computing@computingonline.net
www.computingonline.net

ISSN 1727-6209
International Journal of Computing

АДАПТИВНА ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ ВИЯВЛЕННЯ ТРОЯНСЬКИХ ПРОГРАМ В КОМП'ЮТЕРНИХ СИСТЕМАХ

Олег Савенко, Сергій Лисенко

Хмельницький національний університет,
вул. Інститутська 11, 29016 Хмельницький, Україна
e-mail: kism@beta.tup.km.ua, sirogyk@ukr.net

Резюме: *Розроблено адаптивну інформаційну технологію виявлення троянських програм в комп'ютерних системах, суть якої полягає у використанні поведінкових моделей класів троянських програм, відмінністю якої від відомих є те, що процес виявлення не потребує побудови баз сигнатур, дає змогу виявляти нові невідомі троянські програми та підвищує достовірність і ефективність процесу виявлення.*

Ключові слова: *троянська програма, адаптивна інформаційна технологія, життєвий цикл троянської програми, нечіткий логічний висновок, штучні імунні системи, антивірусний сканер, антивірусний монітор.*

ADAPTIVE INFORMATION TECHNOLOGY OF THE TROJAN' DETECTION IN COMPUTER SYSTEMS

Oleg Savenko, Sergiy Lysenko

Khmelnytskyi National University,
11 Instytutska street, 29016 Khmelnytskyi, Ukraine
e-mail: kism@beta.tup.km.ua, sirogyk@ukr.net

Abstract: *Adaptive information technology of computer systems Trojans diagnosing, which includes methods of diagnosing computer systems in monitor and scanner modes and allows improving reliability and efficiency, is developed. It is based on the behavioral model and the model of diagnosis. Computer system Trojan diagnosis software, which made it possible to detect the new Trojans with high reliability and efficiency, was developed.*

Keywords: *Trojans, computer system Trojan diagnosis, fuzzy logic, artificial immune systems, antivirus software, antiviral monitor, antiviral scanner.*

ВСТУП

Об'єднання комп'ютерних систем (КС) в локальні мережі та їх підключення до глобальної мережі Internet створює проблеми, пов'язані з їх функціюванням та використанням. Серед них вагоме місце займає шкідливе програмне забезпечення (ШПЗ), яке призводить до неправильного функціонування програмного та апаратного забезпечення. Аналіз ситуації щодо ШПЗ показує інтенсивне зростання чисельності троянських програм (ТП), здатних виконувати в КС деструктивні або шкідливі дії. Розробники ТП знаходять нові способи їх потрапляння в КС, застосовують маскування від антивірусного ПЗ, вдосконалюють код ТП. В свою чергу розробники антивірусного ПЗ постійно вдосконалюють інформаційні технології діагностування КС, оновлюють антивірусні бази, застосовують сучасні механізми виявлення ШПЗ. Проте наявні факти викрадення конфіденційної інформації та здійснення деструктивних дій в КС, в якому встановлене антивірусне ПЗ, свідчать про недоліки відомих технологій виявлення ТП в КС, які орієнтовані на виявлення

відомих ТП, та не повністю адаптовані до розпізнавання нових ТП.

Аналіз найпоширеніших інформаційних технологій (ІТ) антивірусного діагностування (АД) КС, якими є сигнатурний аналіз і метод контрольних сум, не здатні виявляти нові ТП, що суттєво знижує достовірність та ефективність діагностування. ІТ, які ґрунтуються на евристичному аналізі, мають високу ймовірність хибних спрацювань [1-3].

Антивірусні компанії увесь час займаються активним вивченням інструментів і методів протидії новим небезпекам. Постійне збільшення швидкості випуску нових видів ШПЗ досягли межі, при якій звичайні системи оновлення для протидії виявились недостатніми (див. рис.1) [4].

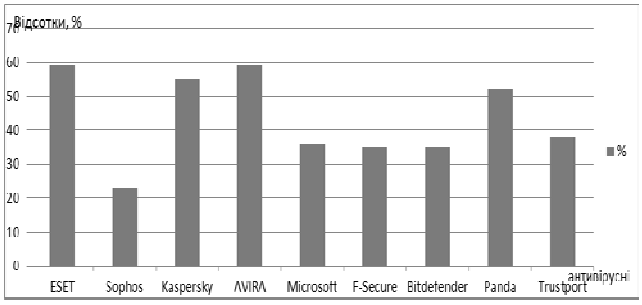

Рис 1. – Достовірність виявлення нових ТП

1. ПОСТАНОВКА ЗАДАЧІ

Новітніми підходами до діагностування КС на наявність невідомих ТП є залучення компонентів штучного інтелекту (ШІ). Доцільність їх застосування пояснюється невизначеністю і практично неформалізованістю поведінки ТП на всіх етапах свого життєвого циклу, а також можливістю надати властивості адаптивності інформаційній технології, за допомогою якої є можливим здійснювати АД.

Таким чином, **постає задача** розроблення нової ІТ, яка б давала змогу виявляти троянські програми в КС в режимах монітора та сканера. Антивірусний монітор повинен аналізувати поведінку програмних об'єктів і робити висновок про небезпеку інфікування КС троянськими програмами. З цією метою доцільним є залучення апарату нечіткої логіки. Антивірусний сканер має дозволити здійснювати сканування КС на предмет підміни системних файлів та інших файлів троянськими версіями і здійснити перевірку цілісності даних КС. Для цього доцільно використати алгоритми штучних імунних систем, що надасть адаптивної характеристики процесу діагностування комп'ютерних систем на наявність ТП. Розробка адаптивної інформаційної технології повинна підвищила достовірність та ефективність виявлення троянських програм в КС.

