DR и OA, като същевременно поддържа ниски нива на фалшиви аларми. Точността е от решаващо значение за разработването на ефективна IDS, тъй като високата скорост на FP или ниската DR ще го правят практически неизползваема.
Системата е инсталирана и тествана в Центъра за Информационни Ресурси при Технически Университет София. Производителността на предложения прототип е тествана в мрежа с 180 работни места. Всяко работно място разполага с Intel core i5-4570 Processor, 3.20 GHz, 6MB cache, 4 cores/4 threads, 4 GB DDR3 RAM с 1333 MHz и Windows 7/XP. Скоростта на данните на мрежата е 1 Gbps. Диапазона на различните активни потребители е от 5 до 170, а средната натовареност на системата се записва. При тестване то на системата, някои от атаките са прилагани директно върху работното място върху което ще се извършва теста. Част от сценариите за симулиране на атаките се базират основно на:

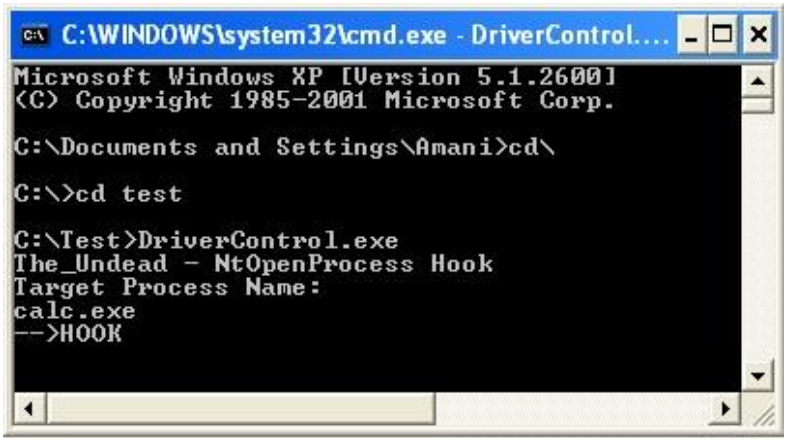
- модела на „веригата на убийството (kill chain)“ - военна концепция, свързана със структурата на атаката, която е адаптирана към информационната сигурност от специалистите от корпорацията "Локхийд-Мартин", които я използват като метод за моделиране на проникване в компютърна мрежа. Този модел постепенно (въпреки някои критики) се приема в общността за информационна сигурност за защита на данните чрез определяне на етапите на кибератаките и съответни мерки за противодействие на всеки етап.
- т.н. „матрица на заплахата“, която отразява моделите на минало и настоящо поведение и историческия преглед на проведени кампании, като се съсредоточава върху успешните атаки.

5.1 HP компонентът

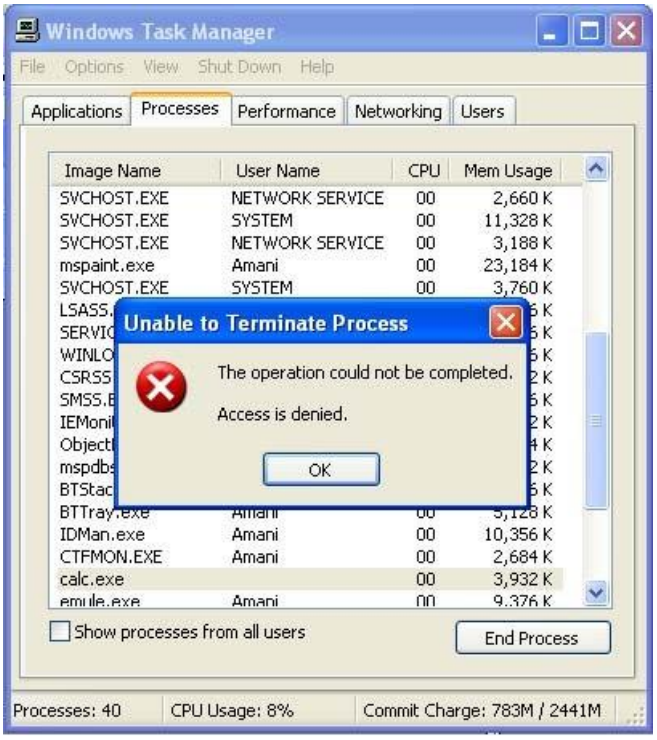
HP Компонентът е основна част от двете софтуерни рамки Network Gateway Monitoring System и Host Based Monitoring System. Предназначението му е да защитава всеки един обект в рамките на операционната система. Разгледани са два основни случая – атака към ядрото на ОС и атака към потребителската активност. В първият случай е разгледана атака към SSDT обект, а във втория превенция от вмъкване на зловреден код.
За тестване на HP компонентът от SSDT hook атака е използван Rootkit NtOpenProcess Hook (фиг. 5-8). Този rootkit използва SSDT, за да се прикачи към процесите от ядрото и да предотврати достъп до програмите от потребителя и да откаже всякакъв достъп до тях. На фигура 5-9 е показано държанието на въпросният rootkit и влиянието му върху операционната система. На фигура 5-10 е показан екран как предложената система го засича.



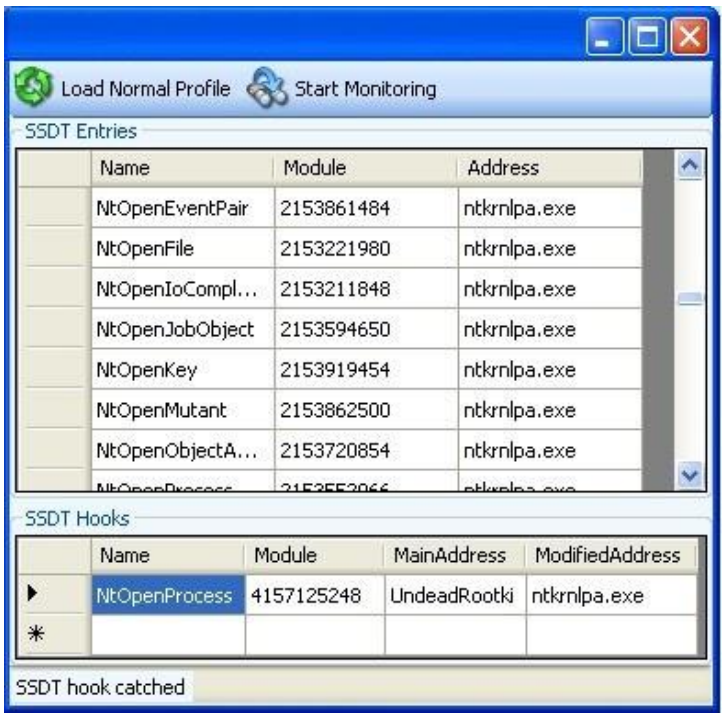
Фиг. 5-13 Сценарий на защита на SSDT компонентите



