

УДК 004.49

DOI: 10.31891/2219-9365-2019-64-14

САВЕНКО О. С., ПАЮК В. П., САВЕНКО Б. О., КАШТАЛЬЯН А. С.
Хмельницький національний університет

МОДЕЛІ НЕЗАДОКУМЕНТОВАНИХ ЗАКЛАДОК ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ В ЛОКАЛЬНИХ КОМП'ЮТЕРНИХ МЕРЕЖАХ

В роботі здійснено постановку актуальної наукової задачі з виявлення в програмному забезпеченні незадокументованих закладок, які можуть бути самостійними об'єктами або частиною певного зловмисного програмного забезпечення. Місцем дослідження вибрано локальні комп'ютерні мережі.

Для незадокументованих закладок програмного забезпечення було проаналізовано види загроз, які можуть бути здійснені ними в локальній мережі, та здійснено їх формалізацію та деталізацію. Така формалізація представлена в моделі зловмисного програмного забезпечення в локальній комп'ютерній мережі частковим випадком. Це дозволило застосувати результати до незадокументованих закладок програмного забезпечення, які є частинами певного зловмисного програмного забезпечення, отримані для нього шляхом використання розподіленої багаторівневої системи виявлення. Для такого застосування були розроблені моделі незадокументованих закладок програмного забезпечення, яке використовується в локальних комп'ютерних мережах.

Моделі дозволили після відповідної формалізації включати їх в засоби виявлення. Застосування розроблених моделей незадокументованих закладок програмного забезпечення в розподіленій багаторівневій системі виявлення дало можливість покращити ефективність виявлення бот-мереж, складовими яких вони були. Підтвердженням результатів покращення виявлення був проведений протягом тривалого часу експеримент з виявлення бот-мереж, в складі кожної з яких були незадокументовані закладки програмного забезпечення.

Ключові слова: незадокументовані закладки, програмне забезпечення, модель, зловмисне програмне забезпечення, локальна комп'ютерна мережа

SAVENKO O., PAIUK V., SAVENKO B., KASHTALIAN A.
Khmelnitskyi National University

MODELS OF SECRET CODES OF SOFTWARE IN LOCAL COMPUTER NETWORKS

The paper deals with the actual scientific task of detecting secret bookmarks in the software, which may be separate objects or part of certain malicious software. Local computer networks are selected as the study site.

For secret software bookmarks, the types of threats that can be committed to the LAN were analyzed and formalized and detailed. This formalization is a partial case of malware on LANs. This allowed the results to be applied to secret software bookmarks that are part of certain malicious software obtained for it by using a distributed multilevel detection system. Models of secret bookmarking software for use on local area networks have been developed for this purpose.

The models were allowed, after appropriate formalization, to include them in detection tools. The use of developed models of secret software bookmarks in a distributed multilevel detection system made it possible to improve the detection efficiency of the botnets they were part of. Confirmation of the results of the improvements was carried out for a long time an experiment to identify botnets, each of which were secret software bookmarks.

Keywords: secret software bookmarks, software, model, malicious software, local computer network

Вступ. Постановка задачі. Розвиток інформаційних технологій в різних сферах продовжує супроводжуватись невинним бажанням зловмисників отримати вигоду за рахунок недоліків в їх захисті. Найбільш актуальними для отримання вигоди з погляду зловмисників є організації та підприємства, в яких функціонують інформаційні технології. Відомо багато способів проникнення в локальні комп'ютерні мережі підприємств (організацій) з метою несанкціонованого доступу до інформації в них. Одним із способів доступу зловмисників до інформаційних ресурсів підприємств (організацій) є використання незадокументованих можливостей в програмному та апаратному забезпеченні персональних комп'ютерів і периферійному обладнанні, які дозволяють здійснювати прихований несанкціонований доступ до ресурсів системи, як правило, за допомогою локальної мережі. Основне призначення незадокументованих програмних закладок - забезпечити несанкціонований доступ до конфіденційної інформації.

Програмна закладка - потай впроваджена програма, яка створює загрозу для інформації, що міститься у комп'ютері [1]. Програмна закладка може бути реалізована у вигляді зловмисної програми чи зловмисного програмного забезпечення або незадокументованого програмного коду в програмному забезпеченні.