2. ПОВЕДІНКОВА МОДЕЛЬ ТРОЯНСЬКИХ ПРОГРАМ

Життєвий цикл (ЖЦ) ТП складається з етапів потрапляння на віддалену КС, активізації та виконання закладених деструктивних дій ТП, на основі чого розроблено їх поведінкову модель [5]:

$$M_v = \langle \Theta, S, V, L, Aff, \varepsilon, Z \rangle, \qquad (1)$$

де Θ – множина усіх троянських програм; S – етапи ЖЦ троянської програми $s_i \in S$, $i = \overline{1,3}$; $V = |V_{mp}|$ – матриця відношень m дій ТП та p портів мережних протоколів; $L = |L_{ab}|$ – матриця відношень дій $a \in A$ ТП і $b \in B$ структурних одиниць операційної системи; Aff – функція, яка визначає взаємодію між об'єктами КС і ТП v_j, тоді множина $a \in Aff(b_i, v_j)$ є набором можливих дій, які ТП v_j завдає об'єкту (об'єктам) b_i; ε – відношення між ТП та її станами, тоді для $v \in \Theta$ та $s \in S$, відношення $v \varepsilon s$ означає, що ТП v перебуває в стані s; відношення $v \bar{\varepsilon} s$ означає, що ТП v не перебуває в стані s; Z – характеристичні параметри відношень, $Z = \{z_k\}$ – вектор деструктивних дій об'єкта з нормованими пріоритетними вагами $P = \{p_k\}$, ($\sum p_k = 1$), що враховують рівень їхньої небезпеки для КС [2]. Також в поведінкову модель ТП введено позначення $\longrightarrow$, суть якого полягає у заданні відношення між трьома поняттями, а саме: якщо $s_i \xrightarrow{a} s_{i+1}$, то дія $a \in A$ спричинює перехід із стану s_i в стан s_{i+1}. Тоді ТП, життєвий цикл якої має усі етапи, проходить можливий шлях: $s_0 \xrightarrow{V,L} s_1 \xrightarrow{V,L} s_2 \xrightarrow{V,L} s_3$, де $s_i \xrightarrow{V,L} s_{i+1}$ означає можливість видозміненого ЖЦ, коли, наприклад, етап потрапляння ТП в КС здійснюється не мережею або етап активізації виконується шляхом надходження сигналу мережею, а не локально.

На основі поведінкової моделі було

побудовано моделі ТП кожного класу з урахування їх особливостей та функціонального навантаження. Так модель класу Trojan-Backdoor матиме вигляд:

$$M_{\Theta_{BD}} = \langle \Theta_{BD}, A_{BD}, B_{BD}, W_v, Inf, X, Y, Z \rangle, \quad (2)$$

де Θ_{BD} – множина троянських програм класу Trojan-Backdoor; $A_{\Theta_{BD}} = A'_{\Theta_{BD}} \cup A''_{\Theta_{BD}}$ – дії ТП, $A'_{\Theta_{BD}}$ - дії ТП, при потраплянні якої відбувається створення нового файлу, $A''_{\Theta_{BD}}$ – дії ТП, при потраплянні якої відбувається підміна системних файлів троянськими версіями; $W_v \in W$ – множини відправлених з КС файлів, шляхом виконання дій $a \in A$, що утворює множину ознак невірного функціонування структурних одиниць ОС КС $b_{BDi} \in B_{BD}$, $B_{BD} = \{b_{BD1}, b_{BD2}, ..., b_{BDn}\}$; Inf – ознака інфікування КС; X – відношення, що описує виконання ТП $v \in \Theta$ дій $a \in A$, $(v,a) \in X$, де $X \subset \Theta \times A$; Y – відношення дій ТП $a \in A$ та структурних одиниць ОС $(a,b) \in Y$, де $Y \subset A \times B$; Z – відношенням дій ТП $a \in A$ та файлів $w \in W$, $(a,w) \in Z$, де $Z \subset A \times W$.

3. ВИЯВЛЕННЯ ТРОЯНСЬКИХ ПРОГРАМ В КОМП'ЮТЕРНИХ СИСТЕМАХ

Процес виявлення троянських програм в комп'ютерних системах складається з двох підпроцесів діагностування в режимах монітора та сканера. Частини процесу виявлення троянських програм в КС позначено як Ω, $\Omega = \{\Omega_1, \Omega_2, \Omega_3, \Omega_4, \Omega_5, \Omega_6\}$ та Δ, $\Delta = \{\Delta_1, \Delta_2, \Delta_3, \Delta_4\}$. Процес виявлення в режимі монітора складається з етапів: Ω_1 – відслідковування потоків, що здійснюються через системні порти КС; Ω_2 – відслідковування виконання системних функцій в КС; Ω_3 – блокування виконання програмним об'єктом системних функцій, підозрілість яких визначена на інших етапах процесу АД; Ω_4 – фазифікація в межах системи нечіткого логічного висновку (НЛВ) для введення нечіткості шляхом задання ступенів підозрілості функціонування ПЗ та ступенів небезпеки інфікування КС; Ω_5 – робота машини логічного висновку в межах системи НЛВ; Ω_6 – виконання процедури дефазифікації в межах системи НЛВ для визначення ступеня небезпеки інфікування КС троянською програмою. Процес сканування включає наступні етапи: Δ_1 – формування набору файлів, що підлягають процедурі створення набору захищених бінарних послідовностей; Δ_2 – генерація набору шаблонів файлів, відібраних на попередньому етапі та виконання кодування даних у визначеному форматі; Δ_3 – генерація детекторів згідно обраного алгоритму; Δ_4 – сканування КС співставлення захищених двійковий послідовностей об'єктів антивірусного діагностування зі згенерованими на попередньому етапі детекторами. З урахуванням зворотних зв'язків між етапами схему процесу виявлення ТП в КС представимо на рис. 2.