Фиг. 5-8 Rootkit NtOpenProcess Hook



Фиг. 5-9 Процесите от ядрото с инсталирания rootkit

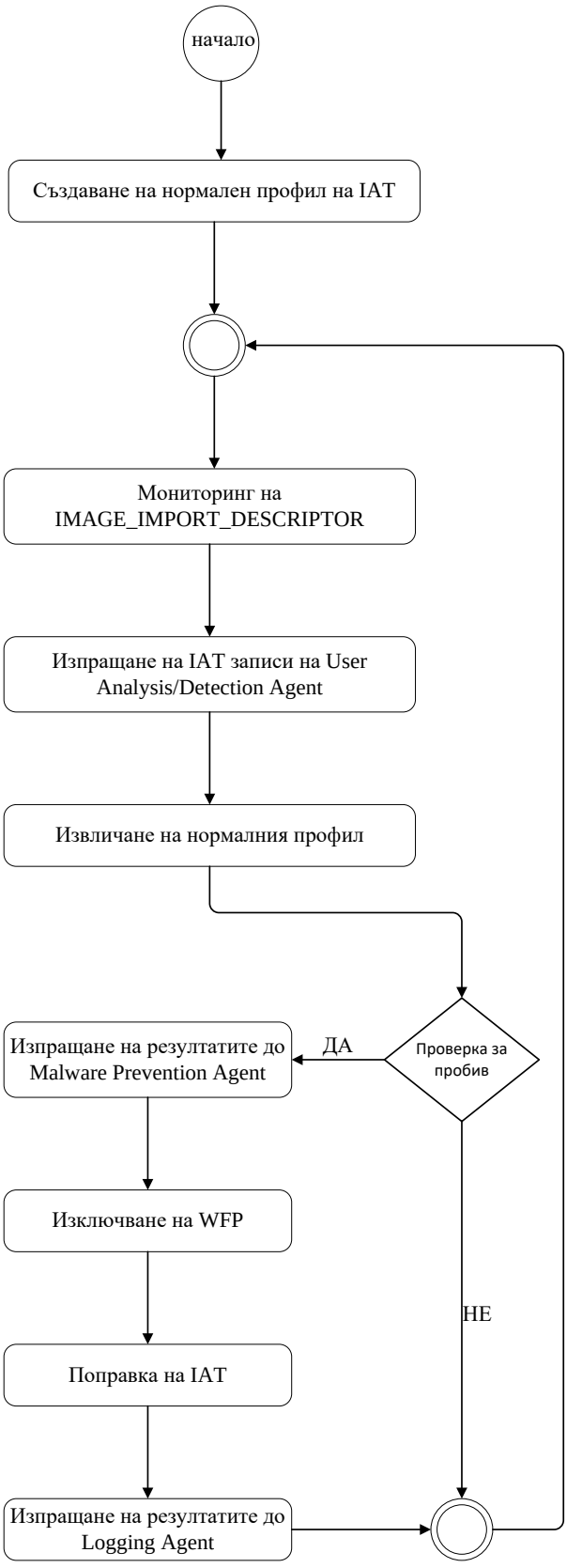


Фиг. 5-10 Системата засича “NtOpenProcess Hook” rootkit

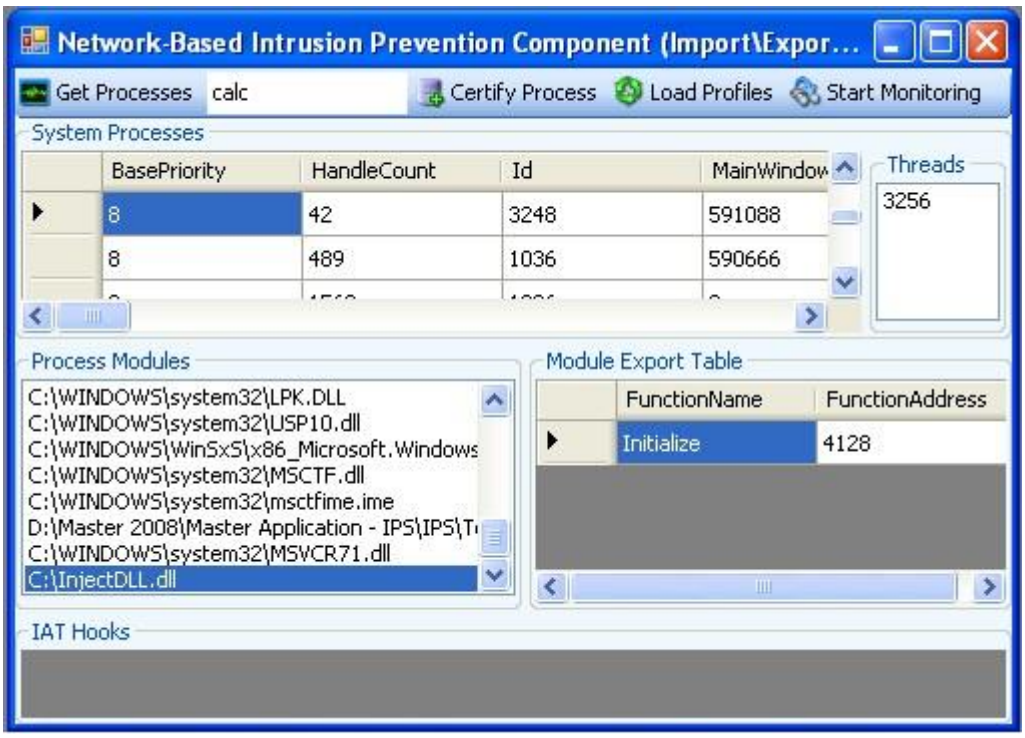
За тестване на HP компонентът за защита от IAT\EAT атаки е използван Import Table Runtime Redirector rootkit. Той вмъква зловреден код в определен процес по време на неговата работа. Използван е още един rootkit InjectDLL, който IAT rootkit и вмъква DLL в адресната таблица на определен процес. На фигура 5-15а е показана действието на първият rootkit, който оказва действие върху ShellAboutW() от Shell32.dll и фигура 5-15б показва действието на вторият rootkit, който вмъква зловреден код с име InjectDll в адресната таблица на процеса. На фигури 5-17 и 5-16 е показан прихващането им от предложената система.

а) Калкулатора преди атаката **б) Калкулатора след атаката**

Фиг. 5-15 Калкулатора преди и след атаката



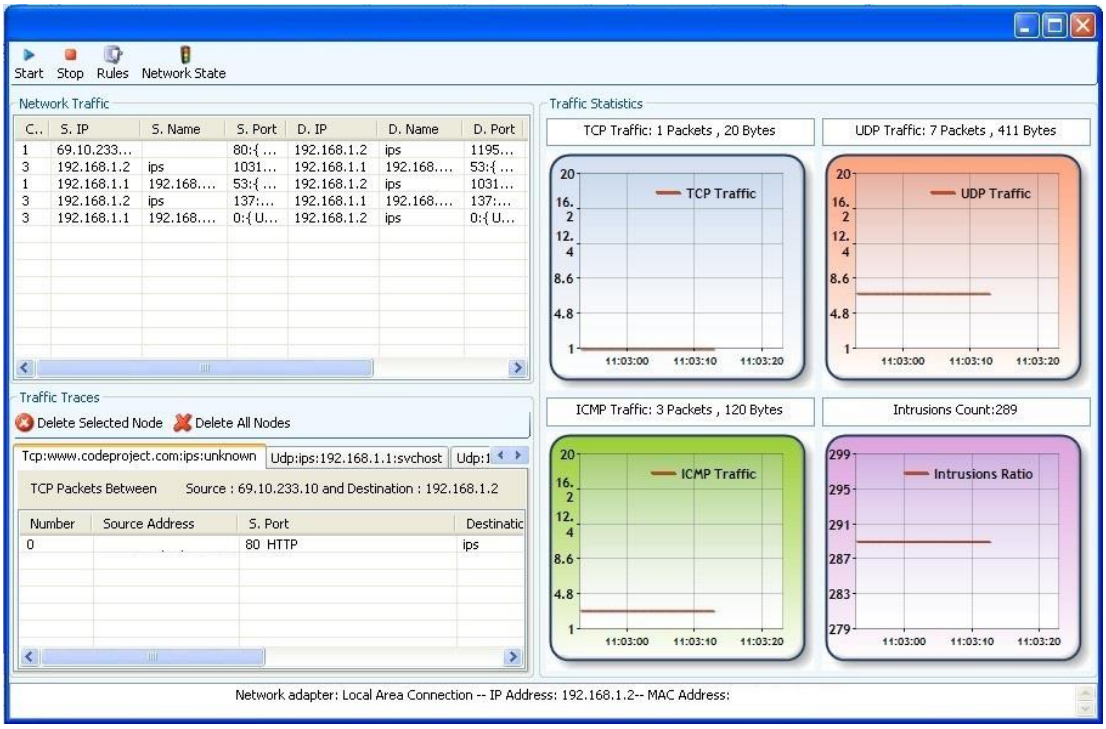
Фиг. 5-14 Диаграма на сценарий за защита на хостове и сървъри от Import Address Table (IAT) rootkits



Фиг. 5-16 Открит rootkit

5.2 NP компонентът

NP Компонентът е част от Network Gateway Monitoring System, която може да се определи като мрежови сървър за защита на мрежата от атаки като foot-printing, мрежово сканиране, SYN flood, ping sweep, сканиране на портове, отвличане на сесия и други. Предложената система е тествана с помощта на Kali OS и материали от Certified Ethical Hacker (CEH).
На фигура 5-18 е показан екран от NGMS системата (Приложение: SecurityMonitor.cs, ClientOperationInformation.cs, NetworkMonitor.cs), който показва съотношението на трафика и прониквания през времето. В таблица 5-2 е дадена статистика на откритите прониквания и атаките към системата.