В якості об'єкту дослідження розглядатимемо незадокументовані закладки програмного забезпечення, яке використовується в локальних комп'ютерних мережах підприємств (організацій).

Складність виявлення такого таємно внесеного в програмне забезпечення функціонального об'єкту, який за певних умов здатний забезпечити несанкціонований програмний вплив, пов'язана з можливістю відсутності його прояву протягом тривалого часу. Такий об'єкт може бути частиною програмного комплексу, який виконує поставлені завдання, замінювати повністю певні частини програмного комплексу,

замінювати певну потрібну програму. Як правило, такі незадокументовані закладки програмного забезпечення дозволяють зберігати заявлені виробником функції програмного забезпечення і реалізуються частиною функцій, які входять до програмного комплексу.

Підприємство може використовувати готове програмне забезпечення, в якому вже присутні незадокументовані закладки, або зроблене під замовлення, в якому була здійснена неякісно його верифікація при прийнятті в експлуатацію.

Програмне забезпечення, яке експлуатується в локальних мережах підприємств, як правило, є розподіленим і тоді незадокументовані закладки програмного забезпечення є активними в усіх комп'ютерах мережі. Це підвищує загрози для підприємств та організацій.

Незадокументовані закладки програмного забезпечення можуть приймати участь в створенні бот-мереж, втіленню троянських програм, тощо. Тому, актуальною продовжує залишатись проблема виявлення зловмисного програмного забезпечення, зокрема, і незадокументованих закладок програмного забезпечення.

Одним із завдань, які потребують вирішення, є розробка моделей незадокументованих закладок програмного забезпечення в локальних комп'ютерних мережах.

Попередні роботи. Незадокументована закладка програмного забезпечення може бути частиною захищеної системи. Тоді вона здатна маскувати свою присутність в комп'ютерній системі. При цьому в системі створюється прихований канал інформаційного обміну. Він, як правило, залишається непоміченим для адміністраторів системи протягом тривалого часу. Виявити незадокументовану закладку програмного забезпечення стандартними засобами адміністрування складно. Вона може функціонувати необмежено тривалий час. Таким чином, протягом цього часу зловмисник отримує необмежений доступ до системних ресурсів.

Оскільки всі події відбуваються в локальних комп'ютерних мережах, то потрібна розподілена система, яка здійснюватиме тривале спостереження та аналіз подій на предмет виявлення незадокументованих закладок програмного забезпечення.

Розроблені в [2]-[5] розподілені системи дозволяють здійснювати виявлення зловмисного програмного забезпечення (ЗПЗ), але зорієнтовані переважно на прояви більшої інтенсивності, ніж прояви, які отримуються від незадокументованих закладок програмного забезпечення (НЗПЗ). Крім того, НЗПЗ мають певну специфіку, яка фактично є лише частиною складного ЗПЗ. Також, засобами виявлення таких НЗПЗ можуть бути спеціального виду приманки [6], які розміщуються в мережі для зловмисника і виступають помилковими об'єктами атаки. Вони дозволяють зловмиснику проявити свої дії на них. І це відбувається швидше, ніж для реальних об'єктів атаки. Але всі розглянуті методи та засоби необхідно відповідним чином скомбінувати, узгодити та налаштувати. Крім того, методи та засоби виявлення мають враховувати наявні моделі НЗПЗ та ті, які з'являтимуться в майбутньому. Тому, виявлення НЗПЗ потребує доповнення розподілених систем виявлення новими методами, розробленими на основі актуальних моделей НЗПЗ.

Метою роботи є розробка нових моделей НЗПЗ та їх використання в РБС [5] виявлення ЗПЗ в ЛКМ для покращення ефективності виявлення.

Основна частина. Розглянемо види загроз від НЗПЗ, які можуть бути здійснені в локальній мережі. Їх аналіз пов'язаний з вимогами, які висуваються до безпеки комп'ютерних систем в мережі: конфіденційність, цілісність, доступність та аутентичність. Для порушення цих вимог розробники НЗПЗ закладають в нього механізми здійснення загроз у вигляді таких атак: переривання, перехоплення, зміна, підrobка. В локальних мережах здійснення таких атак або їх комбінацій відбувається по відношенню до апаратного забезпечення, програмного забезпечення, ліній зв'язку та даних. На рис. 1 зображено об'єкти в комп'ютерних системах локальних мереж, які можуть бути піддані атакам за певним типами загроз.

