

УДК 004.491.42

О.С. САВЕНКО

Хмельницький національний університет

ФОРМАЛІЗАЦІЯ ШКІДЛИВОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ В ЛОКАЛЬНИХ КОМП'ЮТЕРНИХ МЕРЕЖАХ НА ОСНОВІ АЛГЕБРАЇЧНИХ СИСТЕМ

В роботі проаналізовано відомі моделі шкідливого програмного забезпечення. Їх використання дозволило вирішити оцінку складності обчислювального процесу при визначенні віднесення програм до множини шкідливого програмного забезпечення. Вони надають формальне представлення означення комп'ютерних вірусів. Але не в повній мірі охоплюють все шкідливе програмне забезпечення. Тому не можуть бути основою для практичної інтерпретації всього шкідливого програмного забезпечення з метою його представлення для підвищення достовірності ідентифікації. В зв'язку з цим було розроблено алгебраїчні системи алгебри та моделі для формалізованого подання властивостей шкідливого програмного забезпечення, які дозволили створити його удосконалену модель в локальних мережах. Вона, на відміну від класичної моделі Коена, деталізована до рівнів властивостей і дозволяє здійснити представлення шкідливого програмного забезпечення через механізми його поширення в плоскій моделі пам'яті. Її особливістю є розгляд паралельних середовищ поширення в пам'яті різних комп'ютерних систем в локальній мережі. Це надало змогу формалізовано представити шкідливе програмне забезпечення в локальних комп'ютерних мережах з метою його ідентифікації згідно характеристичних властивостей. Місцем можливого розміщення шкідливого програмного забезпечення в комп'ютерних системах локальних комп'ютерних мереж розглядалися внутрішня пам'ять, зовнішня пам'ять та мережні пакети. Таке виділення трьох основних складових необхідно для побудови моделей шкідливого програмного забезпечення стало основою для охоплення всіх його різновидів. Було виділено характеристичні властивості шкідливого програмного забезпечення пов'язані з системними викликами, які відносяться до роботи з 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 оперативного запам'ятовуючого пристрою, команди для роботи в мережі. Сукупність розроблених алгебр є основою для системного розподілу інформації про характерні особливості шкідливого програмного забезпечення в процесі свого життєвого циклу. Використання таких характеристик дозволить здійснювати виявлення шкідливого програмного забезпечення шляхом аналізу особливостей, які проявлятимуться при виконанні функцій.

Ключові слова: алгебраїчна структура, алгебра, модель, шкідливе програмне забезпечення, локальна комп'ютерна мережа.

O.S. SAVENKO

Khmelnytsky National University

FORMALIZING MALICIOUS SOFTWARE IN LOCAL COMPUTER NETWORKS TO BASIS OF ALGEBRAIC SYSTEMS

Familiar models of malicious software are analyzed in the work. Their use made it possible to decide the assessment of the complexity of the computing process in determining the assignment of programs to the set of malicious software. They provide a formal representation of the definition of computer viruses. But not fully cover all malware. Therefore, they can not be the basis for the practical interpretation of all malware in order to present it to enhance the authenticity of the identification. In this connection, algebraic systems of algebra and models for the formal representation of the properties of malicious software were developed, which allowed to create its advanced model in local networks. She, unlike the classic Cohen model, is detailed to the levels of properties and allows the representation of malicious software through the mechanisms of its distribution in a flat memory model. Its feature is the consideration of parallel environments in the memory of various computer systems on the local network. This allowed for the formalization of malicious software in local computer networks in order to identify it according to the characteristic properties. The place of possible placement of malicious software in computer systems of local computer networks considered internal memory, external memory and network packets. Such an allocation of the three major components necessary for the construction of malware models has become the basis for the coverage of all its varieties. Characteristics of malicious software related to system calls that are related to working with files, RAM and commands in the network were identified: creation, opening, closing, deleting, reading, writing, adding, finding, receiving attributes and setting attributes, access commands to the operating memory, commands for working on the network. The set of developed algebras is the basis for the systematic distribution of information about the peculiarities of malware in the process of its life cycle. Using such features will allow detection of malicious software by analyzing the features that will occur when performing functions.

Keywords: algebraic systems, algebra, model, malicious software, local computer network.

Вступ. Постановка задачі. Поширення шкідливого програмного забезпечення (ШПЗ) в інформаційних системах комп'ютерних мереж створює проблеми користувачам. Наявні засоби його виявлення на сьогодні не задовільняють потреб користувачів. Особливо це стосується задач по виявленню ШПЗ на випередження, на етапі його початкового поширення. Як правило, виявлення ШПЗ відбувається вже після того, коли воно поширювалось на протязі певного часу і виконувало деструктивні дії. Відомі різноманітні антивірусні засоби, які здійснюють виявлення ШПЗ на різних етапах його життєвого циклу, не забезпечують високої достовірності його виявлення. Сучасне ШПЗ побудоване, як складні багатофункційні програмні системи та комплекси з використанням ефективних методів створення програмних засобів та

методів поширення зловмисного коду. Для організації ефективної протидії таким засобам, необхідним є подальший розвиток теорії та практики створення систем виявлення ШПЗ. Окремим напрямком, який потребує дослідження та вирішення, є представлення об'єктів ШПЗ такими формальними структурами, що дозволили б підвищувати достовірність ідентифікації ШПЗ. Крім того, на сьогодні недостатньо описано та формалізовано ШПЗ, як об'єкти для ідентифікації, а також як певну загальноприйнятну формалізовану систему, яка була б основою для подальшого розвитку.

Досліджуватимемо здійснення функціонування шкідливого програмного забезпечення в розподілених обчислювальних системах комп'ютерних мереж. Для розробки нових моделей та методів виявлення шкідливого програмного забезпечення необхідним є формалізація безпосередньо шкідливого програмного забезпечення, інших програмних об'єктів, а також апаратних засобів комп'ютерних систем та мереж, як єдиного інформаційного простору.

Інформаційний простір містить такі основні компоненти: інформаційні ресурси, засоби інформаційної взаємодії, інформаційну інфраструктуру. Іншими словами, необхідно формалізувати компоненти інформаційного простору з метою представлення шкідливого програмного забезпечення і середовища їх функціонування для подальшої побудови моделей шкідливого програмного забезпечення і розробки методів та засобів їх виявлення.