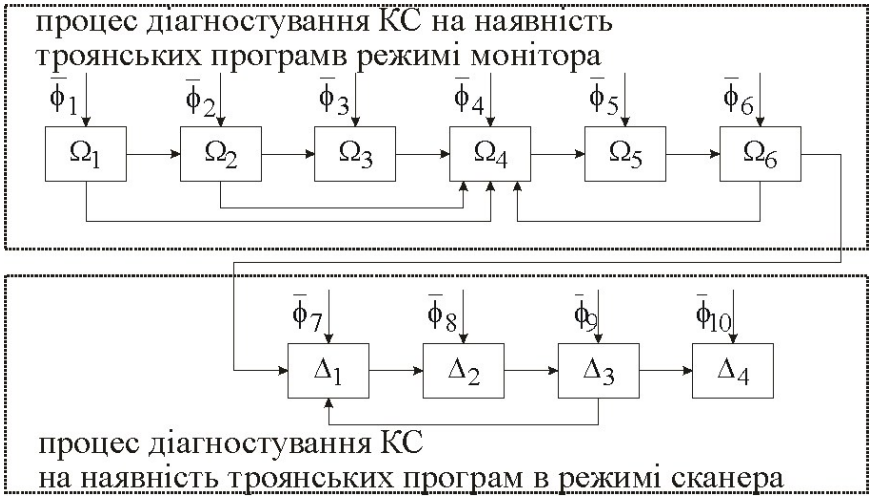

Рис. 2 – Формалізована схема процесу виявлення троянських програм в КС.

Для формалізації виконання етапів АД розроблено модель процесу виявлення ТП в КС [5] з урахуванням параметрів, які використовують вищевказані етапи у вигляді:

$$M_{\upsilon} = \left\langle \{E, R, M_W, f_m\}, \{E, H, S, D, E_v, f_s\} \right\rangle, \quad (3)$$

де для етапів $\Omega_1 - \Omega_6$: E – множина об'єктів діагностування в режимі монітора $e_k \in E$, а саме множина файлів КС, причому $\Theta \in E$; R – результуюче число $R \in [0,1]$, яке свідчить про ступінь небезпеки інфікування КС троянською програмою; відношення ε між об'єктами та станами, при чому для $\nu \in \Theta$ та $s \in S$; $f_m(I_m, I'_m, I''_m)$ – функція адаптивності діагностування КС в режимі монітора, параметри якої змінюються в залежності від вхідних даних, де I_m – набір діагностичної інформації, $I_m = \left\langle \Theta, V, L, R, \right\rangle$); I'_m – вектор результатів антивірусного діагностування, $I'_m = \left\langle R_1, R_2, ..., R_n \right\rangle$; I''_m – набір даних про виявлене ШПЗ, які збираються і використовуються в майбутньому як знання, $I''_m = \left\langle E, R \right\rangle$; для етапів $\Delta_1 - \Delta_4$: E – множина об'єктів діагностування в режимі сканера $e_k \in E$, H – множина об'єктів $h \in H$, що підлягають процедурі сканування на предмет можливого факту їх підміни; S – множина захищених двійкових послідовностей $s \in S$; D – множина детекторів, згенерованих для сканування системи $d \in D$, E_v – множина файлів КС, що були підмінені троянськими версіями; $f_s(I_s, I'_s, I''_s)$ – функція адаптивності діагностування КС в режимі сканера, де I_s – набір діагностичної інформації, $I_s = \left\langle H, S, D \right\rangle$; I'_s – результати антивірусного сканування представлені набором файлів, що були підмінені троянськими версіями, $I'_s = \left\langle E_1, E_2, ..., E_n \right\rangle$; I''_s – вектор інформації про оновлення системних файлів та встановлення нового ПЗ, як компонентів об'єкта діагностування $I''_s = \left\langle E'_1, E'_2, ..., E'_n \right\rangle$.

4. АДАПТИВНА ІТ ВИЯВЛЕННЯ ТРОЯНСЬКИХ ПРОГРАМ В КОМП'ЮТЕРНИХ СИСТЕМАХ

Розроблено адаптивну ІТ виявлення ТП в КС, яка дозволяє здійснити висновок щодо можливого інфікування КС троянською програмою як відомою, так і новою, а також дозволяє виявляти факт підміни системних файлів троянськими версіями [6]. Процес виявлення ТП в КС адаптується шляхом налаштування його параметрів в залежності від особливостей КС, що діагностується, а саме: типу операційної системи, встановленого ПЗ в КС, поведінок ТП, накопичених в процесі експлуатації КС. АІТ виявлення ТП в КС подано схемою на рис. 3.

Рис. 3– Адаптивна інформаційна технологія виявлення ТП в КС.

Розроблена адаптивна ІТ виявлення ТП в КС включає в себе метод виявлення ТП в КС в режимі монітора та метод побудови захищених послідовностей та генерації детекторів для виявлення ТП в КС в режимі сканера. Процес виявлення ТП в КС подано схемою на рис. 4.

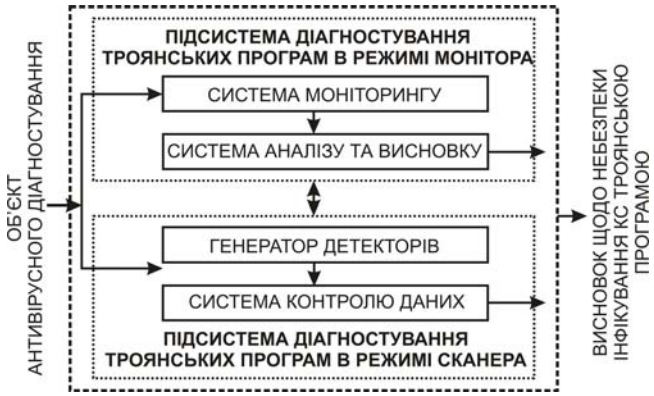

Рис. 4 – Схема процесу виявлення троянських програм в КС

5. МЕТОД ВИЯВЛЕННЯ ТП В КС В РЕЖИМІ МОНІТОРА

Розроблено новий метод виявлення ТП в КС в режимі монітора, який полягає в застосуванні апарату нечіткої логіки, і дає можливість зробити висновок щодо ступеня небезпеки інфікування КС троянською програмою [6]. З цією метою здійснюється побудова вхідної та вихідної лінгвістичних змінних з іменами: "Ступінь підозрілості програмного об'єкта" – для вхідної лінгвістичної змінної, і "Ступінь небезпеки інфікування" – для вихідної.