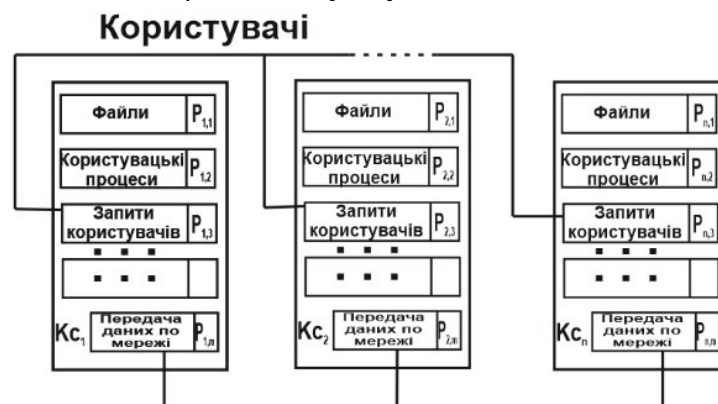


Рис. 1 Об'єкти комп'ютерних систем, які можуть бути піддані атакам

Позначення:
 $i = 1, n$;

- $P_{i,1}$ – множина файлів i – ої комп'ютерної системи;
 $P_{i,2}$ – множина користувацьких процесів i – ої комп'ютерної системи;
 $P_{i,3}$ – множина запитів користувачів i – ої комп'ютерної системи;
 $P_{i,4}$ – множина мережних пакетів i – ої комп'ютерної системи.

Функціонування комп'ютерних систем в локальних мережах пов'язане з обробкою, зберіганням та поширенням інформації. Саме при виконанні цих дій можливим є здійснення атак, узагальнені види яких виділимо наступним чином:

- 1) $Z_{i,1}$ – множина несанкціонованих змін;
- 2) $Z_{i,2}$ – множина підроблених об'єктів, розміщених в систему в результаті атаки;
- 3) $Z_{i,3}$ – множина перехоплень зі сторони зломисника засобами програм або комп'ютерів;
- 4) $Z_{i,4}$ – множина переривань, яка здійснена для виведення з ладу компонентів системи.

Розглянемо детальніше можливі події при проведенні атак. Зокрема, елемент множини $P_{i,1}$ може бути скопійований чи перенесений в інше місце пам'яті. Тоді, можливі два випадки: ця подія відбулась успішно, ця подія не відбулась (або через помилки, пов'язані з роботою операційних систем чи компонентів комп'ютерної системи; або в результаті проведеної атаки). Задамо можливі події за формулою (1):

$$\begin{aligned} p_{f,1} &\xrightarrow{f} p_{f,1} \\ p_{f,1} &\xrightarrow{z} p_{f,1,z} \end{aligned} \quad (1)$$

Задамо матрицями категорії атак [7]. Введемо такі вершини матриці: джерело інформації, отримувач інформації, подія переривання, подія перехоплення, подія зміни, підробка. Зв'язки між цими вершинами зобразимо напрямленими дугами. Зображення графу, що відповідає таким подіям, на рис. 2.

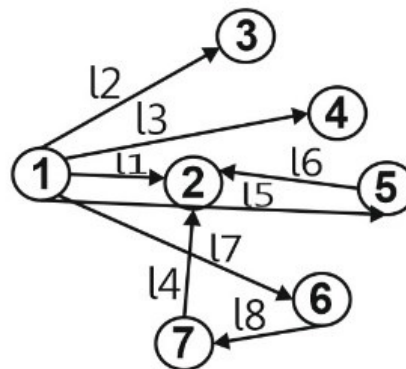


Рис. 2 Граф видів загроз

Позначення:

- 1-джерело інформації;
 2-отримувач інформації;
 3-подія переривання;
 4-подія зміни інформації;
 5-вершина, в якій відбувається перехоплення інформації;
 6-отримання перехопленої інформації;
 7-подія підробки інформації;

l_1 – передавання інформації здійснено вірно;

l_2 – відбулось переривання передавання інформації і вона не дійшла до отримувача;

l_3 – передавання інформації перервано і здійснюється її зміна;

l_4 – змінена інформація надсилається отримувачу;

l_5 – передавання інформації перехоплено, при цьому вона далі передається отримувачу без спотворень;

l_6 – передавання перехопленої інформації далі отримувачу;
 l_7 – перехоплена інформація обробляється зловмисником;
 l_8 – інформація від джерела не надсилалась, але зловмисник надсилає певну підроблену інформацію до отримувача.