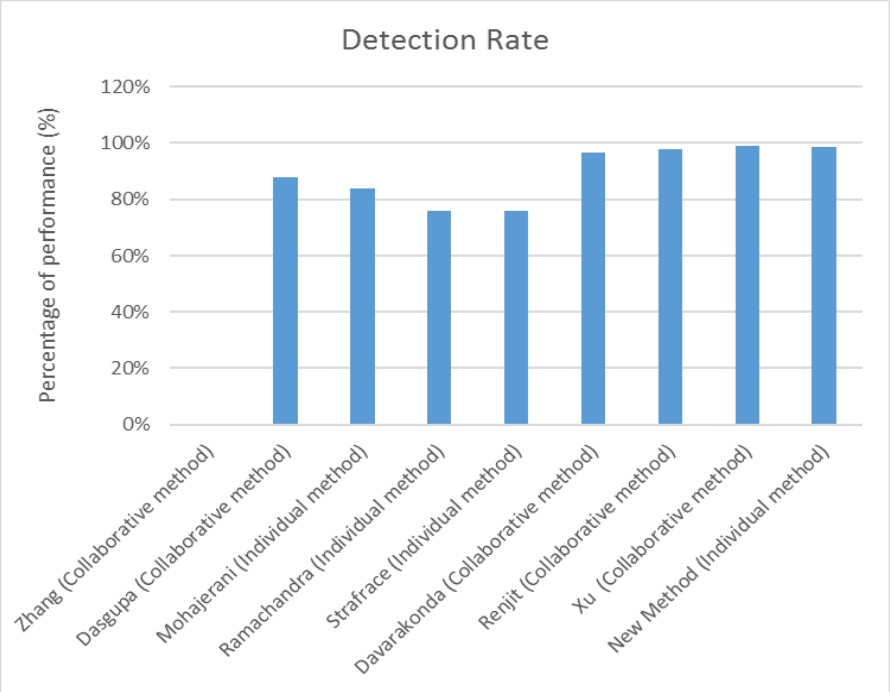


Фиг. 5-18 Network Gateway Monitoring System

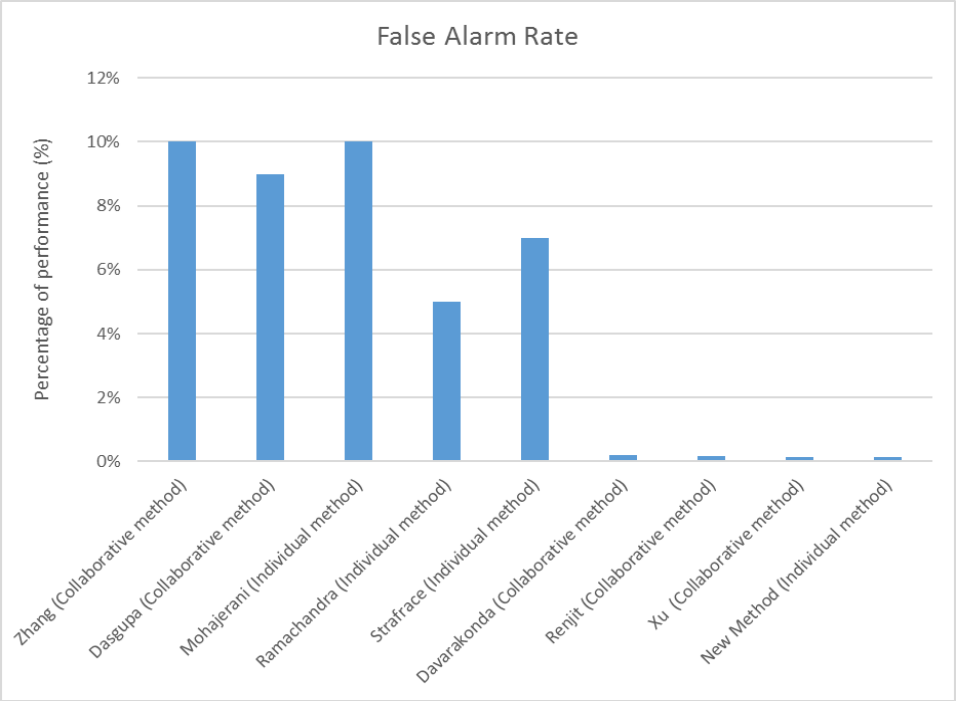
Таблица 5-3 статистика на откритите проникванията и атаките към системата

Име на атаката	Програма	Получени пакети	Открити атаки	Време (sec)	Процент
Network Scan	Angry IP, HPING2, Ping Sweeps	3000	600	20.25	67%
Port Scan	Nmap, NetScan Tools Pro, IPScanner	3300	700	22.3	54%
Enumeration	SuperScan 4, enum, PsGetSid	2000	500	10	54%
Smurf Attack	Land and Latierra	500	400	15.5	68%
SYN Flooding	NemeSys	2000	2000	25.4	62%
Ping Flooding	Crazy Pinger	2000	400	25	53%
Session Hijacking	RST Hijacking Tools, Remote TCP Session Reset Utility	900	900	30	74%

На базата на направените изследвания на съществуващите мулти-агент базирани методи от изкуствения интелект беше определен среден процент на тяхната успеваемост като беше приложена всяка една от атаките посочени в таблица 5-2. На фигура 5-19 може да се види сравнителен анализ успеваемостта на различните методи. На следващата фигура (фиг. 5-20) са измерени отношенията на фалшиво позитивните аларми на системите.



Фиг. 5-19 Сравнение на мулти-агент базираните методи за откриване на атаки 1



Фиг. 5-20 Сравнение на мулти-агент базираните методи за откриване на атаки 2

Разработени са сценарии за тестване, с цел верификация и оценка на ефективността на архитектурата.

Успешно са проведени експерименти, с цел верификация и оценка на отделните компоненти и на цялата платформата. Тези експерименти удостоверяват, че системата отговаря на изисквания на спецификациите.

Предложената система успява в откриване на атаките и зловреден код, които са насочени защитаваната система, с висока степен на точност и в реално време. Компонентът NP успява да характеризира нормалното поведение на TCP\IP протокола и да се открие най-простите атаки, целящи да засегнат заглавната част на пакетите. Компонент HP доказа високата си способност при защита срещу зловреден код, който влияе на операционни системи Windows, независимо дали зловредният код е в ядрото или насочен към потребителската дейност.

ГЛАВА 6. ХИБРИДЕН МОДЕЛ НА СИСТЕМА ЗА ОТКРИВАНЕ НА АТАКИ

На базата на направените до момента експерименти се доказва, че мулти-агент базираните системи за откриване и предотвратяване на атаки са ефикасни. Поради факт, че резултатите могат да бъдат подобрени и направения задълбочен анализ на експерименталното изследване на най-перспективните методи бяха направени допълнителни проучвания за създаване на модифициран хибриден модел. Световната практика отбелязва вече значителен брой от разнообразни приложения на „изкуствен интелект“ в компютърната сигурност. Без да се прави опит за изчерпателна класификация, може да се разделят тези приложения в две основни направления:

А. Условно наречени „разпределени“ или „мрежови“ методи:

А1. Мулти-агентни системи от интелигентни агенти;