Для формування функції належності для вхідної лінгвістичної змінної розроблено новий метод, суть якого полягає в знаходженні для кожної дії найбільш імовірного порту потрапляння шляхом ранжування з побудовою матриці переваги $S = \left| s_{ij} \right|$, де

$$s_{ij} = \sum_{k=1}^{r} s_{ij}^{k} \cdot p_k \Big/ \sum_{k=1}^{r} s_{jk}^{k} \cdot p_k; \qquad s_{ji} = 1/s_{ij};$$

$s_{ii} = 1;\ i, j = \overline{1, m},\ s_{ij} = s_i / s_j,\ 0 < s_{ij} < \infty$. Потім здійснюється знаходження для матриці S власного вектора $\Pi = (\pi_1, ..., \pi_m)$, що відповідає максимальному додатному кореню λ характеристичного полінома $\left| S - \lambda \cdot E \right| = 0$; $S \cdot \Pi = \lambda \cdot \Pi$, де E – одинична матриця. Компоненти вектора Π $(\sum \pi_i = 1)$ ототожнюються з оцінкою $\mu_{X^p}(x_i, y_j)$. У результаті одержуємо матрицю відношення $V_p = \left| x_i, y_j \right|$, у якій кожному відношенню (x_i, y_j) відповідає значення $0 \le \pi \le 1$. Наступним кроком методу є побудова оптимізованої матриці $V_p^* = \left| x_i, y_j \right|$, з відношень (x_i, y_j) із значеннями $\pi_{\max}$ $(0 \le \pi_{\max} \le 1)$ та побудова нормованої кривої функції належності $\mu_{X^p}(R)$ вхідної змінної.

Приведемо фрагмент бази правил, згідно з якою здійснюється висновок, і у якій для кожного правила використані зв'язування "ТА" і значення Fi дорівнюють від 0 до 1:

П.1. If (penetration is L) and (aktyvisation is L) and (execution is L) then (suspiciousness is L) (1);

П.2. If (penetration is L) and (aktyvisation is L) and (execution is H) then (suspiciousness is M) (1);

П.3. If (penetration is L) and (aktyvisation is H) and (execution is L) then (suspiciousness is L) (1);

П.4. If (penetration is L) and (aktyvisation is H) and (execution is H) then (suspiciousness is H) (0,8);

П.5. If (penetration is H) and (aktyvisation is L) and (execution is L) then (suspiciousness is L) (1);

П.6 If (penetration is H) and (aktyvisation is L) and (execution is H) then (suspiciousness is M) (1);

П.7. If (penetration is H) and (aktyvisation is H) and (execution is L) then (suspiciousness is ML) (1);

……..

Можливі і інші стратегії побудови бази правил, де використовуються лише ті правила, які містять умови з ненульовими значеннями належності.

Для вирішення поставленої задачі було реалізовано систему нечіткого логічного висновку з використанням алгоритму Мамдані. Графічне представлення результатів систему НЛВ, реалізованої в середовищі Matlab, представлено на рис. 5. На рис. 6 зображені функції належності для побудованих вхідної та вихідної лінгвістичних змінних.

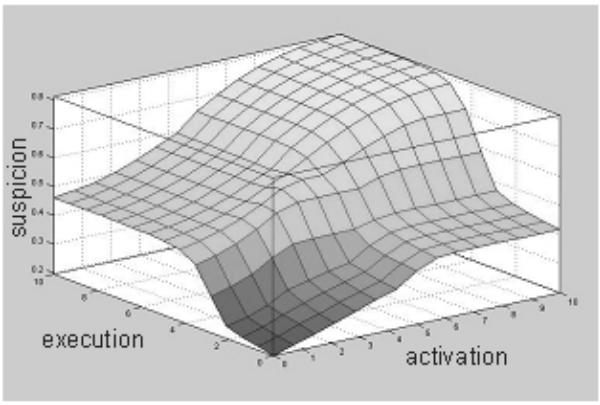

Рис. 5 – Результат нечіткого логічного висновку.

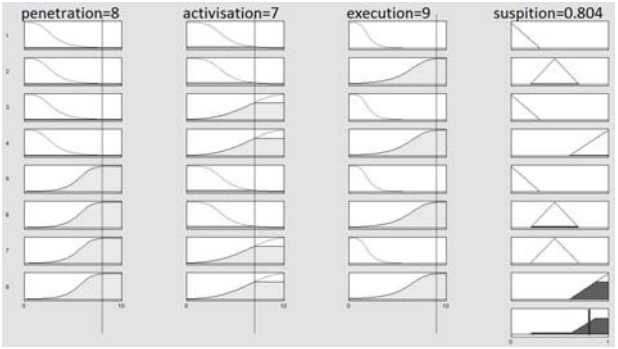

Рис. 6 – Графічне представлення правил та результату нечіткого логічного висновку.

Одержаний результат нечіткого логічного висновку 0,804 інтерпретується як ступінь небезпеки інфікування КС троянською програмою. Якщо отримане число перевищує певний наперед заданий поріг небезпеки інфікування, то згідно з прийнятою стратегією безпеки виконується блокування дій даного програмного об'єкту.

Метод також передбачає можливість накопичення нових поведінок ТП як знань для підвищення достовірності виявлення ТП в КС.

6. МЕТОД ПОБУДОВИ ЗАХИЩЕНИХ ПОСЛІДОВНОСТЕЙ ДЛЯ ВИЯВЛЕННЯ ТП В КС В РЕЖИМІ СКАНЕРА

Для виявлення троянських програм в КС в режимі сканера розроблено новий метод побудови захищених послідовностей та генерації детекторів. В основі методу лежить застосування алгоритмів штучних імунних систем (ШІС). Штучні імунні системи є адаптивними системами, принципи і моделі яких застосовані в багатьох прикладних задачах, таких як розпізнавання образів, комп'ютерна безпека, системи для вірусного виявлення, пошук даних, виявлення помилок, класифікація та ін [7].

Розроблений метод дозволяє виявляти факт підміни системних фалів троянськими версіями в залежності від типу ОС та ПЗ окремо взятої КС [8]. Метод передбачає наступі етапи:

1) формування набору системних файлів для сканування, які можна вважати еталонними;

2) генерація захищених двійкових послідовностей згідно визначеного формату;

3) на етапі антивірусного сканування КС виконання співставлення захищених двійкових послідовностей з детекторами;

4) у випадку збігу захищених послідовностей з детектором виконання сповіщення про виявлення підміни та перевірка на підозрілість поведінки програмного об'єкту.