Інформаційна інфраструктура включає програмно-технічні засоби комп'ютерних мереж, які забезпечують організацію взаємодії інформаційних потоків, функціонування та розвиток засобів інформаційної взаємодії та інформаційного простору організації. В якості середовища функціонування шкідливого програмного забезпечення і його виявлення розглядатимемо корпоративну мережу, яка складається з групи локальних мереж організації. Середовищем перебування шкідливого програмного забезпечення в корпоративній мережі будуть засоби пам'яті та процесори. Тому, необхідно при формалізації інформаційного простору враховувати ці апаратні засоби. Корпоративна мережа є частиною відкритого інформаційного простору і, як правило, є частиною Internet простору. Шкідливе програмне забезпечення може функціонувати і поширюватись в одній комп'ютерній системі, при розширенні обчислювальних ресурсів до локальної комп'ютерної мережі – в комп'ютерних системах локальної мережі і при розширенні обчислювальних ресурсів до глобальних комп'ютерних мереж – в комп'ютерних системах глобальних мереж. Таким чином, інформаційний простір, в якому може функціонувати ШПЗ має підпростори на рівні корпоративних мереж та на рівні окремих комп'ютерних систем, де зберігаються ті ж можливості для розвитку і поширення ШПЗ, які є на рівні глобальної мережі. Враховуючи можливість порівняння результатів функціонування на поширення ШПЗ в різних комп'ютерних системах в локальній мережі на відміну від розгляду тільки однієї комп'ютерної системи, тоді найменшим підпростором будемо вважати корпоративну мережу і результати отримані в ній можна поширити на фрагменти всього простору тобто глобальної мережі, розглядаючи її як складену з локальних мереж.

Пов'язані роботи. Можливість створення самовідтворюваних автоматів представлено в роботі [1] Д. фон Нейманом. Подані формалізації будувались на представленні програм як кінцевих автоматів. Враховуючи сьогодиніший розвиток інформаційних технологій та комп'ютерних засобів такі представлення не можуть бути використані з метою формалізації ШПЗ, бо відносяться до рівня абстракції, який знаходиться найближче до апаратних засобів, і не враховує таких особливостей абстрагування, як моделі поведінок ШПЗ. Крім того, врахування представлення через кінцеві автомати суттєво збільшує обчислювальну складність алгоритмів ідентифікації ШПЗ.

Перше означення та модель комп'ютерного вірусу представив Фред Коен. Зокрема в [2, 3] на основі машини Тюрінга були описані множини комп'ютерних вірусних програм. Основою поданої Коеном моделі є можливість поширення вірусного коду в плоскій моделі пам'яті. При такому переміщенні отримуються нові елементи, які здатні знову поширювати себе. Розроблена модель не враховує конструкційних особливостей комп'ютерних систем і зокрема їх розміщення в мережах.

Модель Л.Адлемана, яка представлена в [4] базується на нумеруванні символів та призначенні їм Геделевого номеру, за яким шляхом певних обчислень можна було б встановити приналежність програми до множини ШПЗ. Розроблена ним модель була використана для встановлення можливості спроможності розрізнити комп'ютерних вірус та корисну програму і для оцінки обчислювальної складності цього процесу. Результати отримані Л. Адлеманом дають негативну відповідь на можливість вирішення цієї проблеми. Запропонована ним модель не може бути використана для опису всього ШПЗ, яке наявне на сьогодні.

В роботах [5, 6] Ж. Бонфана, М. Качмарека і Ж.-І. Маріона запропонована модель визначення комп'ютерних вірусів базується на теоремі Кліні з теорії алгоритмів, згідно якої, по аналогії з моделлю Л.Адлемана, кожній функції ставиться у відповідність число. Тоді, результатом функції від числа буде саме це число, що означає можливість тиражування програм. Запропонована модель була використана для більш точної оцінки обчислення складності алгоритмів розпізнавання комп'ютерних вірусів.

Подловченко Р.І. в роботах [7–9] було запропоновано формальні моделі програм та використано їх для оцінки складності обчислювального процесу. Крім того, нею було досліджено та розроблені моделі еквівалентних програм, які відповідали поліморфним та метаморфним вірусам, і для них здійснено віднесення до класів обчислювальних задач, враховуючи особливості коду, за складністю обчислювального процесу. Запропоновані моделі можуть бути використані при розробці засобів виявлення комп'ютерних вірусів, особливо з метаморфним і поліморфним навантаженням, але не враховують наявності поширення

ШПЗ в комп'ютерних мережах.

Таким чином, відомі моделі дозволяють вирішити оцінку складності обчислювального процесу при визначенні віднесення програм до множини ШПЗ, надають формальне представлення означення комп'ютерних вірусів, але не в повній мірі охоплюють все ШПЗ і тому не можуть бути основою для практичної інтерпретації всього ШПЗ з метою його представлення для підвищення достовірності ідентифікації.

Основна частина. Позначимо множину всього шкідливого програмного забезпечення V , яке перебуває в комп'ютерних системах локальних мережах. Тобто розглядатимемо те ШПЗ, 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. Представимо ШПЗ в локальних комп'ютерних мережах, особливістю якого є втілення у виконувани файли, завантажувальний сектор жорсткого диску, оперативний запам'ятовуючий пристрій та поширення мережею своїх копій, алгебраїчною системою типу $\tau = (\alpha, \beta)$:

$$\mathcal{U}_V = (V, \Omega_F, \Omega_P), \quad (1)$$

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 $\beta_1 = 0, 1, 2, \dots$; $\alpha = 1$, $\beta = 1$ – парності операцій, тому тип системи $\tau = (1, 1)$. Елементами множини $v_j \in V$ ($j = 1, 2, \dots$) вважатимемо всі об'єкти файлової системи, завантажувального сектору диску, оперативної пам'яті, мережні пакети, які відносяться до розглядуваного ШПЗ. Елементи $v_0 \in V_0 \subseteq V$ є одиничними елементами, тобто такими, що містять єдиний функціонал, вміст якого полягає у необхідності здійснення самокопіювання з метою поширення, але без конкретного функціонального наповнення для виконання технічно цих дій. Решта операцій представлені іншими функціями. Ці елементи, що формують множину V_0 є породжуючими для решти різних елементів множини V . Функції з множини Ω_F виконуються на елементах v_0 , що формує інші об'єкти, які належатимуть множині V , а також можуть виконуватись на інших елементах множини V , які не належать множині V_0 . Функції з множини Ω_F не завжди успішно виконуватимуться по відношенню до елементів з множини V , тому для представлення ШПЗ в локальних мережах вибрано також множину предикатів, яка відображатиме результат успішного/неуспішного виконання функцій.