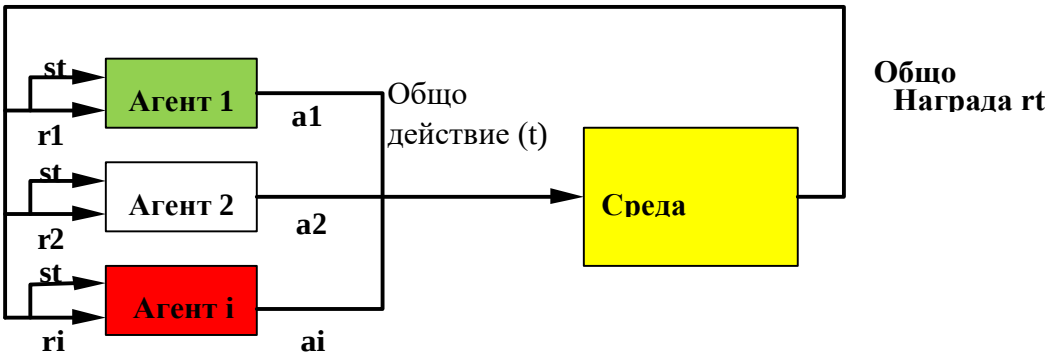
А2. Неврони мрежи;

А3. Изкуствени имунни системи и генетични алгоритми и т.н.;

Б. Условно наречени „компактни“ методи:

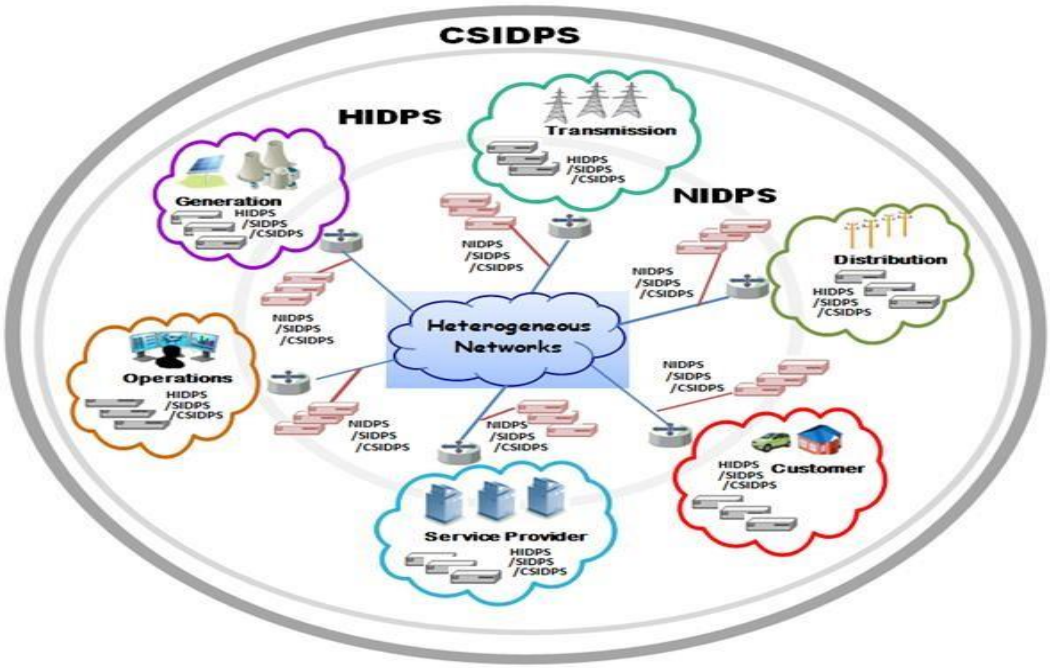
Б1. Системи за машинно самообучение (Machine Learning), в т.ч.: асоциативни методи, индуктивно логическо програмиране, Бейсова класификация и пр.
Б2. Алгоритми за разпознаване на образи;
Б3. Експертни системи;
Б4. Размита логика и пр.
Наличните академични ресурси показват многобройни реализирани приложения в борбата с компютърните престъпления. Изборът на приложение от изкуствения интелект, което да бъде включено в експеримента, е направено въз основа на анализ на източниците на информация относно подобни приложения и на резултатите от подобни експерименти.
През последните години системите за откриване и предотвратяване на проникванията станаха много важни. За да се справят с атаките на сигурността по отношение на инфраструктурата през последното десетилетие, организациите за сигурност обърнаха специално внимание на спестяването на разходи, като концепцията за IDPS е от особен интерес. От тази гледна точка, самооптимизацията обикновено включва настройка на мрежовите параметри. Независимо от това, наборът от параметри на мрежата, които могат да бъдат оптимизирани, е изключително голям, тъй като на него се изпълняват безброй IDPS алгоритми и техните параметри трябва да бъдат усъвършенствани. Освен това, дори ако процесът на оптимизация се извършва само на няколко важни параметъра, връзката между тяхната настройка и работата на мрежата не е ясна. По тази причина оптимизирането на параметрите на IDPS трябва да се извършва интелигентно. В резултат този тип системи биха могли да променят параметрите си по отношение на интелигентни IDPS (IIDPS), основани на разузнаването, за да постигне оптимални резултати без човешка намеса. Въпреки това, безброй параметри могат да се променят дистанционно в системата на IIDPS. От гледна точка на оператора настройването на параметрите на IIDPS, които не са обвързани с времето, е предпочитаната алтернатива.
Оптимизационните алгоритми помагат да се сведе до минимум (или да се максимизира) функцията на обекта E(x), която е просто математическа функция, зависима от вътрешните параметри, които могат да бъдат изучавани от модела, които се използват за изчисляване на целевите стойности (Y) от групата на прогнозни (X), използвани в модела. Вътрешните параметри на модела играят много важна роля за ефикасно и ефективно обучение на модела и за постигане на точни резултати. Ето защо използваме различни стратегии и алгоритми за оптимизация, за да актуализираме и изчислим подходящи и оптимални стойности на параметрите на този модел, които влияят върху процеса му на обучение и неговите резултати.
На базата на направено проучване на използваните интелигентни методи и двете основни категории на алгоритми за оптимизиране може се каже, че предимствата на втория ред пред първия са, че първо не пренебрегва кривата на повърхността и второ по отношение на производителност е по-добър. Основният недостатък е, че производна може да е малко скъп за намиране и изчисляване метод.
От гледна точна на използване може да се направи заключение, че:
1. Алгоритмите за оптимизация от първи ред са лесни за изчисляване и отнемат по-малко време, при работа с големи масиви от данни;
2. Алгоритмите за оптимизация от втори ред са по-бързи само когато е известна втората производна, в противен случай са по-бавни и заемат повече памет.
Динамичните среди имат не само множество агенти, но и множество последователни решения, които увеличават тяхната сложност. В тези настройки агентите трябва да координират и да обсъдят текущото състояние на динамичната си среда с много ограничена информация. Обикновено агентите в динамична среда не могат да наблюдават действията на други агенти или да видят какви възнаграждения получават като последствие, въпреки че действията на другите агенти оказват влияние върху непосредствената им среда, заедно с наградите, които могат да получат. В много сложни среди агентите може да не са наясно, че съществуват други агенти и могат да

интерпретират тяхната среда като нестационарна. Подобни, еднакво сложни среди позволяват на агентите да имат достъп до информация, но пространствата за стойността не са благоприятни за учене поради тяхната сложност и необходимата координация между агентите. Преди да може да се развие ефективен мулти-агентен подход, трябва да се обърне внимание на всички тези предизвикателства. Стандартен учебен модел за укрепване на множество агенти е представен на фигура 6-5.



Фиг. 6-5 Множество агенти, действащи в една и съща среда

Независимо от учебната сложност, търсенето на системи с множество агенти продължава да се увеличава. В случаите на системи, които са децентрализирани и единни, централните методи на обучение са непрактични. Такива системи могат да бъдат намерени, когато данните са подложени на смущения, причинени от множество конфликтни цели или ако един централизиран контролер изисква твърде много ресурси. Настройките на множество роботи, разпределеното балансиране на натоварването, децентрализираното мрежово маршрутизиране, електронните търгове и системите, предназначени за управление на трафика, са примери за такива типове системи.
В резултат на търсенето на адаптивни мулти-агентни системи и сложността на справянето с взаимодействащите обучаеми, все по-голям брой изследователи са работили за разработването на методи за подсилване.
Фигура 6-6 показва комбинацията от мрежови и хост-базирани IDPS (NIDPS, HIDPS) в напълно разпределена рамкова структура в мрежова среда с Хибридна Интелигентна IDPS Тази формация е лесно приложима към всяка мрежа и нейните изисквания. Той демонстрира влиянието на мулти-агентната система за компютърна разузнавателна техника върху повишаването на ефективността на откриване и фалшивите аларми. В контекста на Хибридна Интелигентна IDPS техниките обучение с утвърждение са адекватни за оптимизиране на мрежовите параметри поради сложността и динамиката на мрежата. Основните предимства на прилагането на такива техники са спестяването на разходи и подобрената производителност на мрежата.