Для генерації детекторів застосовано модифікований жадібний алгоритм, який дозволяє зменшити кількість детекторів, і у якому на відміну від алгоритмів, запропонованих в [9], генерація послідовностей виконується із застосування попередньої підготовки даних, а кодування захищених послідовностей та детекторів здійснюємо з урахуванням особливостей КС, що діагностується [8].

Генерація детектора відбувається у спосіб, при якому здійснюється вибір такого можливого детектора, який буде найчастіше збігатися з набором послідовностей, яких ще немає серед множини детекторів. З цією метою введено поняття шаблону, який є послідовністю r-бітів рядка довжини l, що містить (l-r)-порожній позицій. Таким чином, шаблоном $t_{i,s}$ є послідовність, в якій r визначених бітів починаються з позиції i та подані послідовністю s довжини r. Наприклад, для l=8, r=4, s=0111 шаблон $t_{3,s}$ має вигляд $t_{3,s}$=**0111**. Вважатимемо, що дві послідовності збігаються, якщо вони ідентичні хоча б в r позиціях (rcb-правило). Правим (лівим) набором шаблону вважатимемо шаблон, що містить усі порожні позиції заміщені бітами, наприклад, ***10011 – допустимий правий набір для **010011.

Для вибраної множини файлів КС $h \in H$, що підлягатимуть процесу діагностування в режимі сканера, будуємо множину захищених послідовностей $s \in S$ визначеного формату. Для означених послідовностей довжини r при $1 \le i \le (l-t+1)$ здійснюємо формування масивів S_r і S_l для правих і лівих наборів відповідно з $t_{i,s}$, що не збігаються з будь-якою послідовністю з S у спосіб: у випадку збігу шаблону в позиції $(l-r+1)$ масив заповнюється нулем, і одиницею – у іншому випадку. Кожен запис в S_r відповідає послідовності r шаблону $t_{i,s}$ і міститиме всі можливі варіанти, при яких дві послідовності можуть збігатися в r суміжних бітах. Зокрема, для $i = l - r + 1$ шаблон $t_{i,s}$ містить (l-r)-порожніх позицій, що слідують після r-суміжних бітів. Таким чином, $S' = S_r \times S_l$ представляє множину повністю визначених послідовностей, що відповідають даному шаблону, і якщо шаблон має нульовий запис в S', то відомо, що усі послідовності, які містять даний шаблон, збігатимуться з певною захищеною послідовністю. Фактично множина шаблонів у S' з ненульовими записами є набір допустимих детекторів. Аналогічно S' визначимо $D' = D_r \times D_l$ як кількість послідовностей для кожного шаблона, що не збігаються з

детекторами, які згенеровані на кожному попередньому кроці. Очевидно, що для порожньої множини детекторів D кількість кандидатів у детектори D' складе $2^{(l-r)}$. Для кожного нового детектора, який буде створений, обираються шаблони, що найбільше не збігаються із послідовностями. Оновлення множини D' виконується щоразу, коли створюється новий детектор, тому необхідно оновлювати множини правих та лівих наборів D_r і D_l. Для кожного нового детектора виконується пошук в множині D' для допустимого шаблону детектора з найбільшим ненульовим набором. Якщо таких записів кілька, то вибираємо один випадковим чином. Починаючи з даного шаблону здійснюється прохід через множину D' зліва та справа з додаванням щоразу 0 або 1 до початкового шаблону в залежності від того, який шаблон містить найбільше число послідовностей, що не збігаються з D. Множини D_r і D_l поетапно оновлюються щоразу при створенні нового детектора, встановлюючи відповідні нульові записи. Процес генерації детекторів і оновлення D_r, D_l і D' повторюється поки всі допустимі шаблони детекторів мають нульові записи в D'. Для будь-якого шаблону, що не належить S більше немає послідовностей, що збігаються з детекторами. Це означає, що здійснено покриття усіх захищених послідовностей детекторами. Необхідна кількість детекторів диктується необхідним рівнем ймовірності P_f. У кращому випадку необхідна кількість детекторів може скласти $N_R \geq (1-P_f)/P_M$ [9]. Кожен детектор може збігатися з 2^r шаблонів у кожній (l-r) позиції множини шаблонів $N_R \leq 2^r$. Також, якщо взяти до уваги захищені послідовності, то кожен шаблон не збігатиметься з S з ймовірністю $(1-2^{-r})^{N_S}$. Тоді кількість детекторів складе $N_R \approx 2_r \cdot (1-2^{-r})^{N_S}$.

7. РЕЗУЛЬТАТИ ДОСЛІДЖЕНЬ

Для запропонованої ІТ розроблено програмне забезпечення (ПЗ), що реалізує адаптивну інформаційну технологію виявлення ТП в КС. В процесі експлуатації розробленого ПЗ відбувається автоматичне накопичення інформації про виявлене ШПЗ, занесення його до бази поведінок (для антивірусного монітора), та автоматична генерація набору захищених послідовностей та детекторів у випадку оновлення чи встановлення нового ПЗ (для антивірусного сканера).

Для визначення достовірності роботи ПЗ було програмно згенеравано 3240 програмних об'єкти з функційним навантаженням троянських програм усіх класів. Дані програми потенційно невідомі для антивірусних баз фірм-розробників антивірусного ПЗ. Результати діагностування переставлено гістограмою на рис. 7 [10].

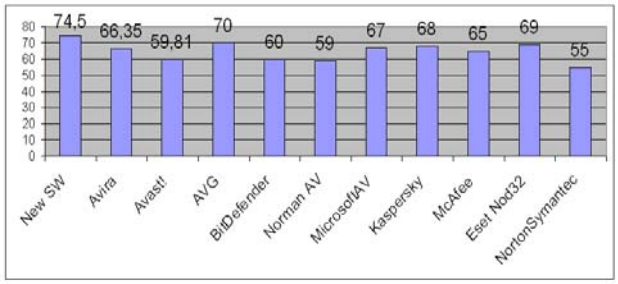

Рис. 7 – Достовірність виявлення ТП в КС у порівнянні з відомими технологіями

Отримані результати виявлення ТП в КС показали приріст достовірності 5-15% та підвищення ефективності у 1,4 рази у порівнянні з відомими засобами виявлення нових ТП в КС.