Функції F_{α_1} ($\alpha_1 = 0, 1, 2, \dots$) з множини Ω_F визначимо, як такі що здійснюватимуть відображення елементів з множини V на неї. Їх конкретне визначення залежатиме від поділу множини Ω_F на підмножини за різними характеристичними властивостями ШПЗ. Предикати P_{β_1} ($\beta_1 = 0, 1, 2, \dots$) з множини Ω_P визначимо, як такі що будуть істинними при успішному виконанні операцій і хибними – в іншому випадку.

Множину Ω_F представимо її підмножинами $\Omega_{F_{\alpha_1}}$, які відображатимуть такі характеристичні для ШПЗ властивості та закладені в його функціонал особливості:

- 1) зберігання знань про механізм місцерозміщення своїх наступних копій;
- 2) пошук місця в пам'яті для розміщення своєї копії;
- 3) знання про механізми втілення у виконувани програми;
- 4) механізми запису в оперативну пам'ять;
- 5) приховування свого перебування в комп'ютерних системах;
- 6) пошук інших вузлів мережі для свого поширення;
- 7) механізми для формування і відправки мережних пакетів;
- 8) подолання механізмів захисту;
- 9) техніки запису своїх копій в головний завантажувальний сектор;
- 10) виконання деструктивних дій.

Ці характеристичні властивості ШПЗ пов'язані з системними викликами, які відносяться до роботи з 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. Відповідність характеристичних ознак ШПЗ системним викликам представлено в таблиці 1.

Реалізація характеристичних властивостей ШПЗ 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.

Під новими копіями ШПЗ вважатимемо співпадіння ШПЗ за семантикою, а не тільки за синтаксисом. Тому, при поширенні ШПЗ у випадку зміни синтаксису важливими є особливості функціоналу незалежно від синтаксису коду.

Місцем можливого розміщення ШПЗ в комп'ютерних системах локальних комп'ютерних мереж може бути оперативна пам'ять, зовнішня пам'ять та мережні пакети. Представлення місця перебування шкідливого програмного забезпечення в комп'ютерних системах зображено на рис. 1. Таке виділення трьох основних складових необхідно для побудови моделей ШПЗ, які б стали основою для розробки нових методів їх виявлення. Наявність ШПЗ в зовнішній пам'яті є характерним для усіх його різновидів. Перебування ШПЗ в мережних пакетах для частини ШПЗ є обов'язковим, оскільки характеризує механізми його поширення. Для іншої частини ШПЗ перебування в мережних пакетах не є обов'язковим, тобто їх поширення може відбуватись іншими шляхами, зокрема і через носії зовнішньої пам'яті. Використання

ШПЗ для свого перебування можливе тільки при ввімкненій працюючій КС. Для певної частини ШПЗ використання оперативної пам'яті є обов'язковим місцем розміщення та функціонування, а для іншої тільки місцем на час виконання. Врахування в моделях ШПЗ місця їх перебування є важливим елементом, який може бути використаний при їх виявленні, оскільки розробники ШПЗ закладають в нього знання про їх місцезнаходження в КС. Тобто ШПЗ володіє техніками перевірки свого місцезнаходження, що є важливим при розробці його моделей. Елементи місць розміщення ШПЗ потребуватимуть деталізації в залежності від їх функційного призначення та технічних характеристик, що впливатиме на моделі ШПЗ і потребуватиме їх деталізації і уточнення.

Таблиця 1

Відповідність характеристикних ознак ШПЗ системним викликам

Характеристичні властивості ШПЗ	створення	відкриття	закриття	видалення	читання	записування	додавання	знаходження	отримання атрибутів	встановлення атрибутів	команди доступу до ОП	команди для роботи в мережі
зберігання знань про механізм місцерозміщення своїх наступних копій								+	+		+	
пошук місця в пам'яті для розміщення своєї копії									+		+	
знання про механізми втілення у виконуваних програми	+	+	+	+							+	
механізми запису в оперативну пам'ять	+	+	+	+	+	+		+			+	
приховування свого перебування в комп'ютерних системах	+	+	+	+	+	+	+	+			+	
пошук інших вузлів мережі для свого поширення									+		+	+
механізми для формування і відправки мережних пакетів									+		+	+
подолання механізмів захисту	+	+	+	+	+	+	+	+	+	+	+	
техніки запису своїх копій в головний завантажувальний сектор						+					+	
виконання деструктивних дій	+	+	+	+	+	+	+	+	+	+	+	+



Рис. 1. Місця перебування ШПЗ

Розглядаючи ШПЗ з точки зору його місця розміщення та пошуку ним його для зберігання себе і своїх копій при поширенні, представимо модель ШПЗ за цією характерною властивістю. Виділимо в множині всіх програм підмножину шкідливого програмного забезпечення для якого характерна властивість полягає у збереженні закладеної у ШПЗ інформації про механізми поширення в частині перебування їх в зовнішній пам'яті, оперативній пам'яті і мережних пакетах. Здійснимо її формалізоване представлення для використання в процесі пошуку ШПЗ.

Виділимо в множині Ω_F підмножину Ω_{F_p} таким чином, щоб $\Omega_F = \bigcup_{p=1}^k \Omega_{F_p}$, де k – кількість характеристикних властивостей ШПЗ. Тоді, алгебраїчну систему для першої властивості, що характеризує знання про місцезнаходження ШПЗ при $p=1$ для всієї локальної мережі задамо так:

$$\mathcal{A}_{V,1} = \langle V, \Omega_{F_1}, \Omega_{F_1} \rangle \quad (2)$$

де Ω_{F_1} – множина операцій заданих на множині V , Ω_{F_1} – множина предикатів заданих на множині V .