Фиг. 6-6 Хибридна Интелигентна IDPS

Модифицираният NP компонент е част от Network Gateway Monitoring System, която може да се определи като мрежови сървър за защита на мрежата от атаки като foot-printing, мрежово сканиране, SYN flood, ping sweep, сканиране на портове, отвличане на сесия и други

Съвместната интелигентна сложност на IDPS включва три основни концепции за управление на откриванията: обучение за размито укрепване, управление на знания (KM) и управление на множество агенти (MA) в основния архитектурен проект. Сътрудничеството между техниките за оптимизация описва ясната представа за съвместно откриване на базата на ученето, за да се удовлетвори изискванията на Хибридна Интелигентна IDPS.

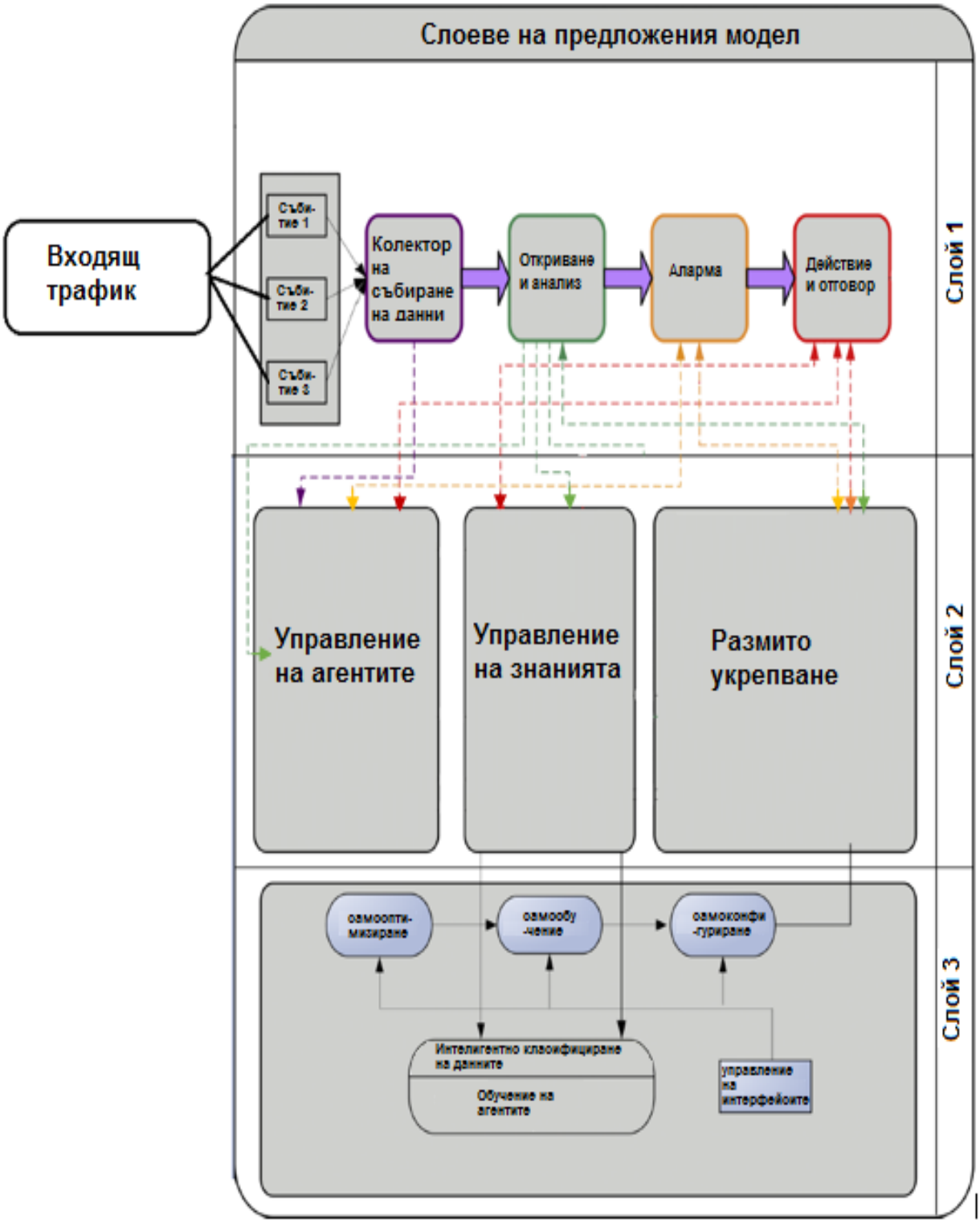
Концептуалният модел на системата за защита на компютърната мрежа се основава на три слоя. Първият слой показва традиционните системни компоненти, които наблюдават и събират данните от одита чрез сензорите, анализират данните и откриват проникванията, генерират аларми и обявяват правилната реакция чрез задействащите устройства. Следващите два слоя са разширени слоеве, базирани на предложената Хибридна Интелигентна IDPS.

Вторият слой показва традиционните методи за изкуствен интелект, които взаимно си общуват за по-точни резултати. Управлението на знанията дава възможност да се охарактеризират знанията за профила на аномалиите като набор от свързани понятия в домейна на аномалия. Карти с политики на мулти-агент мениджъра се използва за прогнозиране на аномалното поведение. Те се комбинират с адаптивните техники за оптимизация като размита логика и подсилване, за да се открият проникванията и да се захранват получените резултати в компонентите, включващи самооптимизатор, самообучение и самоконфигурация (трети слой). Последните компоненти на слоя се дефинират в принципите на автономното изчисление в реално време без човешка намеса. Това премахва човешкия фактор като причина за грешки. Стрелките посочват информационния поток между компонентите, докато стрелките отдясно показват логическата комуникация между компонентите (фиг. 6-7).

Разумно укрепване (Управление на обучението):

Управлението на Fuzzy Armor Learning (FRL) на тази архитектура на Хибридна Интелигентна IDPS включва обучение за укрепване (RL) и размити множества (FS). Като се има предвид случай на аномалия, FRL се анализира вътрешно и актуализира Q-

стойността на агента на обучаемия. Ако е необходимо, той автоматично актуализира новооткрития случай на нахлуване чрез прилагане на техники за изчислителна интелигентност и управление на знания, базирани на аномалии, в рекурсивна итерация на неговия цикъл на изпълнение.



Фиг. 6-7 Хибриден модел на система за откриване на проникванията

Управление на знанието:
За да се споделят знанията и да се даде възможност за сътрудничество между други мениджъри (т.е. учещи за размито укрепване и мениджъри на множество агенти),

мениджърът на знанието (KM) използва четири типа механизъм за вземане на решения: политика, онтология, профил на аномалиите и базирани на знания. Компонентът, базиран на знания, се свързва директно с Хибридна Интелигентна IDPS, за да съхрани процеса на обучение и тестване на алгоритми за CI. Целта на онтологията за вземане на решения (DO) е да предостави основа за представяне, моделиране на аномалии и изследване на решението за идентифициране на ненормално поведение. Правилата действат като селектор на действия и използват изпълнителен агент. Политиката за действие на механизма KM се адаптира към FRL, за да групира инцидентите според тежестта и да предизвика тревога.

Управление на няколко агента:

Многоредовият мениджър дава приоритет на аномалия според уязвимостта на жертвата. Има три възможни сценария в това състояние. Първият случай е събирането на шаблони чрез агента за събиране на данни (DC). Във втория случай, ако инцидентът е тежко нахлуване или ниско нахлуване, анализаторът (AA) и решаващият агент (DA) са асоциирани към базата от онтологии и знания, която включва техниките за изчислителни разузнавания за идентифициране на проникване. Третата възможна ситуация е свързана с изпълнителния агент (EA), който е защитен от нахлувания преди да настъпи загуба или повреда на данните. Тя дава тласък на системата да се самообучи от всякакви атаки, а едновременно с това осигурява защита и възможности за предотвратяване надолу по веригата в автономния режим на работа.