Матриця інцидентності, що відповідає графу з рис. 2 зображена табл. 1.

Таблиця 1

Матриця інцидентності видів загроз

	l_1	l_2	l_3	l_4	l_5	l_6	l_7	l_8
1	1	1	1	0	1	0	0	0
2	-1	0	0	-1	0	-1	0	-1
3	0	-1	0	0	0	0	0	0
4	0	0	-1	1	0	0	0	0
5	0	0	0	0	-1	1	1	0
6	0	0	0	0	0	0	-1	0
7	0	0	0	0	0	0	0	1

Види загроз залежать від особливостей комп'ютерних систем та їх компонентів. Виділимо компоненти, для яких встановимо їх взаємозв'язок з видами загроз: апаратне забезпечення, програмне забезпечення, данні, засоби організації зв'язку. Апаратне забезпечення через його доступність після збоїв викликаних втручанням може відмовляти в обслуговуванні. Програмне забезпечення може мати такі види загроз: відмова користувачам в доступі, несанкціоноване копіювання, зміна функціоналу програми. Данні, які зберігаються, можуть бути видалені через видалення файлів, відмовити в доступі до них користувачам, несанкціоновано прочитані, змінені їх вміст. Засоби організації зв'язку можуть мати такі загрози, при здійсненні яких дозволять читання повідомлень, спостереження за трафіком, зміну вмісту, зміну часу доставки, порядку доставки повідомлень або їх дублювання, підробка повідомлень, видалення повідомлень. Таким чином, здійснено виділення типових загроз у локальних мережах та запропоновано їх формалізоване представлення, використання якого є важливим при створення розподілених систем виявлення зловмисного програмного забезпечення.

Позначимо множину всього ЗПЗ через V , яке перебуває в комп'ютерах локальних мережах. Тобто розглядатимемо те ЗПЗ, яке за певних обставин та на протязі певного часу експлуатації локальних комп'ютерних мереж, проникло в комп'ютерні системи, змогло пройти певні системи захисту і функціонує там. Серед такого ЗПЗ є, також, незадокументовані закладки програмного забезпечення окремими об'єктами або частинами інших об'єктів. Представимо ЗПЗ в локальних комп'ютерних мережах алгебраїчною системою типу $\tau = (\alpha, \beta)$:

$$\mathcal{A}_V = \langle V, \Omega_F, \Omega_P \rangle, \quad (2)$$

де $\Omega_F = \{F_0, F_1, F_2, \dots, F_{\alpha_1}, \dots\}$ – множина операцій заданих на множині V для кожного $\alpha_1 = 0, 1, 2, \dots$; $\Omega_P = \{P_0, P_1, P_2, \dots, P_{\beta_1}, \dots\}$ – множина предикатів заданих на множині V для кожного $\beta_1 = 0, 1, 2, \dots$; $\alpha = 1, \beta = 1$ –

файлами, оперативною пам'яттю та командами роботи в мережі: створення, відкриття, закриття, видалення, читання, записування, додавання, знаходження, отримання атрибутів і встановлення атрибутів, команди доступу до ОП, команди для роботи в мережі. Реалізація характеристичних властивостей ЗПЗ пов'язана з системними викликами та командами для роботи в мережі визначатиме наповнення функцій з множини Ω_F і залежатиме від них, що дозволить ідентифікувати такі дії.

На основі видів загроз і моделі ЗПЗ задамо для НЗПЗ моделі, які потрібні для систем їх виявлення.