8. ВИСНОВКИ

Розроблено поведінкову модель ТП та поведінкові моделі класів ТП шляхом врахування особливостей функціонування протягом їх життєвого циклу та деструктивного характеру дій в КС, що уможливлює підвищити достовірність їх виявлення в КС.

Розроблено метод виявлення ТП в КС в режимі монітора на основі нечіткого логічного висновку з оцінкою лінгвістичної змінної на базі попарного порівняння експертом з урахуванням оціночних ознак, яке дозволяє врахувати особливості порівнюваних об'єктів, і яке не вимагає виконання умови транзитивності, що дає можливість визначити ступінь небезпеки інфікування КС троянськими програмами в умовах апріорної невизначеності. Розроблено метод побудови захищених послідовностей та генерації детекторів на основі штучних імунних систем шляхом, що уможливлює визначити факт підміни файлів троянськими версіями в процесі виявлення ТП в КС в режимі сканера.

Розроблено адаптивну інформаційну технологію виявлення ТП в КС, суть якої полягає у аналізі поведінки програмного об'єкту в КС та виявленні факту підміни системних файлів троянськими версіями, відмінністю якої від відомих є те, що параметри діагностування автоматично налаштовуються в залежності від особливостей КС, що дає можливість виявляти нові ТП, підвищити достовірність та

ефективність процесу виявлення ТП в КС, і не потребує побудови баз сигнатур. Розроблені алгоритми виявлення ТП в КС. Дослідження їх складності показало можливість їх програмної реалізації в межах адаптивної інформаційної технології. Розроблено ПЗ, що реалізує адаптивну інформаційну технологію виявлення ТП в КС, і задачею якого є діагностування КС на наявність як відомих так нових ТП. Отримані результати досліджень показали підвищення достовірності діагностування на 5-15%, та ефективності у 1,4 рази у порівнянні з відомими засобами виявлення нових ТП.

9. СПИСОК ЛІТЕРАТУРИ

[1] Yevgen Kasperskyi, *Computer hucking, 1-st edition,* Spb.: Piter, 2009. 208 p. (in Russian)

[2] Szor P., *The Art of Computer Virus Research and Defense,* Addison Wesley Professional, 2005. 744 p.

[3] Erbschloe M., *Trojans, worms and spyware. A Computer Security Professional's Guide to Malicious Code,* Burlington. Elsevier Butterworth-Heinemann, 2005. 212 p.

[4] AV Comparatives, Independent Tests of Anti-Virus Software, http://www.av-comparatives.org.

[5] O. Savenko, S. Lysenko, Model search process Trojans in personal computers, *Radio electronic and computer systems,* 7 (2008). – pp. 87-92. (in Ukrainian)

[6] R. Grafov, O. Savenko, S. Lysenko, Using fuzzy logic to search for Trojan software in computing systems, *Visnyk of Chernivtsi national university,* 6 (2009). – pp. 85-91. (in Ukrainian)

[7] M. Ayara, J. Timmis, L.N. de Lemos, R. de Castro, R. Duncan In J. Timmis, P.J. Bentley, Negative selection: How to generate detectors, *1st International Conference on Artificial Immune Systems, University of Kent at Canterbury*, 2002. p. 89-98.

[8] O. Savenko, S. Lysenko. The development of process of Trojan detection using artificial immune systems, *Visnyk of Khmelnytskyi national university,* 5 (2008). – pp. 183-188. (in Ukrainian)

[9] Forrest S., Perelson A.S., L. Allen, Self-nonself discrimination in a computer, *Proc. IEEE Symposium on Research in Security and Privacy*, Oakland, CA, 1994. p. 202-212.

[10] S. Lysenko, A. Gontar, A. Shevtsov, Software development of the intelligent method of the trojan detection in personal computers, *Visnyk of Khmelnytskyi national university,* 1 (2010). – pp.98-105. (in Ukrainian)

***Савенко Олег Станіславович**, доцент, к.т.н., декан факультету комп'ютерних систем та програмування Хмельницького національного університету; у 1999 році захистив кандидатську дисертацію на тему "Методи та засоби антивірусного комбіневаного діагностування персональних"; науковими інтересами є антивірусний захист комп'ютерних систем.*

***Сергій Лисенко**, к.т.н., старший викладач кафедри системного програмування Хмельницького національного університету; у 2005 закінчив магістратуру Хмельницького національного університету; у 2011 році захистив кандидатську дисертацію на тему "Адаптивна інформаційна технологія діагностування комп'ютерних систем на наявність троянських програм"; науковими інтересами є дослідження штучних імунних систем та апарату нечіткої логіки як засобів антивірусного діагностування комп'ютерних систем.*

computing@computingonline.net
www.computingonline.net

ISSN 1727-6209
International Journal of Computing

ADAPTIVE INFORMATION TECHNOLOGY OF THE TROJAN' DETECTION IN COMPUTER SYSTEMS

Oleg Savenko, Sergiy Lysenko

Khmelnytskyi National University,
11 Instytutska street, 29016 Khmelnytskyi, Ukraine
e-mail: kism@beta.tup.km.ua, sirogyk@ukr.net

Abstract: *Adaptive information technology of computer systems Trojans diagnosing, which includes methods of diagnosing computer systems in monitor and scanner modes and allows improving reliability and efficiency, is developed. It is based on the behavioral model and the model of diagnosis. Computer system Trojan diagnosis software, which made it possible to detect the new Trojans with high reliability and efficiency, was developed.*

Keywords: *Trojans, computer system Trojan diagnosis, fuzzy logic, artificial immune systems, antivirus software, antiviral monitor, antiviral scanner.*

1. INTRODUCTION

The analysis of the situation of development of the malware shows dynamic growth of their quantity. Among them the special place occupies a class of viruses – Trojans which unlike virus programs penetrate into computer system (CS) for the purpose of information plunder that represents real danger [1].