Нехай $V = \bigcup_{s=1}^n V_s$, тобто виділимо в кожній КС локальної мережі підмножину ШПЗ V_s , де $s = 1, 2, \dots, n$. Для виділених підмножин з множини V в момент часу $t=0$ буде справедливе твердження $\bigcap_{s=1}^n V_s = \emptyset$. Дійсно, з самого початку початкового встановлення програмного забезпечення на всі КС мережі в них немає ШПЗ. В процесі збільшення часу їх роботи, тобто при $t > 0$, ймовірність появи ШПЗ певного на різних КС мережі зростає, тому справедливим може бути твердження $\bigcap_{s=1}^n V_s \neq \emptyset$. Задамо алгебру для першої властивості, що характеризує знання про місцезнаходження ШПЗ при $p=1$ для однієї з КС мережі так:

$$\mathfrak{B}_{V_s,1} = \langle V_s, \Omega_{F_1} \rangle, \quad (3)$$

де s – кількість вузлів ЛКМ; Ω_{F_1} – множина функцій заданих на множині V , яка впливає на місцерозміщення наступних копій ШПЗ. Ця множина функцій здійснює відображення копії ШПЗ в певний об'єкт зовнішньої пам'яті, оперативного запам'ятовуючого пристрою та мережного пакету. Якщо $v_{s,j,l} \in V_s$, де j – це номер елемента ШПЗ, l – номер версії j елемента, тоді $F_{1,k} \in \Omega_{F_1}$, де k – кількість функцій в множині Ω_{F_1} , $k \in N$. Ці функції $F_{1,k}$ здійснюють відображення елемента $v_{s,j,l}$ в множину V_s , тобто $F_{1,k}(v_{s,j,l}) = v_{s,j,l+1}$, де $v_{s,j,l+1} \in V_s$, якщо наступна копія ШПЗ створюватиметься в тій же КС мережі. Але наступна копія може створюватись і на іншій КС мережі, тому функцію $F_{1,k}$ можна задати так:

$$F_{1,k}(v_{s,j,l}) = \begin{cases} v_{s,j,l+1}, & \text{якщо } v_{s,j,l+1} \in V_s \\ v_{s',j,l'}, & \text{якщо } v_{s',j,l'} \in V_{s'} \end{cases}, \quad (4)$$

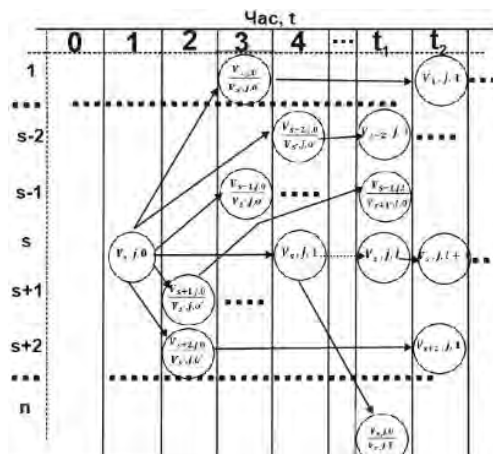
де s' – номер КС в мережі відмінний від s , l' – номер копії ШПЗ відмінний від l . Для функції $F_{1,k}$ існує також обернена функція $F_{1,k}^{-1}$, яка встановлює відповідність між елементами множини V так:

$$F_{1,k}^{-1}(v_{s,j,l+1}) = v_{s,j,l}, \quad F_{1,k}^{-1}(v_{s',j,l'}) = v_{s,j,l}. \quad (5)$$

Таблиця 2

Приклад поширення одного елемента ШПЗ в залежності від часу

Номер КС	Час, t									
	0	1	2	3	4	...	t_1	...	t_z	...
1				$v_{1,j,0}$ ($v_{s',j,0r}$)		...			$v_{1,j,1}$...
...
s-2					$v_{s-2,j,0}$ ($v_{s',j,0r}$)	...	$v_{s-2,j,1}$...
s-1				$v_{s-1,j,0}$ ($v_{s',j,0r}$)		...	$v_{s-1,j,1}$ ($v_{s+1,j,0r}$)			...
s		$v_{s,j,0}$			$v_{s,j,1}$...	$v_{s,j,1}$...	$v_{s,j,l+1}$...
s+1			$v_{s+1,j,0}$ ($v_{s',j,0r}$)		
s+2			$v_{s+2,j,0}$ ($v_{s',j,0r}$)			...			$v_{s+2,j,1}$...
...
n						...	$v_{n,j,0}$ ($v_{s',j,2r}$)			...

Рис. 2. Граф поширення ШПЗ в часі та в ЛКМ, представлений станами для одного з елементів множини V

Для КС з номером s у випадку появи j -го елемента множини V через час t від початку її функціонування в мережі можуть бути створені копії безпосередньо в цій же s -й КС або інших КС локальної мережі. Тому, результат поширення одного елемента ШПЗ можна відобразити послідовностями в таблиці 2. Дане представлення залежить від часу та вузла ЛКМ, тому його можна інтерпретувати як часову модель поширення ШПЗ в ЛКМ. Представлені в таблиці елементи, які відповідають копіям одного з елементів множини V , графом станів, де у вершинах розмістимо ці елементи, а дуги відповідатимуть за зв'язки між попередніми та наступними копіями елементів, зобразимо на рис. 2.

Зображені дугами графа на рис. 2 переходи, які здійснюватимуться між елементами, між різними вузлами ЛКМ міститимуть проміжні рівні, що відповідають за формування, пересилання та обробку мережних пакетів. Також, в дугах є ще один рівень переходів, який відображає виконання операції поширення копій із використанням оперативного запам'ятовуючого пристрою. Часову діаграму для графу з рис. 2 зобразимо на рис. 3.

		Час, t									
		0	1	2	3	4	...	t_1	...	t_z	...
1						
...
s-2						
s-1						
s						
s+1						
s+2						
...
n						

Рис. 3. Часова інтерпретація поширення елемента $v_{s,j,l}$ з множини V

Зведення даних про поширення копій зі всіх КС локальної мережі, наприклад, зображено на рис. 4.

Час, t																							
t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}	t_{11}	t_{12}	t_{13}	t_{14}	t_{15}	t_{16}	t_{17}	t_{18}	t_{19}	t_{20}	t_{21}	t_{22}	...	
																						...	