Впровадження незадокументованих програмних закладок на різних етапах життєвого циклу програмного забезпечення може відбуватись так:

- 1) робота зловмисників в складі розробників програмних засобів;
- 2) створення команд, які динамічно формуються, або паралельних обчислювальних процесів;
- 3) здійснення переадресації команд та запис зловмисної інформації в використовувані інформаційною системою або іншими програмами ділянки пам'яті;
- 4) внесення в програмний код НЗПЗ;
- 5) створення замаскованого пускового механізму НЗПЗ;
- 6) внесення НЗПЗ в окремі підпрограми і в керуючу програму;
- 7) підготовка тестових даних для виявлення НЗПЗ;
- 8) приховування НЗПЗ внесенням в програмний засіб помилок;
- 9) розміщення НЗПЗ в гілках програмного засобу, які не перевіряються при контролі;

- 10) участь зловмисників при здійсненні верифікації;
- 11) розробка НЗПЗ при доопрацюванні програмного засобу;
- 12) розробка оновлення та доповнення для НЗПЗ.

Подальше використання НЗПЗ може здійснюватись зловмисниками, які працюють безпосередньо на підприємстві особисто або через третіх осіб, або віддалено з використанням відповідних технічних засобів.

Моделі НЗПЗ в комп'ютерах локальних мереж:

1) модель «перехоплення», в якій НЗПЗ розміщується в програмне забезпечення, зберігає всі або вибрані фрагменти ПЗ, вводиться або виводиться в прихованій області локальної або віддаленої зовнішньої пам'яті прямого доступу; об'єктом збереження може бути клавіатурне введення, документи, що виводяться на принтер, або знищуються файли-документи; для цієї моделі потрібна наявність у зовнішній пам'яті місця зберігання інформації, яке має бути організоване таким чином, щоб забезпечити її збереження протягом заданого проміжку часу і можливість подальшого знімання та приховування від інших користувачів чи процесів;

2) модель «спостерігач», в якій НЗПЗ вбудовується в мережне або телекомунікаційне програмне забезпечення; дане програмне забезпечення, як правило, завжди активне, тому НЗПЗ здійснює контроль за процесами обробки інформації в комп'ютері, установку і видалення закладок, а також знімання накопиченої інформації; НЗПЗ може ініціювати події для раніше впроваджених закладок;

3) модель «компрометація», в якій НЗПЗ або передає задану зловмисником інформацію (наприклад, клавіатурний ввід) в канал зв'язку, або зберігає її, не покладаючись на гарантовану можливість подальшого прийому або зняття; НЗПЗ може, також, ініціювати постійне звернення до інформації, що приводить до зростання відносини сигнал / шум при перехопленні побічних випромінювань;

4) модель «спотворення або ініціатор помилок», в якій НЗПЗ спотворює потоки даних, що виникають при роботі прикладних програм (вихідні потоки), або спотворює вхідні потоки інформації, або ініціює (або пригнічує) виникають при роботі прикладних програм помилки;

5) модель «прибирання сміття», в якій НЗПЗ при здійсненні прямого впливу на програмний засіб може і не створити руйнівного результату; основною метою такого впливу є забезпечення максимізації утворених «залишків» інформації для подальшого вичення; зловмисник отримує або дані фрагменти, використовуючи закладки попередніх моделей, або безпосередній доступ до комп'ютера під виглядом ремонту або профілактики.

НЗПЗ в процесі свого функціонування створюватиме свої процеси в комп'ютерних системах та впливатиме на інформаційні потоки. Крім того, результати її функціонування можуть впливати на інші комп'ютери в локальній мережі, що може і повинно досліджуватись з метою виявлення НЗПЗ. Для цього потребуватимуть моніторингу інформаційні процеси, які відносяться до звернень до мережних та файлових ресурсів, а також, до паролів користувача.