The actual problem of safety of various CS is a development of new more perfect information technologies which provide increase of reliability and efficiency of anti-virus software diagnosis.

2. TROJAN BEHAVIORAL MODEL

The behavioral Trojan model which takes into account the features of Trojans and formalizes the process of functioning of Trojans in CS during its life cycle and to takes into account the destructive nature of its actions in the CS Trojan was developed:

$$M_v = \langle \Theta, S, V, L, Aff, \varepsilon, Z \rangle, \qquad (1)$$

where Θ – the set of all the Trojans; S – Trojan's life cycle stages, that are penetration, activation and executing destructive actions, $s_i \in S$, $i = \overline{1,3}$; $V = |V_{mp}|$ – the matrix of relationship in which $m = \overline{1,k}$ are functions (mechanisms) which perform ways of the penetration of Trojans to CS of the user via system ports $p = \overline{1,h}$ of network protocols; $L = |L_{ab}|$ – a matrix of relationship, in which $a = \overline{1,\sigma}$ are operations of Trojan which negatively influence the structural components $b = \overline{1,\tau}$ of operating system; Aff – a function that defines the interaction between objects of CS and Trojan, thus the set $a \in Aff(b_i, \nu_j)$ is a set of possible actions, that Trojan causes the object (objects); ε – the ratio between Trojan and its stages then for $\nu \in \Theta$ and $s \in S$, the relationship means $\nu \varepsilon s$ that Trojan is in stage s; Z – characteristic parameters of mentioned relations, $Z = \{z_k\}$ – set of destructive actions with normalized priority weights $P = \{p_k\}$ ($\sum p_k = 1$) which take into account the level of danger to the CS.

To define the relationship between the Trojans, its actions and stages of its life cycle, the designation was worked in $\longrightarrow$ that means: if $s_i \xrightarrow{a} s_{i+1}$ then the action $a \in A$ causes a transition from s_i stage to stage s_{i+1}. Then the Trojan, which has a life cycle with all stages passes a possible way, is:

$$s_0 \xrightarrow{V,L} s_1 \xrightarrow{V,L} s_2 \xrightarrow{V,L} s_3, \qquad (2)$$

where $s_i \xrightarrow{V,L} s_{i+1}$ means the possibility of customized life cycle [2].

The models of each classes of Trojan with the regard to their specificity and functional significance

and is based on its behavioral model were produced.

A process of computer systems Trojans diagnosis consists of two subprocesses of monitoring Ω, $\Omega = \{\Omega_1, \Omega_2, \Omega_3, \Omega_4, \Omega_5, \Omega_6\}$ and scanning Δ, $\Delta = \{\Delta_1, \Delta_2, \Delta_3, \Delta_4\}$. The process of Trojan diagnosis in the monitor mode consists of: Ω_1 - monitoring the flows, carried out through the system ports of the CS; Ω_2 – monitoring the execution of the system functions in the CS; Ω_3 – blocking the implementation of application functions, defined as suspicious; Ω_4 – procedure of fuzzification within the fuzzy inference system (FIS) by entering the degrees of suspicion and the degrees of computer system infection danger; Ω_5 – implementation of the fuzzy logic engine; Ω_6 – procedure of defuzzification within the FIS to determine the risk of CS infection by Trojans [3].

In order to formalize the implementation of antivirus diagnosing stages the model of the Trojan diagnosing process was developed:

$$M_\upsilon = \langle \{E, R, M_W, f_m\}, \{E, H, S, D, E_v, f_s\} \rangle, \quad (4)$$

where for steps $\Omega_1 - \Omega_6$: E – set of diagnosing objects in the monitor mode $e_k \in E$, that is the set of CS files $\Theta \in E$; R – the resultant number $R \in [0,1]$ that indicates the danger degree of infection with Trojan; ε – ratio between objects $v \in \Theta$ and stage $s \in S$; M_W – set of behavioral Trojan models; $f_m(I_m, I'_m, I''_m)$ – adaptability function of computer system diagnosis in monitor mode, whose arguments vary with the input data, where I_m – set of diagnosis information, $I_m = \langle \Theta, V, L, R, \rangle$; I'_m – set of the antivirus diagnostics results, $I'_m = \langle R_1, R_2, ..., R_n \rangle$; I''_m – set of detected malicious software, $I''_m = \langle E, R \rangle$; for steps $\Delta_1 - \Delta_4$: E – set of diagnosing objects in the scanner mode, $e_k \in E$; H – set of objects to be scanned; S – set of protected binary sequences; D – set of generated detectors, $d \in D$, E_v – set of files that were substituted by Trojan versions; $f_s(I_s, I'_s, I''_s)$ – adaptability function of computer system diagnosis in scanner mode, where I_s – set of diagnosis information, $I_s = \langle H, S, D \rangle$; I'_s - antivirus scan results represented with a set of files that were substituted by Trojan versions, $I'_s = \langle E_1, E_2, ..., E_n \rangle$; I''_s – set of updated system files or installed new software, $I''_s = \langle E'_1, E'_2, ..., E'_n \rangle$.

3. INFORMATION TECHNOLOGY OF TROJANS DIAGNOSIS

To solve this problem a new information technology (IT) of CS Trojans diagnosing was proposed. It includes Trojan behavioral models and techniques of computer systems diagnosing in monitor and scanner modes with automated setting up of antivirus diagnosis parameters with the improving reliability and efficiency. New technology allows detecting known an unknown Trojans. The proposed AIT allows making a conclusion of regarding the degree of danger of CS infection by Trojans and to reveal the fact of system files substitution by Trojans' versions.

4. TECHNIQUE OF TROJAN DIAGNOSIS PROCESS IN MONITOR AND SCANNER MODES

A new technique for computer system Trojan diagnosis in monitor mode which uses fuzzy logic and is based on behavioral model was developed. It enables to make a conclusion about the degree of danger of CS infection by Trojans. For this purpose we construct the input and output linguistic variables with names: "suspicion degree of software object" – for the input linguistic variable, and "danger degree of the infection" – for output one.