Самооптимизиране, самообучение и самообучение:

В случай на избор на действие, проследяващите агенти в Хибридна Интелигентна IDPS активират компонента за самооптимизиране, за да гарантират, че системата се защитава. В друга ситуация се отнася до блокиране на проникването преди да настъпи загуба на данни. Тук системата автоматично влиза в състояние на самообучение. И при двата случая, самооптимизиращото се състояние се задейства непосредствено след управлението на знанията и самообучаването, след като управлението на обучението за размита укрепване (Fuzzy Armor Learning) за защита на системата чрез учебно укрепване. Този метод се задейства, за да се защити системата чрез актуализиране на Хибридна Интелигентна IDPS като цяло. После се изпращат сигнали за активиране на изпълнителните механизми за изпълнение на превенцията в мрежова среда.

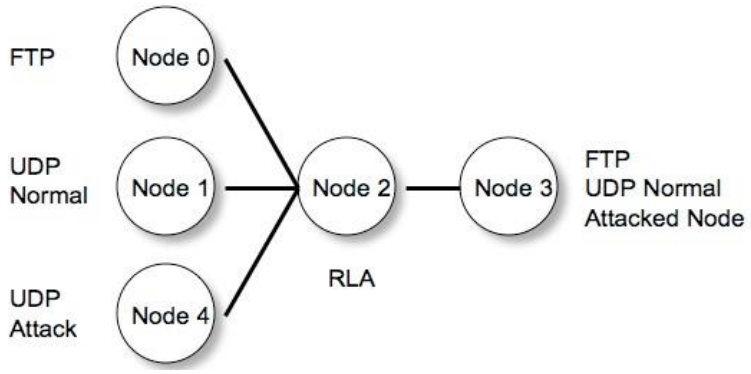
Разработени са сценарии за тестване, с цел верификация и оценка на ефективността на архитектурата.

Времето за предварителна обработка включва времето, прекарано в извличането и нормализирането на характеристиките. Времето за обучение зависи от броя пъти, в които класификаторът се нуждае от обучение, което от своя страна зависи от средната квадратична грешка между повторенията, достигащи целта минимум. Времето за тестване включва времето, прекарано в тестването на небелязаните случаи с претеглена средна стойност. Таблица 5.3 показва сравнението на ефективността на FQL по отношение на времето на консумация, получено по време на експериментите. От таблица 5.3 може да се види, че времето за обучение на FQL е подобно на стандартен Q-Learning, но изразходва повече време за тестване, отколкото FLC, Q-learning и FQL.

Таблица 6-4 Време на тестване и тренировка

Данни	Алгоритъм	Време за обучение	Време за тестване
Реални данни	**Стандартен MAC**	3.10	1.31
	Fuzzy Logic Controller	3.10	1.30
	Q-learning	3.14	1.36
	Fuzzy Q-learning	3.16	1.40

Времето за тестване на предложения метод е малко високо поради комбинацията от различни методи, като например размита Q-learning и алгоритъм за споделяне на стратегии за тегло, но в модифицирания метод е постигната по-голяма точност на откриване. Ускорението може да се подобри, когато хибридният класификатор се изпълнява в паралелни процесори. По този начин всички модули могат да бъдат обработвани паралелно от различните ядра, за да се намали значително общото време за обработка.
Проведени са поредица от тестове, за да установи дали използването заедно на мулти-агенти с RL може да позволи на агентите да се научат да категоризират нормалната и необичайната активност в мрежата, като използват информация за потока. Симулаторът на мрежата е конфигуриран да използва проста мрежова топология от 4 възли, както е показано на Фигура 8.2. Възловата точка 0 генерира нормален трафик по FTP, докато възел 1 генерира нормално UDP трафик. Възел 4 е атакуващ, създаващ поток от UDP трафик. Възел 2 е учебният агент за усилване (RLA), който трябва да се научи да разграничава нормалния трафик от възел 0 и 1 от ненормалното поведение на възел 4. Възел 3 е възелът под атака и получава валидни данни от възли 0 и 1 и информация от себе си. Параметрите на възлите, зададени от това изпитване, са показани в Таблица 8.2.



Фиг. 6-8 Примерна мрежова топология от четири мрежови възела

Таблица 6-5 Параметри на възлите

Възел	Протокол	Порт	Големина на пакета	Скорост
Възел 0	6 TCP	21 FTP	variable	variable
Възел 1	17 UDP	5080	512 Bytes	16 kbps
Възел 4	17 UDP	1580	110 Bytes	1 Mbps

За да се обучат RLA агентите, бяха генерирани нормални потоци от трафик от възли 0 и 1. За да се симулира истинска мрежа, потоците бяха спирани и пускани в случайни моменти. Активната атака е емулирана от възел 4 в определени часове по време на обучението. След като агентът се е научил, бяха извършени разнообразни тестове при различни условия на трафика, различна информация за потока и използване на различен брой функции за анализ. Прилагахме алгоритъма за обучение на тестовете с помощта на 2 и 3 функции. Както е показано в Таблица 6-3, беше използвана цифровата стойност за характеристика на протокола и порта като функции 1 и 2. Размерът на пакета е функция номер 3.
За първия тест бяха добавени още източници на FTP, UDP без атака и UDP атака, комбинирани с промени в изпращащите модели на приложенията и нападателите. Този тест е обозначен като Трафик Модел. Използвайки две и три характеристики на потока, RLA нямаше проблем при идентифицирането и категоризирането на потоците, както е показано в Таблица 6-3. Въпреки че агентът беше обучен само с потоците, генерирани от

възли 1, 2 и 4, той успя да разпознае новите потоци, генерирани от новите източници на трафик.
За да се симулира как в един реален свят нападател ще се опита да заобиколи защитните стени, като промени информацията в атаките си, беше създаден нов тест, в който е сменен порта, използван от атакуващия. Нарекохме този тест "Атака". За този тест порта беше сменен за атаки от оригиналния 5080 на 6665. Както и в предишния тест, RLA нямаше проблем да разпознае атаката с три функции. И все пак, използвайки само две функции, агентът не успя да разпознае анормалната активност. При по-сложен тест за атака, беше опитано да се симулира как атакуващият би се опитал да скрие атаката си, като използва същия протокол и порт на валидно приложение, идентифицирахме този тест Mimic атака. Подходът с две характеристики на трафика отново не беше в състояние да направи разлика между атаката и не-атакуващите потоци. Използването на три функции обаче позволи на RLA да идентифицира нападението, в този случай по целия размер на пакета. Това повдига въпроса: Какво се случва, ако нападателят има способността да модифицира приложението, за да използва същия размер, протокол и порт на пакета като валидно приложение?

Таблица 6-6 Цифрова стойност за характеристика на потока

Номер на характеристиката	Характеристика
2	Номер на протокола и порта
3	Номер на протокола, порта и големина на пакета

Таблица 6-7 Тестови резултати на Хибридния модел само в HBMS

Тест	Характеристика на потока	Прецизност	Отговор
Трафик Модел	2	100%	100%
Трафик Модел	3	100%	100%
тест "Атака"	2	0%	0%
тест "Атака"	3	100%	100%
Mimic атака	2	0%	0%
Mimic атака	3	100%	100%
Max Mimic атака	2	0%	0%
Max Mimic атака	3	0%	0%

За да се отговори на този въпрос, бяха разработени тестовете, посочени като Max Mimic атака. Подобно на предишния тест, беше установено, че номерът на протокола и портовете е същият като валидно приложение. Освен това, симулирайки сложна атака, като променим размера на пакета на трафика на нападателя. В този случай и двата подхода и двата типа характеристики не са в състояние да идентифицират потоците на нападателя. Важно е да се отбележи, че това не означава, че информацията за потока не е полезна за откриване на атаки или че RL не е подходящо решение. Той просто илюстрира, че като всеки друг подход, тези техники може да не са сто процента надеждни за всякакви сценарии. Като използваме последния тест като пример, можем да видим, че агентът няма начин да разграничи нормалните и атакуващите потоци, защото всички функции са еднакви. За да се открият тези скрити потоци, е необходимо да се изследват по-сложни откривателни изпълнения, евентуално използващи повече от един източник на информация.