Рис. 4. Зведена часова інтерпретація поширення елемента $v_{s,j,l}$ з множини V

Наслідком з такої інтерпретації поширення елемента $v_{s,j,l}$ з множини V є можливість обчислення середнього часу поширення для одного елемента та багатьох за формулою:

$$\Delta t = \frac{t_z - t_0}{\sum_{i=1}^I I_i} \quad (6)$$

де I_i – кількість копій елемента $v_{i,j,l}$ з множини V в кожному вузлі ЛКМ, I – кількість КС в мережі, t_0 – початковий час, t_z – поточний час роботи. Також, цю формулу можна узагальнити для всіх елементів з множини V , які можуть поширюватись в ЛКМ, тоді середній час поширення для всіх елементів визначається за формулою:

$$\Delta t = \frac{t_z - t_0}{\sum_{j=1}^J \sum_{i=1}^I I_{i,j}}, \quad (7)$$

де $I_{i,j}$ – кількість копій j -го елемента $v_{i,j,l}$ з множини V в кожному вузлі ЛКМ, I – кількість КС в мережі, j -й елемент з множини V , t_0 – початковий час, t_z – поточний час роботи. Крім того, з формули 6 можна виразити швидкість поширення ШПЗ в ЛКМ за певний час:

$$W = \frac{\sum_{j=1}^J \sum_{i=1}^I I_{i,j}}{t_z - t_0}. \quad (8)$$

Цю швидкість W можна використати для оцінки прогнозу поширення ШПЗ на протязі деякого часу.

Важливим, також, є дослідження кількості поширення елементів з множини V в конкретних вузлах мережі в позиціонуванні їх від місця поширення та від певної копії одного елемента, тобто визначення кількості поширених копій у вузлах ЛКМ одного елемента $v_{i,j,l}$ з множини V . Матрицею суміжності представимо в таблиці 3 залежність породжених копій елементів у вузлах ЛКМ від копій елементів з різних вузлів ЛКМ.

Матриця є несиметричною, бо граф поширення ШПЗ в ЛКМ є орієнтованим. Очевидно, що копії елементів, які поширені з однієї КС можуть поширити свої копії теж на цю ж КС. Тому, за умови поширення ШПЗ протягом тривалого часу можливою є наявність всіх не нулевих елементів матриці. Такі матриці для одного або більше елементів з множини V , отримані багатократно на протязі певного часу, дозволяють визначати рівень безпеки розподіленої багаторівневої системи виявлення ШПЗ в ЛКМ.

Таблиця 3

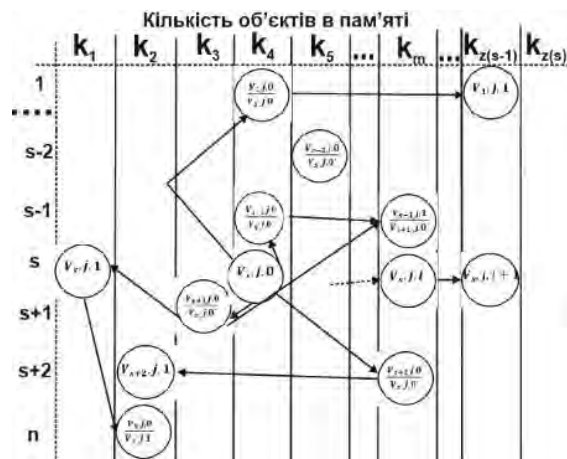
Матриця суміжності породжених копій елементів у вузлах ЛКМ

Номер КС	1	2	...	n	Разом:
1	$l_{1,1}$	$l_{1,2}$...	$l_{1,n}$	$\sum_{j=1}^n l_{1,j}$
2	$l_{2,1}$	$l_{2,2}$...	$l_{2,n}$	$\sum_{j=1}^n l_{2,j}$
...
n	$l_{n,1}$	$l_{n,2}$...	$l_{n,n}$	$\sum_{j=1}^n l_{n,j}$
Разом:	$\sum_{i=1}^n l_{i,1}$	$\sum_{i=1}^n l_{i,2}$...	$\sum_{i=1}^n l_{i,n}$	$\sum_{j=1}^n \sum_{i=1}^n l_{i,j}$

Таблиця 4

Приклад поширення одного елементу ШПЗ залежно від кількості об'єктів в пам'яті

Номер КС	Кількість об'єктів в пам'яті									
	k_1	k_2	k_3	k_4	k_5	...	k_m	...	$k_{z(s)-1}$	$k_{z(s)}$
1				$v_{1,j,0}$ ($v_{2',j,0}$)		...			$v_{1,j,1}$	
...
s-2					$v_{s-2,j,0}$ ($v_{s',j,0}$)	...	$v_{s-2,j,1}$			
s-1				$v_{s-1,j,0}$ ($v_{s',j,0}$)		...	$v_{s-1,j,1}$ ($v_{s+1,j,0}$)			
s	$v_{s,j,1}$			$v_{s,j,0}$...	$v_{s,j,1}$...	$v_{s,j,1+1}$	
s+1			$v_{s+1,j,0}$ ($v_{s',j,0}$)			...				
s+2		$v_{s+2,j,1}$...	$v_{s+2,j,0}$ ($v_{s',j,0}$)			
...
n		$v_{n,j,0}$ ($v_{s',j,1}$)				...				

Рис. 5. Граф поширення ШПЗ в ЛКМ, представлений копіями для одного з елементів множини V

Розглянемо ШПЗ відносно місця розміщення без врахування часу, протягом якого воно поширювалось і буде поширюватись. Особливою ознакою виділимо кількість об'єктів, які можуть бути розміщені в пам'яті кожної КС мережі. Ця кількість є скінченною і залежить від обсягу пам'яті, причому в різних КС вона може бути різною. Для КС з номером s у випадку появи j -го елементу множини V через час t від початку її функціонування в мережі можуть бути створені копії безпосередньо в цій же s -й КС або інших КС локальної мережі. Тому, результат поширення одного елементу ШПЗ можна відобразити послідовностями в таблиці 4, які на відміну від часової моделі можуть бути розміщеними не спочатку, а в

певній частині пам'яті і поширюватись в об'єкти як до так і після об'єкту з $v_{s,j,l}$. Зображення у вигляді графу подано на рис. 5.