The task of determination of membership function for input variable we will consider as the task of the ranking for each of mechanisms (functions) m_i of penetration ports p_j with the set of indications of danger Z and a choice of the most possible p_j with activation of some function m_i. Then we generate a matrix of advantage $S = |s_{ij}|$. Elements of given matrix s_{ij} are positive numbers: $s_{ij} = s_i / s_j$, $0 < s_{ij} < \infty$; $s_{ji} = 1/s_{ij}$, $s_{ii} = 1$, $i, j = \overline{1,l}$, l – amount of possible results. Elements s_{ij} of matrix S are defined by calculation of values of pair advantages to each indication separately taking into account their scales $Z = \{z_k\}$; $k = \overline{1,r}$ with usage of such formula

$$s_{ij} = \sum_{k=1}^{r} s_{ij}^k \cdot p_k \Big/ \sum_{k=1}^{r} s_{jk}^k \cdot p_k . \quad (5)$$

Eigenvector $\Pi = (\pi_1, ..., \pi_m)$ is defied by using a

matrix of advantage. This eigenvector answers maximum positive radical λ of characteristic polynomial $|S-\lambda\cdot E|=0$. $S\cdot\Pi=\lambda\cdot\Pi$, where E is an identity matrix. Elements of vector Π ($\sum\pi_i=1$) are identified with an estimation of experts who consider the accepted indications of danger. The same procedure is performed for all m_i. As a result we receive a matrix of relationship $V_p=|m_i,p_j|$, in which each pair (relationship) m_i, p_j value $0\le\pi\le1$ responds. Using matrix $V_p=|m_i,p_j|$, we build matrix $V_p^*=|m_i,p_j|$ in which the relationship (m_i,p_j) is used and the elements of this relationship have value $\pi_{\max}$ ($0\le\pi_{\max}\le1$). Using matrix $V_p^*=|m_i,p_j|$, we build normalized curve for membership function $\mu_{X^p}(R)$ of an input variable. As a part of the solution of the problem the FIS using Mamdani algorithm was realized [4].

A new technique for constructing the protected sequences and generation of detectors based on the use of algorithms for artificial immune systems was produced. It makes it possible to reveal the fact of system files substitution of Trojans' versions [5].

The method involves the following steps: forming a set of files to be scanned: system libraries, executables system services and device drivers, which can be taken as the samples; generate protected sequences and detectors depending on operating system; comparson of the protected sequences with detectors at the stage of virus scanning; notification about the substitution when the protected sequences match with detector; check the suspicion of software actions.

Thus protected sequences and detectors have format for GNU / Linux operating system:

$$D_i^L=\left\langle\begin{matrix}m_1...m_i...m_{x1},u_1...u_i...u_{x2},g_1...g_i...g_{x3},\\ s_1...s_i...s_{x4},t_1...t_i...t_{x5},C_1...C_i...C_{x6}\end{matrix}\right\rangle,(6)$$

where $m_1...m_i...m_{x1}$ - file mode (type, permissions); $u_1...u_i...u_{x2}$ – identifier of the file owner; $g_1...g_i...g_{x3}$ – identifier of the group owner; $s_1...s_i...s_{x4}$ – file size; $t_1...t_i...t_{x5}$ – time of last file modification; $C_1..C_i...C_{x6}$ – CRC of the file, $i=\overline{1,n}$, n – number of detectors.

Protected sequences and detectors have format for MS Windows operating system:

$$D_i^W=\left\langle\begin{matrix}s_1...s_i...s_{z1},t_1...t_i...t_{z2},\\ a_1...a_i...a_{z3},C_1...C_i...C_{z4}\end{matrix}\right\rangle \qquad (7)$$

where $s_1...s_i...s_{z1}$ – file size; $t_1...t_i...t_{z2}$ – time of last file modification; $a_1...a_i...a_{z3}$ – file attribute (read-only, hidden, system, archived); $C_1..C_i...C_{z4}$ – CRC of the file, $i=\overline{1,n}$, n – number of detectors.

Generation of detectors is performed using the modified negative selection algorithm.

Antivirus software for computer system diagnosing based on proposed algorithms was developed. The results confirmed that the use of AIT of computer system Trojan diagnosis increases the diagnosing reliability by 5-15%, and efficiency – by 40% in comparison with the known antivirus technologies [6].

6. CONCLUSIONS

The article is devoted to solving important scientific problem – increasing reliability and efficiency of computer systems diagnosing of the Trojans existence. The new techniques for computer system Trojan diagnosis in monitor and scanner modes which allows improving reliability and efficiency, are developed. Also information technology is based on the model of the Trojan diagnosing process, which allows performing diagnosing with high reliability was developed. Computer system Trojan diagnosis software, which made it possible to detect the new Trojans with high reliability and efficiency was developed.

7. REFERENCES

[1] Yevgen Kasperskyi, *Computer hucking, 1-st edition,* Spb.: Piter, 2009. – 208 p. (in Russian)

[2] Oleg Savenko, Sergiy Lysenko, Model search process Trojans in personal computers, *Radio Electronic and Computer Systems,* 7 (2008). – pp. 87-92. (in Ukrainian)

[3] Oleg Savenko, Sergiy Lysenko, Intelligent mehod and algorithms of the Trojan search in the computers systems, *Visnyk of the Vinnytsia Politechnical Institute,* 6 (2008). – pp. 129-137. (in Ukrainian)

[4] R. Grafov, O. Savenko, S. Lysenko, Using fuzzy logic to search for Trojan software in computing systems, *Visnyk of Chernivtsi National University,* 6 (2009). – pp. 85-91. (in Ukrainian)

[5] O. Savenko, S. Lysenko. The development of process of Trojan detection using artificial immune systems, *Visnyk of Khmelnytskyi National University,* 5 (2008). – pp. 183-188. (in Ukrainian)

[6] S. Lysenko, A. Gontar, A. Shevtsov, Software development of the intelligent method of the trojan detection in personal computers, *Visnyk of Khmelnytskyi National University,* 1 (2010). – pp. 98-105. (in Ukrainian)