Ето защо с прилагането на Хибридния модел и в двете софтуерни рами HBMS и NGMS резултатите са очаквано по-добри (Таблица 6-5).

Таблица 6-8 Хибридния модел и в двете софтуерни рами HBMS и NGMS

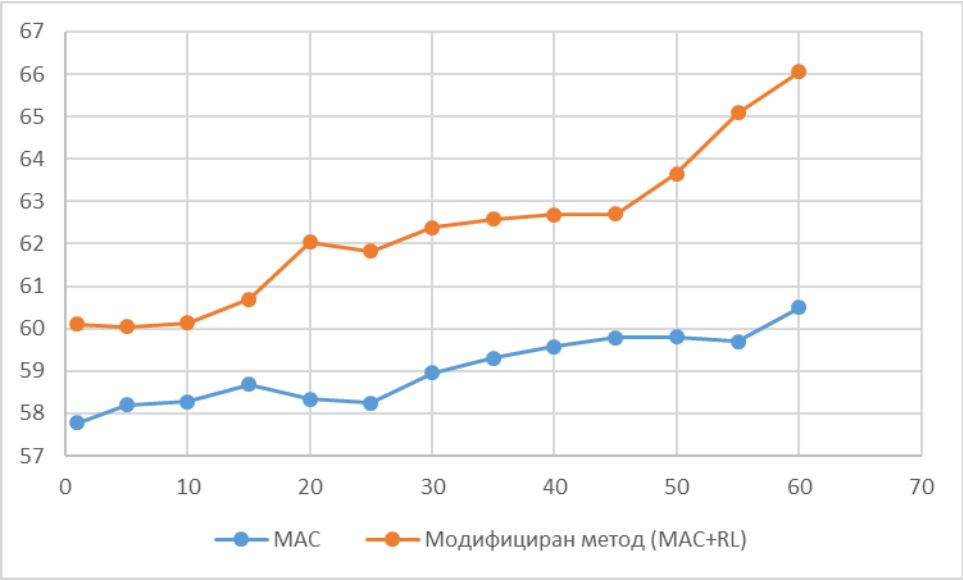
Тест	Прецизност	Отговор	Точност	FP	FN
Трафик Модел	100%	99%	99%	0%	3%
тест "Атака"	89%	97%	93%	11%	3%
Mimic атака	96%	73%	86%	0%	20%
Max Mimic атака	100%	29%	66%	0%	40%
Множество UDP източници	100%	97%	93%	9%	4%
Множество FTP източници	91%	97%	93%	9%	4%
Множество UDP атакуващи източници	100%	99%	93%	0%	4%

Извършени са симулации за откриване на DDoS атаки и резултатите са обобщени в таблица 6-6.

Таблица 6-9 Резултати от симулации за откриване на DDoS атаки

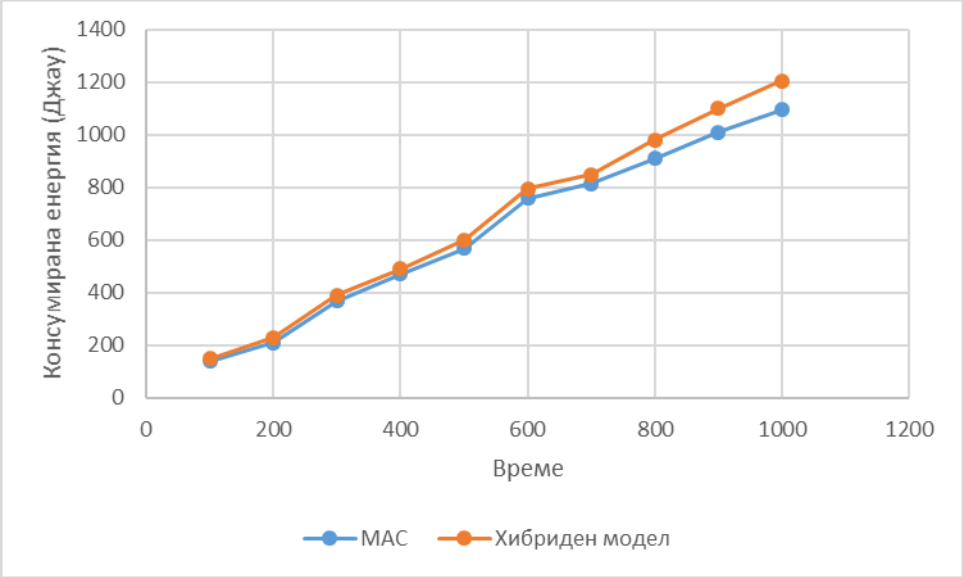
Атака (% от трафика)	MAC				Модифициран метод (MAC+RL)			
	SP %	FP %	FN %	OA	SP %	FP %	FN %	OA
1	80.10	1.20	1.10	57.78	83.20	1.20	1.10	60.10
5	81.20	1.40	1.30	58.20	83.40	1.30	1.20	60.05
10	82.50	1.90	1.70	58.28	84.30	1.50	1.60	60.13
15	83.70	2.10	2.00	58.68	85.60	1.70	1.80	60.70
20	83.90	2.40	2.20	58.33	87.90	1.90	2.00	62.03
25	84.20	2.60	2.30	58.25	88.30	2.10	2.30	61.83
30	85.80	2.80	2.60	58.95	89.70	2.40	2.50	62.38
35	86.40	2.90	2.70	59.30	90.50	2.60	2.70	62.58
40	87.70	3.20	3.00	59.58	91.70	3.10	3.00	62.68
45	88.50	3.40	3.20	59.78	92.40	3.20	3.40	62.70
50	89.60	3.90	3.50	59.80	94.20	3.30	3.70	63.65
55	90.40	4.10	4.00	59.70	96.50	3.50	3.80	65.08
60	92.40	4.50	4.30	60.50	98.20	3.70	3.90	66.05
Средна стойност	85.88	2.80	2.60	59.01	89.68	2.42	2.54	62.30

Очевидно е, че модифицирания механизъм постига най-голямото увеличение на точността на откриване. От фигура 6-9 може също така да се заключи, че точността на откриване на процент от атаката е по-висока, отколкото другите методи.



Фиг. 6-9 Сравнение на стойностите на точността на откриване

В този експеримент също така се изследва енергията, консумирана от предложените алгоритми по време на DDoS атаки на сензорни възли. Фигура 6-10 осигурява сравнението между споменатите алгоритми по отношение на общата консумирана енергия от сензорните възли.



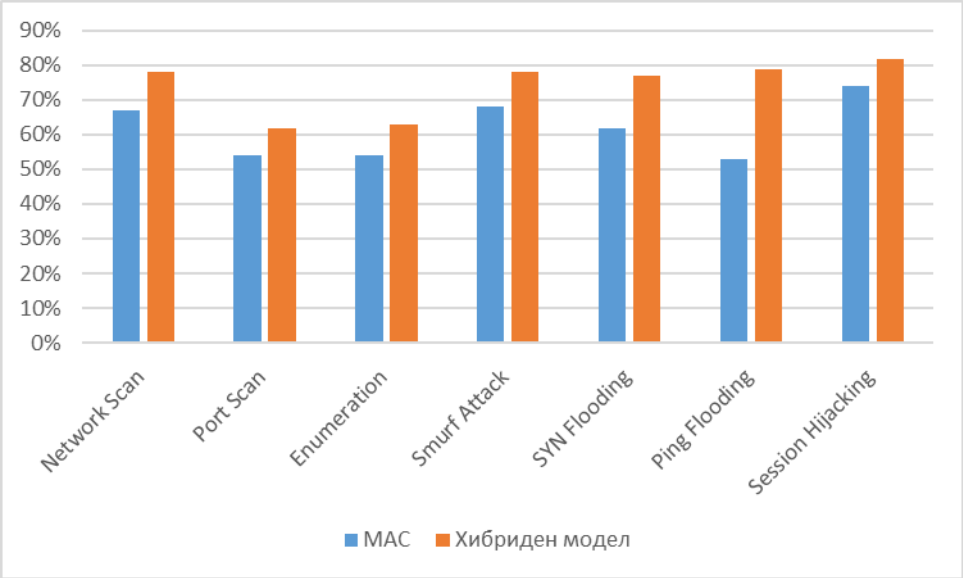
Фиг. 6-10 Консумация на енергия

Разходите за консумация на енергия могат да бъдат значителни, ако броят на съдействащите агенти или резултатът от отчитането е много голям. Както може да се види консумацията на енергия на модифицирания модел е по-голяма.
В таблица 6-7 е дадена статистика на откритите проникванията и атаките към системата.