Задамо узагальнене представлення даних з таблиці 3 матрицею суміжності. Для цього введемо позначення об'єктів пам'яті змінними та їх представлення у вигляді лінійних многочленів з коефіцієнтами. Дійсно, кожному об'єкту, відомості про який представлено в таблицях файлових систем можна поставити у взаємно-однозначну відповідність змінну з індексом. Оскільки кількість таких об'єктів скінченна, тоді многочлен буде мати скінченну кількість доданків. Побудуємо коефіцієнти змінних многочлена в такий спосіб, щоб вони містили інформацію про ШПЗ, його походження та різні його інші атрибути. Виділимо з них такі та введемо їх позначення: (1) номер об'єкта з цього ж місця перебування; (2) номер об'єкта цієї ж КС але іншого місця перебування; (3) номер об'єкта з іншої КС, з якого надійшло ШПЗ; (4) номер КС, з якої надійшов пакет з ШПЗ. Для їх відображення коефіцієнтами використаємо базис i, j, k в такий спосіб: $\alpha_{1,s,p} + \alpha_{2,s,p} i + \alpha_{3,s,p} j + \alpha_{4,s,p} k$, де $\alpha_{x,s,p}$ - x -й коефіцієнт, що вибирається з множини $\{1;2;3;4\}$, s – номер КС в мережі, p – номер об'єкту, для якого задано коефіцієнт. Крім того, для спрощення представлення таких коефіцієнтів, в яких зберігаються відомості про ШПЗ, здійснимо їх подання в матричному вигляді так:

$$\begin{pmatrix} \alpha_{1,s,p} & \alpha_{2,s,p} \\ \alpha_{3,s,p} & \alpha_{4,s,p} \end{pmatrix} \text{ або } \begin{pmatrix} \alpha_{1,s,p} & \alpha_{2,s,p} \\ \alpha_{3,s,p} & \alpha_{4,s,p} \end{pmatrix} \quad (9)$$

Позначимо $A_{s,p} = \alpha_{1,s,p} + \alpha_{2,s,p} i + \alpha_{3,s,p} j + \alpha_{4,s,p} k$, тоді в результаті отримуємо таке представлення об'єктів в мережі:

$$\begin{matrix} A_{1,1}x_{1,1} & +A_{1,2}x_{1,2} & +A_{1,3}x_{1,3} & \dots & +A_{1,p_1}x_{1,p_1} \\ A_{2,1}x_{2,1} & +A_{2,2}x_{2,2} & +A_{2,3}x_{2,3} & \dots & +A_{2,p_2}x_{2,p_2}, \\ \dots & \dots & \dots & \dots & \dots \\ A_{n,1}x_{n,1} & +A_{n,2}x_{n,2} & +A_{n,3}x_{n,3} & \dots & +A_{n,p_n}x_{n,p_n} \end{matrix} \quad (10)$$

де кожен рядок-вираз відображає стан об'єктів в КС, а всі рядки – стан об'єктів в локальній мережі, причому p_s – кількість об'єктів в КС, s – кількість КС в мережі. Для представлення матрицею відомостей про об'єкти необхідно доповнити всі послідовності виразів до $\max p_s$ доданками, в яких змінні $x_{s,p_s+j} = 0$, причому $p_s + j \leq \max(p_s)$ для всіх $j = 0, 1, \dots, \max(p_s) - p_s$. Таким чином, отримаємо таку матрицю відомостей про об'єкти в мережі:

$$\begin{pmatrix} \begin{pmatrix} \alpha_{1,1,1} & \alpha_{2,1,1} \\ \alpha_{3,1,1} & \alpha_{4,1,1} \end{pmatrix} & \begin{pmatrix} \alpha_{1,1,1} & \alpha_{2,1,1} \\ \alpha_{3,1,1} & \alpha_{4,1,1} \end{pmatrix} & \begin{pmatrix} \alpha_{1,1,1} & \alpha_{2,1,1} \\ \alpha_{3,1,1} & \alpha_{4,1,1} \end{pmatrix} & \dots & \begin{pmatrix} \alpha_{1,1,1} & \alpha_{2,1,1} \\ \alpha_{3,1,1} & \alpha_{4,1,1} \end{pmatrix} \\ \begin{pmatrix} \alpha_{1,2,1} & \alpha_{2,2,1} \\ \alpha_{3,2,1} & \alpha_{4,2,1} \end{pmatrix} & \begin{pmatrix} \alpha_{1,2,1} & \alpha_{2,2,1} \\ \alpha_{3,2,1} & \alpha_{4,2,1} \end{pmatrix} & \begin{pmatrix} \alpha_{1,2,1} & \alpha_{2,2,1} \\ \alpha_{3,2,1} & \alpha_{4,2,1} \end{pmatrix} & \dots & \begin{pmatrix} \alpha_{1,2,1} & \alpha_{2,2,1} \\ \alpha_{3,2,1} & \alpha_{4,2,1} \end{pmatrix} \\ \dots & \dots & \dots & \dots & \dots \\ \begin{pmatrix} \alpha_{1,n,1} & \alpha_{2,n,1} \\ \alpha_{3,n,1} & \alpha_{4,n,1} \end{pmatrix} & \begin{pmatrix} \alpha_{1,n,1} & \alpha_{2,n,1} \\ \alpha_{3,n,1} & \alpha_{4,n,1} \end{pmatrix} & \begin{pmatrix} \alpha_{1,n,1} & \alpha_{2,n,1} \\ \alpha_{3,n,1} & \alpha_{4,n,1} \end{pmatrix} & \dots & \begin{pmatrix} \alpha_{1,n,1} & \alpha_{2,n,1} \\ \alpha_{3,n,1} & \alpha_{4,n,1} \end{pmatrix} \end{pmatrix} \quad (11)$$

Враховуючи, що можливими місцями перебування ШПЗ можуть бути основна та вторинна пам'ять і мережні пакети в кожній КС мережі, то задамо їх множиною $Q = \{0; 1; 2\}$, де елемент 0 відповідатиме за перебування в основній пам'яті, елемент 1 – вторинній пам'яті, елемент 2 – мережному пакеті. Знаходження копії елементу множини V в мережному пакеті може бути в момент часу, коли вже здійснено формування пакету та відбувається його пересилання і отримання, тому саме в цей момент часу копія не перебуватиме в основній чи вторинній пам'яті. Позначивши місце перебування M_q , де $q \in Q$, введемо відповідну функцію так:

$$R_{1,k}(v_{s,j,l}) = \begin{cases} 0, \text{ якщо } v_{s,j,l} \in M_0 \\ 1, \text{ якщо } v_{s,j,l} \in M_1, \\ 2, \text{ якщо } v_{s,j,l} \in M_2 \end{cases} \quad (12)$$

Далі використовуємо значення цієї функції $R_{1,k}(v_{s,j,l})$ в матриці відповідності кожній КС мережі місця перебування елементу множини V і такі матриці будемо для кожного j -го елементу.