Дослідження НЗПЗ професійними фахівцями з кібербезпеки дозволяє встановити їх наявність за такими ознаками: наявність модулів ПЗ, які не відповідають призначенню процесу; наявність об'єктів операційних систем, які відкриті процесом, що не відповідають призначенню процесу; висока інтенсивність операцій введення-виведення зі сторони певного процесу; великий відсоток завантаження процесора або внутрішньої пам'яті зі сторони певного процесу; подібність імені файла до імені файла, що відноситься до операційної системи; виконуваний файл процесу операційної системи розміщено не в загальноприйнятному каталозі; процес, який відноситься до операційної системи, виконується від імені локального користувача; система захисту від виконання коду в області даних, які ввімкнена для всіх процесів, для розглядуваного процесу відімкнена; для процесу, що відноситься до операційної системи, задіяно інших каталог, відмінний від того, що повинен бути для такого процесу; відсутній цифровий підпис у виконуваних файлах програмного засобу; велика мережна активність процесу, який повинен працювати локально; тощо. Але для покращення ефективності виявлення НЗПЗ потрібні засоби, які дозволять здійснювати встановлення факту наявності НЗПЗ без втручання адміністратора мережі, який може не опрацювати певні з ознак з різних причин. НЗПЗ можуть використовувати засоби маскування в системі, що ускладнює їх виявлення.

З технічної сторони при створенні ПЗ з наявними НЗПЗ використовують методи програмування, які не є поширеними при створенні типового програмного забезпечення, тому ці особливості можуть теж бути додатковими ознаками для їх виявлення. Зокрема, для виконуваного файлу операційного середовища Windows такими ознаками можуть бути такі: додаткова секція в кінці файла; точка входу вказує на перехід в середину секції, яка не є секцією коду; точка входу вказує на команду переходу, яка задає перехід за секцією коду; наявність ознак секції коду, яка не є секцією коду. Аналогічно для інших середовищ в комп'ютерній системі, інформація з яких може бути пов'язана з розміщенням в оперативній пам'яті.

Моделі НЗПЗ є основою для їх подальшої формалізації та використання в розподілених системах виявлення.

Експериментальні дослідження. Для проведення експериментів було використано розподілену багаторівневу систему виявлення ЗПЗ [5]. НЗПЗ було розроблено як складову кожної з типових бот-мереж. Тоді, метою експериментів була перевірка застосування методу виявлення бот-мереж, роботи класифікатора в структурі розподіленої системи та визначення залежності відсотку виявлених вузлів бот-мережі від їх представлення векторами, в складі яких були НЗПЗ. Для проведення експериментів було здійснено

конструювання 28 штучних бот-мереж та отриманих кодів відомих виявлених бот-мереж, згруповано їх за класами, виділено в них 25 структурних елементів в трьох стадіях функціонування і 81 функцію, причому не всі так отримані бот-мережі містили повністю всі структурні елементи та функції. Експеримент проводився для класифікатора без додавання екземплярів створених бот-мереж та з ними, тобто здійснювалась перевірка без навчання класифікатора на створених зразках і з попереднім віднесенням зразків по класах. Другий варіант є необхідними для перевірки точності віднесення до класів тих же зразків є в них введені, бо при здійсненні моніторингу API функцій можуть бути похибки. Тривалість моніторингу КС локальної мережі становила 350 годин для кожного екземпляру бот-мережі кожного з двох класифікаторів. Атака з вузлів бот-мережі не здійснювалась. Вузли бот-мережі працювали тільки в режимі контролю КС та підтримки структури бот-мережі через відправлені повідомлення. Таким чином, для компонент РБС об'єктами дослідження були запущені в КС процеси і відповідно побудова векторів по них. Для проведення експерименту було обрані бот-мережі, які використовують стратегію отримання повного контролю в КС за рахунок активації їх складових НЗПЗ. Для здійснення експерименту засобами API моніторингу в КС було отримано вектори, які по чергово оброблено класифікатором компоненти. Результати обробки представлено в табл. 2.

Експерименти передбачали визначення наступних показників ефективності виявлення вузлів бот-мереж для класів і підкласів класифікатора:

- 1) $P_{1,1}$ – відсоток векторів зловмисних дій та атак для вузлів бот-мереж, що належать даному класу відносно всіх тестових зразків, які система віднесла до цього класу з використанням попереднього навчання;
- 2) $P_{1,2}$ – аналогічно до п. 1, однак без використання попереднього навчання;
- 3) $P_{2,1}$ – відсоток векторів зловмисних дій та атак для вузлів бот-мереж, що належать даному підкласу класу відносно всіх тестових векторів, які система віднесла до цього підкласу класу в тестовій вибірці (ті, які були правильно віднесені до підкласів) з використанням попереднього навчання;
- 4) $P_{2,2}$ – аналогічно до п. 3, однак без використання попереднього навчання;
- 5) $P_{3,1}$ – відсоток правильно виявлених вузлів бот-мереж з використанням попереднього навчання;
- 6) $P_{3,2}$ – аналогічно до п. 5, однак без використання попереднього навчання;
- 7) $P_{4,1}$ – відсоток неправильно класифікованих вузлів бот-мереж як корисних додатків (помилка 1-го роду) з використанням попереднього навчання;
- 8) $P_{4,2}$ – аналогічно до п. 7, однак без використання попереднього навчання;
- 9) $P_{5,1}$ – відсоток неправильно класифікованих вузлів бот-мереж як таких, що є вузлами бот-мереж, але віднесені не до того класу (помилка 3-го роду), з використанням попереднього навчання;
- 10) $P_{5,2}$ – аналогічно до п. 9, однак без використання попереднього навчання.

Результати оцінки ефективності виявлення програмного забезпечення вузлів бот-мереж на основі роботи двох класифікаторів для введених класів та підкласів у класифікаторі наведено у табл. 2.

Таблиця 2

Результати експерименту								
Показники експерименту	Отримані значення для різних класів							Середні значення
	Class 0	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	
P _{1,1} , %	90,74	84,29	73,66	86,30	94,04	94,18	96,60	89,44
P _{1,2} , %	75,93	63,57	60,22	70,32	68,77	67,60	69,36	67,71
P _{2,1} , %	85,80	83,57	72,58	85,39	98,88	93,92	96,60	88,42
P _{2,2} , %	74,69	63,57	59,14	70,32	67,37	66,58	67,66	66,80
P _{3,1} , %	92,11	84,21	71,93	89,47	90,53	88,42	93,68	87,72
P _{3,2} , %	76,32	57,89	63,16	64,91	71,58	54,74	75,79	65,89
P _{4,1} , %	7,89	14,47	28,07	10,53	7,37	11,58	6,32	11,70
P _{4,2} , %	21,05	40,79	36,84	31,58	24,21	44,21	22,11	31,97
P _{5,1} , %	0	1,32	0	0	2,11	0	0	0,01
P _{5,2} , %	2,63	1,32	0	3,51	4,21	1,05	2,11	2,14

В результаті проведення експерименту отримано віднесення до потрібного підкласу та класу отриманих на основі моніторингу векторів з точністю до 66% для класифікатору без введених векторів 28 штучно згенерованих бот-мереж та 88% для класифікатору, в який попередньо було додано вектори шляхом здійснення його навчання, зберігаючи в ньому шаблони попередніх наповнень. Відсоток ознак, які були використані РБС для виявлення бот-мереж і пов'язані з проявами НЗПЗ, становить приблизно 27% від загальної кількості виявлених. Інтенсивність проявів від НЗПЗ суттєво нижче від типових проявів бот-мереж. Таким чином, НЗПЗ в складі бот-мереж можуть бути виявлені розподіленими багаторівневими системами [8] і напрям таких досліджень є перспективним.

Висновки. Незадокументовані закладки програмного забезпечення, яке використовується в локальних комп'ютерних мережах, можуть завдавати значної шкоди користувачам персональних комп'ютерів, а особливо підприємствам, які експлуатують комп'ютерні мережі та використовують спеціалізоване програмне забезпечення.

Моделі незадокументованих закладок програмного забезпечення дозволяють після відповідної формалізації включати їх в засоби виявлення. Застосування розроблених моделей НЗПЗ в РБС [8] дало можливість покращити ефективність виявлення бот-мереж, складовими яких вони були.

Напрямок подальших досліджень є конкретизація та визначення множини функцій, які формуватимуть елементи НЗПЗ, з метою представлення їх поведінковими сигнатурами для покращення ефективності виявлення.