Таблица 6-10 Статистика на откритите проникванията и атаките към системата

Име на атаката	Програма	Получени пакети	Открити атаки	Време (sec)	Процент
Network Scan	Angry IP, HPING2, Ping Sweeps	3000	600	20.25	78%

Port Scan	Nmap, NetScan Tools Pro, IPScanner	3300	700	22.3	62%
Enumeration	SuperScan 4, enum, PsGetSid	2000	500	10	63%
Smurf Attack	Land and Latierra	500	400	15.5	78%
SYN Flooding	NemeSys	2000	2000	25.4	62%
Ping Flooding	Crazy Pinger	2000	400	25	79%
Session Hijacking	RST Hijacking Tools, Remote TCP Session Reset Utility	900	900	30	74%



Фиг. 6-11 Сравнение на двата предложени модела

Успешно са проведени експерименти, с цел верификация и оценка на отделните компоненти и на цялата платформата. Тези експерименти удостоверяват, че системата отговаря на изисквания на спецификациите.

ПРИНОСИ НА ДИСЕРТАЦИОННИЯ ТРУД

НАУЧНО-ПРИЛОЖНИ ПРИНОСИ:

1. Въз основа на изчерпателен анализ на съвременните кибер-заплахи и приложението на методите на изкуствения интелект е формулирана съвкупност от критерии за избор на подходящ метод при системите за откриване и превенция на проникване.
2. Предложен е нов, по-систематичен и ефективен модел за откриване и противодействие на DoS- и DDoS-атаки от IDPS системи на базата на мулти-агентни технологии, който има по-добра мащабируемост, гъвкавост и осигурява възможности за защита в реално време.
3. Доразвит е предложения модел с хибриден подход чрез метода на обучение с утвърждаване и използване на размита логика и е доказана експериментално ефективността на надграждането с него на мулти-агентната система, при което се реализира значително подобрение на „успеваемостта“ на кибер-защитата.
4. Експериментално е доказано предимството на прилагане, при превантивна защита срещу „zero-day“ атаки и зловреден код, на поведенчески техники за анализ, които се фокусират върху поведението, а не върху самите заплахи.
5. Доказано е експериментално, че технологията на самообучение и обучение с утвърждаване и използване на размита логика, е в състояние да защити критични обекти вътре в операционната система без изграждане, съхраняване и актуализиране на база данни със сигнатури.

ПРИЛОЖНИ ПРИНОСИ:

6. Програмно е реализиран модел, базиран на две отделни мулти-агент-базирани софтуерни рамки за защита на мрежов сървър и хостовете в мрежата, с използване на мобилни агенти в системата.
7. Разработени са тестови сценарии за оценка на ефективността на мулти-агент-базирана система за откриване на проникване и превенция, които ясно и еднозначно потвърждават постигнатата ефективност на предложения модел.
8. Разработени са различни алгоритми, които симулират ефективността на хибриден модел и са изградени различни сценарии за симулация.

СПИСЪК НА ПУБЛИКАЦИИТЕ ПО ДИСЕРТАЦИОННИЯ ТРУД

1. R. Trifonov, G. Tsochev, R. Yoshinov, S. Manolov, G. Pavlova, Conceptual model for cyber intelligence network security system, International Journal of Computers, Volume 11, 2017, ISSN: 1998-4308, pp. 85-92

2. Georgi Tsochev, An Investigation of Multi-Agent System Model for Intrusion Detection/Prevention, International Journal of Advanced Research in Computer and Communication Engineering, Vol. 6, Issue 9, September 2017, ISSN (Online) 2278-1021, pp 1-6

3. G. Tsochev, Roumen Trifonov, G. Popov, FIPA Standardization for Agent Based Technology, Computer&Communications Engineering V.10/1, 2016, ISSN 1314-2291, pp 32-34

4. G. Tsochev, R. Trifonov, G. Popov, A security model based on multi agent systems, participation in XXIX International Conference on Information Technologies InfoTech-2016, St. St. Constantin and Elena resort, Bulgaria 2016, ISSN: 1314-1023, pp. 146-153

5. G. Tsochev, R. Trifonov, R. Yoshinov, Multi-agent framework for intelligent networks, 29th Int. Conference on Information Technologies (InfoTech-2015), September 2015 Varna – St. St. Const. and El., Bulgaria, ISSN: 1314-1023, pp. 109-116

6. G. Tsochev, R. Trifonov, G. Naydenov, Agent communication languages comparison: FIPA-ACL and KQML, Int. Scientific Conference COMPUTER SCIENCE'2015, September 2015, Durrës, Albania, ISBN: 978-619-167-177-9, pp.268-273

7. Trifonov R., S. Manolov, G. Tsochev, Application of multi-agent systems for network and information protection, 28th International Conference on Information Technologies (InfoTech-2014), 18-19 September 2014 Varna – St. St. Constantine and Elena resort, Bulgaria, ISSN: 1314-1023, pp 137-142

SUMMARY

STUDY OF METHODS OF ARTIFICIAL INTELLIGENCE FOR APPLICATION IN COMPUTER NETWORK SECURITY

MSc. eng. Georgi Tsochev

It is very important for an organization to develop security mechanisms to prevent unauthorized access to system resources and confidential company data. There are many ways to protect an enterprise's network infrastructure, including penetration detection and prevention systems. Over the last two decades, the development of computer networks has been a field for innovation improvements. There are many "Next Generation" varieties available on the market that provide a fairly good set of tools for interacting and preventing network attacks.

To build a security system that has the ability to protect the network (servers and hosts) against attacks and malicious software, and at the same time to be one step ahead of hackers' creativity and threats to zero-day attacks , but not yet corrected by manufacturers with appropriate software updates), a number of implementation challenges need to be addressed as follows:

1) detection of zero-day threats, without prior knowledge of their signatures or features, with low level of false signals.

2) The system must work in unison with no chance of failure because it can seriously affect network performance, and some attempts at threats may remain undetected.

3) Active blocking of malicious traffic in the network and on system objects, while allowing normal traffic and objects to pass freely, and weaken the occurrence of false positive and false negative results.

4) High performance and real-time protection without network latency or overload.

5) Avoiding the huge amount of signatures and patterns of attacks used to identify malware and software, as well as identifying zero-day threats.

This thesis explores artificial intelligence methods for application in computer network security to enhance the reliability and capability of preventive detection and detection of attacks on network computing.

Fuzzy logic, validation training, and multi-agent technologies from artificial intelligence to create a detection and detection system were used to successfully accomplish the goal.

In connection with the main objective, the following tasks were carried out:

1) Two software frameworks were built: a host based monitoring system and a network gateway monitoring system

2) Multi-agent technology was used as a paradigm for designing the software frameworks of the system as well as applying the strategy to protect the defense in depth within the protection mechanisms

3) Different IDPS technologies have been integrated into one platform to exploit the benefits of different integrated technologies

4) Behavioral analysis and detection techniques are applied that focus on the behavior of the guarded object, not the behavior of the threat or the hacker

5) The system includes three main concepts for discovery management: Fuzzy Strength Training, Knowledge Management (KM), and Multiple Agent Management (MA) in the Basic Architecture Project

6) Test scenarios have been developed to evaluate the effectiveness of a multi-agent-based intrusion detection and prevention system that clearly and unambiguously confirms the effectiveness of the proposed model.

7) Different algorithms have been developed that simulate the efficiency of a hybrid model and have constructed various simulation scenarios.

The proposed system protects the network server, Internet access point (separate server) and individual hosts against attacks and malware without relying on any database of ready-made signature databases. The proposed system works in the Microsoft Windows environment and protects the critical operating system objects. It also protects network-level protocols, especially TCP, ICMP, and UDP.