Задамо алгебру для властивості, що характеризує пошук ШПЗ місця в пам'яті для розміщення своєї копії в комп'ютерних системах. При цьому встановимо $p=2$ для однієї з КС мережі так:

$$\mathfrak{B}_{V_{s,2}} = \langle V_s, \Omega_{P_2} \rangle, \quad (13)$$

де s – кількість вузлів ЛКМ; Ω_{P_2} – множина функцій заданих на множині V , яка здійснює пошук ШПЗ місця в пам'яті для розміщення своєї копії в комп'ютерних системах.

Введемо предикати на множині V таким чином, що вони відображатимуть результат виконання відповідних функцій в множину $\{0;1\}$, тобто наявність зв'язку між елементами $v_{s,j,l}$ та $v_{s,j,l+1}$, так:

$$F_{1,k}(v_{s,j,l}, v_{s,j,l+1}) = \begin{cases} 1, \text{ якщо } F_{1,k}(v_{s,j,l}) = v_{s,j,l+1} \\ 0, \text{ якщо } F_{1,k}(v_{s,j,l}) \neq v_{s,j,l+1} \end{cases}, \quad (14)$$

де s – номер КС в мережі, l – номер копії ШПЗ, $v_{s,j,l} \in V_s$. Тоді, задамо модель наступним чином:

$$\mathfrak{M}_{V,1} = \langle V; \Omega_{F_{1,k}} \rangle, \quad (15)$$

де $\Omega_{F_{1,k}}$ – множина предикатів, заданих на множині V .

Розділимо всі функції, які задані на множині V і виконують дії по втіленню ШПЗ у виконуваних програми, на підмножини так: функції запису в початок виконуваної програми із збереженням її функціоналу, функції запису в середину виконуваної програми із збереженням її функціоналу, функції

запису в кінець виконуваної програми із збереженням її функціоналу, функції запису в різні частини виконуваної програми із збереженням її функціоналу, функції запису в початок виконуваної програми без збереження її функціоналу, функції запису в середину виконуваної програми без збереження її функціоналу, функції запису в кінець виконуваної програми без збереження її функціоналу, функції запису в різні частини виконуваної програми без збереження її функціоналу. Задамо алгебру для цієї властивості, що характеризує механізм втілення ШПЗ у виконувану програму при $p=3$ для однієї з КС мережі так:

$$\mathfrak{B}_{V,3} = \langle V, \Omega_{F_3} \rangle, \quad (16)$$

де s – кількість вузлів ЛКМ; Ω_{F_3} – множина функцій заданих на множині V , яка здійснює втілення своїх копій ШПЗ у виконувану програму. Цю множину функцій Ω_{F_3} розділимо на вісім підмножин за способом втілення у корисну програму $\Omega_{F_3} = \bigcup_{r=1}^8 \Omega_{F_3,r}$. Процес втілення у вибрану виконувану програму здійснюється шляхом виконання відповідної функції $\Omega_{F_3,r}$ ($r = 1, 2, \dots, 8$) з врахуванням структури виконуваної програми та типу операційної системи. Відмінність між корисною виконуваною програмою і програмою, в яку втілено ШПЗ, відобразиться в структурі і кожного разу при використанні типової функції буде однаковим. Тобто результатом успішного виконання функції буде типова послідовність дій, результат якої відобразиться в результуючому коді виконуваної програми. Тип операційної системи, в якій можуть активуватись функції $\Omega_{F_3,r}$ ($r = 1, 2, \dots, 8$) теж закладено їх розробниками і враховується при пошуку об'єктів для втілення. Введемо для відображення типу операційної системи множину $O = \{a_1, a_2, \dots, a_g\}$, де g – кількість типів.

Додамо до цієї множини функцій Ω_{F_3} підмножини за способом втілення не в корисну програму, а формуванням окремого файлового об'єкту. Для таких функцій приймемо $r = 9$ і відповідна підмножина $\Omega_{F_3,9}$.

Задамо алгебру для властивості, що характеризує механізми запису ШПЗ в оперативну пам'ять при $p=4$ для однієї з КС мережі так:

$$\mathfrak{B}_{V,4} = \langle V, \Omega_{F_4} \rangle, \quad (17)$$

де s – кількість вузлів ЛКМ; Ω_{F_4} – множина функцій заданих на множині V , яка здійснює запис ШПЗ в оперативну пам'ять. В множину функцій входять ті функції, які відносяться до того ШПЗ, яке постійно перебуває в оперативній пам'яті, і позначимо його підмножиною $\Omega_{F_4,1}$. До другої підмножини віднесемо ті функції, які переводять об'єкти з вторинної пам'яті в оперативну пам'ять, і після цього вони поширюватимуться вже перебуваючи там. Позначимо цю підмножину $\Omega_{F_4,2}$. До третьої підмножини $\Omega_{F_4,3}$ віднесемо всі об'єкти з вторинної пам'яті, які при поширенні використовуватимуть оперативну пам'ять, і ці функції виконуватимуть дії по реалізації механізмів перенесення та виконання в ній команд ШПЗ.

Задамо алгебру для властивості, що характеризує механізми приховування ШПЗ свого перебування в комп'ютерних системах при $p=5$ для однієї з КС мережі так:

$$\mathfrak{B}_{V,5} = \langle V, \Omega_{F_5} \rangle, \quad (18)$$

де s – кількість вузлів ЛКМ; Ω_{F_5} – множина функцій заданих на множині V , яка здійснює приховування ШПЗ свого перебування в комп'ютерних системах.

Задамо алгебру для властивості, що характеризує механізми пошуку інших вузлів мережі для свого поширення при $p=6$ для однієї з КС мережі так:

$$\mathfrak{B}_{V,6} = \langle V, \Omega_{F_6} \rangle, \quad (19)$$

де s – кількість вузлів ЛКМ; Ω_{F_6} – множина функцій заданих на множині V , яка здійснює пошуку інших вузлів мережі для свого поширення.

Задамо алгебру для властивості, що характеризує механізми для формування і відправки мережних пакетів в комп'ютерних системах при $p=7$ для однієї з КС мережі так:

$$\mathfrak{B}_{V,7} = \langle V, \Omega_{F_7} \rangle, \quad (20)$$

де s – кількість вузлів ЛКМ; Ω_{F_7} – множина функцій заданих на множині V , яка здійснює для формування і відправки мережних пакетів в комп'ютерних системах.