Література

1. ДЕРЖАВНИЙ СТАНДАРТ УКРАЇНИ Захист інформації. Технічний захист інформації. Терміни та визначення. ДСТУ 3396.2-97
2. Kumar, N.J., Singh, P., Bali, R.S., Misra, S., Ullah, S. (2015). An intelligent clustering scheme for distributed intrusion detection in vehicular cloud computing, *Cluster Computing*, 18(3), 1263–1683. DOI: 10.1007/s10586-015-0463-7
3. Boukhlof, D., Kazar, O., Kahloul, L. (2016). Network Security: Distributed Intrusion Detection System using Mobile Agent Technology, *International Journal of Communication Networks and Distributed Systems*, 16(4). DOI: 10.1504/IJCND.2016.10001612
4. Boukhlof, D., Kazar, O. (2012). Hybrid Approach based Mobile Agent for Distributed Intrusion Detection System, *Journal of Information Security Research*, 3(1), 30–40. DOI: 10.1109/ICEEL.2012.6360647
5. Markowsky, G., Savenko, O., Sachenko, A. (2019). Distributed Malware Detection System Based on Decentralized Architecture in Local Area Networks. *Advances in Intelligent Systems and Computing III*, 871, 582–598. DOI: 10.1007/978-3-030-01069-0_42
6. Sochor T. Attractiveness Study of Honeypots and Honeynets in Internet Threat Detection / T. Sochor, M. Zuzcak // *Proceedings of the 22-nd International Conference Computer Networks*. – Brunów (Poland), June 16-19, 2015, Vol. 522. – Pp. 69-81.
7. Савенко О. С. Генерація моделей комп'ютерних вірусних програм в системі оцінки достовірності результатів роботи антивірусних засобів / О. С. Савенко, С. В. Мостовий // *Вісник Хмельницького національного університету. Технічні науки*. – 2005. – № 4, т. 1. – С. 198-200.
8. Савенко О. С. Архітектура розподіленої багаторівневої системи виявлення шкідливого програмного забезпечення в локальних комп'ютерних мережах / О. С. Савенко // *Вчені записки Таврійського національного університету. Технічні науки*. – 2018. – Т. 29 (68), № 2. – С. 172–181.

Reference

1. DERZhAVNIY STANDART UKRAYINI Zahist informaciyi. Tehnichnij zahist informaciyi. Termini ta viznachennya. DSTU 3396.2-97
2. Kumar, N.J., Singh, P., Bali, R.S., Misra, S., Ullah, S. (2015). An intelligent clustering scheme for distributed intrusion detection in vehicular cloud computing, *Cluster Computing*, 18(3), 1263–1683. DOI: 10.1007/s10586-015-0463-7
3. Boukhlof, D., Kazar, O., Kahloul, L. (2016). Network Security: Distributed Intrusion Detection System using Mobile Agent Technology, *International Journal of Communication Networks and Distributed Systems*, 16(4). DOI: 10.1504/IJCND.2016.10001612
4. Boukhlof, D., Kazar, O. (2012). Hybrid Approach based Mobile Agent for Distributed Intrusion Detection System, *Journal of Information Security Research*, 3(1), 30–40. DOI: 10.1109/ICEEL.2012.6360647
5. Markowsky, G., Savenko, O., Sachenko, A. (2019). Distributed Malware Detection System Based on Decentralized Architecture in Local Area Networks. *Advances in Intelligent Systems and Computing III*, 871, 582–598. DOI: 10.1007/978-3-030-01069-0_42
6. Sochor T. Attractiveness Study of Honeypots and Honeynets in Internet Threat Detection / T. Sochor, M. Zuzcak // *Proceedings of the 22-nd International Conference Computer Networks*. – Brunów (Poland), June 16-19, 2015, Vol. 522. – Pp. 69-81.
7. Savenko O. S. Generaciya modelej komp'yuternih virusnih program v sistemi ocinki dostovimosti rezul'tativ roboti antivirusnih zasobiv / O. S. Savenko, S. V. Mostovij // *Visnik Hmel'nickogo nacionalnogo universitetu. Tehnichni nauki*. – 2005. – № 4, t. 1. – S. 198-200.
8. Savenko O. S. Arhitektura rozpodilenoj bagatorivnevoyi sistemi viyavleniya shkidlivogo programnogo zabezpechennya v lokalnih komp'yuternih merezhah / O. S. Savenko // *Vcheni zapiski Tavrijskogo nacionalnogo universitetu. Tehnichni nauki*. – 2018. – T. 29 (68), № 2. – S. 172–181.

Рецензія/Peer review : 26.11.2019

Надрукована/Printed : 09.01.2020