Задамо алгебру для властивості, що характеризує механізми подолання систем захисту в комп'ютерних системах при $p=8$ для однієї з КС мережі так:

$$\mathfrak{B}_{V,8} = \langle V, \Omega_{F_8} \rangle, \quad (21)$$

де s – кількість вузлів ЛКМ; Ω_{F_8} – множина функцій заданих на множині V , яка здійснює подолання систем захисту в комп'ютерних системах.

Задамо алгебру для властивості, що характеризує техніки запису своїх копій в головний завантажувальний сектор в комп'ютерних системах при $p=9$ для однієї з КС мережі так:

$$\mathfrak{B}_{V,9} = \langle V, \Omega_{F_9} \rangle, \quad (22)$$

де s – кількість вузлів ЛКМ; Ω_{F_9} – множина функцій заданих на множині V , які здійснюють запис своїх копій в головний завантажувальний сектор в комп'ютерних системах.

Задамо алгебру для властивості, що характеризує реалізацію виконання деструктивних дій в комп'ютерних системах при $p=10$ для однієї з КС мережі так:

$$\mathfrak{B}_{V,10} = \langle V, \Omega_{F_{10}} \rangle, \quad (23)$$

де s – кількість вузлів ЛКМ; $\Omega_{F_{10}}$ – множина функцій заданих на множині V , які здійснюють виконання деструктивних дій в комп'ютерних системах. Ці деструктивні дії відмінні від функцій реалізованих у властивостях 1–9.

Для заданих множин функцій алгебр, що відображають формальні властивості ШПЗ в процесі його поширення, введемо множини предикатів $\Omega_{F_{s,k}}$ так, що вони відображатимуть результат успішного/неуспішного виконання відповідних функцій в множину $\{0; 1\}$, де s – номер КС в мережі. Тоді, задамо моделі, які відповідатимуть розглядуваним властивостям наступним чином:

$$\mathfrak{M}_{V,k} = \langle V; \Omega_{F_{s,k}} \rangle, \quad (24)$$

де $\Omega_{F_{s,k}}$ – множина предикатів, заданих на множині V , k – номер властивості ШПЗ.

Висновки. Сукупність розроблених алгебр є основою для системного розподілу інформації про характерні особливості ШПЗ в процесі свого життєвого циклу. Використання таких характеристик дозволить здійснювати виявлення ШПЗ шляхом аналізу особливостей, які проявлятимуться під час виконання функцій. Тобто виконання кожної функції на множині ШПЗ здійснюватиметься типовим способом, знання про який використовуватиметься при виявленні.

Формалізовані властивості ШПЗ представлені розробленими алгебрами задано моделями, які дозволили створити удосконалену модель ШПЗ в локальних мережах, яка на відміну від класичної моделі Коена, деталізована до рівнів властивостей ШПЗ, дозволяє представити ШПЗ через механізми його поширення в плоскій моделі пам'яті, особливістю якої є розгляд паралельних середовищ поширення в пам'яті різних КС в локальній мережі. Це надасть змогу формалізовано представити ШПЗ в локальних комп'ютерних мережах з метою його ідентифікації згідно з характеристичними властивостями.

Отримані моделі є важливими для теорії і практики створення ефективних систем виявлення шкідливого програмного забезпечення в локальних комп'ютерних мережах.

Напрямок подальших досліджень є конкретизація та визначення множини функцій, які формуватимуть елементи ШПЗ, з метою представлення їх поведінковими сигнатурами для підвищення ефективності їх ідентифікації.

Література

1. Фон Нейман Дж. Теория самовоспроизводящихся автоматов / Фон Нейман Дж. – М. : Мир, 1971. – 281 с.
2. Cohen F. Computational aspects of computer viruses / F. Cohen // Computers and Security. – 1989. – Vol. 8. – P. 325–344.
3. Cohen F. Computer viruses: theory and experiments / F. Cohen // Computers and Security. – 1987. – Vol. 6. – P. 22–35.
4. Adleman L. An Abstract Theory of Computer Viruses / L. Adleman // CRYPTO '88. – P. 354–374.
5. Bonfante G., Kaczmarek M., Marion J.-Y. A Classification of viruses through recursion theorems / G. Bonfante, M. Kaczmarek, J.-Y. Marion // CiE. – 2007. – P. 73–82.
6. Bonfante G. Abstract detection of computer viruses / G. Bonfante, M. Kaczmarek, J.-Y. Marion // Munich, APPSEM II, 2005.
7. Подловченко Р.И. Иерархия моделей программ / Р.И. Подловченко // Программирование. – 1981. – № 2. – С. 3–14.
8. Подловченко Р.И. Полугрупповые модели программ / Р.И. Подловченко // Программирование. – 1981. – № 4. – С. 3–13.
9. Подловченко Р.И. Регулярные модели программ / Р.И. Подловченко, Н.А. Аланакян // Программирование. – 1993. – № 4. – С. 3–11.

References

1. Fon Neiman Dzh. Teoriya samovosproizvodiaschchykh avtomatov / Fon Neiman Dzh. – M. : Myr, 1971. – 281 s.
2. Cohen F. Computational aspects of computer viruses / F. Cohen // Computers and Security. – 1989. – Vol. 8. – P. 325–344.
3. Cohen F. Computer viruses: theory and experiments / F. Cohen // Computers and Security. – 1987. – Vol. 6. – P. 22–35.
4. Adleman L. An Abstract Theory of Computer Viruses / L. Adleman // CRYPTO '88. – P. 354–374.
5. Bonfante G., Kaczmarek M., Marion J.-Y. A Classification of viruses through recursion theorems / G. Bonfante, M. Kaczmarek, J.-Y. Marion // CiE. – 2007. – P. 73–82.
6. Bonfante G. Abstract detection of computer viruses / G. Bonfante, M. Kaczmarek, J.-Y. Marion // Munich, APPSEM II, 2005.
7. Podlovchenko R.Y. Yerarkhiya modelei prohramm / R.Y. Podlovchenko // Prohrammyrovanye. – 1981. – № 2. – S. 3–14.
8. Podlovchenko R.Y. Poluhgruppovyye modely prohramm / R.Y. Podlovchenko // Prohrammyrovanye. – 1981. – № 4. – S. 3–13.
9. Podlovchenko R.Y. Rehuliarnyye modely prohramm / R.Y. Podlovchenko, N.A. Alanakian // Prohrammyrovanye. – 1993. – № 4. – S. 3–11.

Рецензія/Peer review : 17.04.2018 р.

Надрукована/Printed : 10.05.2018 р.

Стаття рецензована редакційною колегією