УКРАЇНА **(19) UA** **(11) 135205** **(13) U**

(51) МПК
G06F 21/55 (2013.01)

МІНІСТЕРСТВО ЕКОНОМІЧНОГО РОЗВИТКУ І ТОРГІВЛІ УКРАЇНИ

(12) ОПИС ДО ПАТЕНТУ НА КОРИСНУ МОДЕЛЬ

(21) Номер заявки: **u 2018 12864**
(22) Дата подання заявки: **26.12.2018**
(24) Дата, з якої є чинними права на корисну модель: **25.06.2019**
(46) Публікація відомостей про видачу патенту: **25.06.2019, Бюл.№ 12**

(72) Винахідник(и):
Савенко Олег Станіславович (UA)
(73) Власник(и):
ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ,
вул. Інститутська, 11, м. Хмельницький, 29016 (UA)

(54) СПОСІБ ОРГАНІЗАЦІЇ ВЗАЄМОДІЇ КОМПОНЕНТІВ ДЕЦЕНТРАЛІЗОВАНИХ РОЗПОДІЛЕНИХ СИСТЕМ ВИЯВЛЕННЯ ЗЛОВМИСНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ НА ОСНОВІ РІВНІВ ЇХ БЕЗПЕКИ В ЛОКАЛЬНИХ КОМП'ЮТЕРНИХ МЕРЕЖАХ

(57) Реферат:

Спосіб організації взаємодії компонентів децентралізованих розподілених систем виявлення зловмисного програмного забезпечення на основі рівнів їх безпеки в локальних комп'ютерних мережах. Для функціонування таких децентралізованих розподілених систем взаємна координація роботи їх компонентів між собою в локальній мережі враховує рівень достовірності виявлення зловмисного програмного забезпечення в конкретних комп'ютерних системах мережі з використанням заданого порядку взаємодії компонентів системи, який включає визначення станів програмних модулів через рівні їх безпеки, обробку відповідей від кожної компоненти іншими компонентами на відправлені пакети, обробку компонентами системи невизначеностей, пов'язаних з відсутністю відповідей на відправлені пакети в мережі з використанням сканування заданих портів комп'ютерних систем, оцінку стану компонент системи та перевіряння цих оцінок між собою всіма компонентами, визначення стану децентралізованої розподіленої системи на основі обчислення рівня її безпеки, прийняття рішення про подальшу роботу системи в цілому на основі дослідження її стану кожним її компонентом окремо, вплив подій, які активують методи виявлення зловмисного програмного забезпечення, на зміну стану окремих компонентів та системи, здійснення дослідження інших комп'ютерних систем компонентами системи на наявність подібних проявів зловмисного програмного забезпечення та обмін такими отриманими результатами, обробку та оптимізацію статистичних даних, накопичених в системі кожною компонентою окремо, обміном знаннями всередині децентралізованої розподіленої системи, сумісного виконання завдань компонентами системи, зміні конфігурації та архітектури децентралізованої розподіленої системи на основі значення станів кожної компоненти від початку поточного запуску, часу перебування в кожному стані кожної компоненти, рівнів безпеки в кожному стані кожної компоненти, які задано відповідно матрицями W, T, $P_{s,ДРС}$. Для визначення рівня безпеки системи здійснюють його обчислення.

UA 135205 U

Корисна модель належить до інформаційних технологій та кібербезпеки і може використовуватись для організації взаємодії компонентів децентралізованих розподілених програмних систем виявлення та локалізації зловмисного програмного забезпечення (ЗПЗ) у локальних комп'ютерних мережах.

Мережне антивірусне програмне забезпечення є спеціальним типом антивірусного програмного забезпечення, що призначене для використання у комп'ютерних мережах. Воно переважно використовується разом із засобами антивірусного захисту вузлів мережі (робочих станцій і серверів) як другий рівень захисту для підвищення ймовірності виявлення і блокування зловмисного програмного забезпечення. Якщо на одному рівні захисту атаку чи ЗПЗ не буде виявлено, тоді зберігається ймовірність його блокування на іншому рівні захисту. Важливими елементами організації таких мережних програмних засобів є способи взаємодії компонентів таких розподілених систем.

У мережних антивірусах ESET® Endpoint Security для Windows система захисту кінцевих точок у корпоративних мережах [1] містить антивірусне програмне забезпечення, контроль над встановленими додатками, контроль за веб-трафіком і контроль пристроїв, що підключаються. Спосіб управління всіма функціями та компонентами системи базується на використанні єдиного центру ESET Security Management Center, який інсталюється на операційних системах Windows або Linux.

Недоліком такого способу є надмірна централізація, яка може бути використана зловмисниками шляхом атаки на виявлений центр.

Мережним антивірусом від компанії "Dr.Web" є спеціалізоване рішення "Dr.Web CureNet! ". В основу способу, згідно з яким він побудований, закладено централізацію для страховки і посилення безпеки мереж, в яких використовується антивірус інших виробників [2].

Недоліком такого антивірусу є спосіб, при якому наявність на робочих станціях мережної компоненти є не постійною. Це не дозволяє використати можливості решти робочих станцій для виявлення ЗПЗ.

В мережному антивірусі Symantec Endpoint Protection адміністратор мережі забезпечується необхідними інструментами з розгортання антивірусної мережі, її моніторингу, а також з управління параметрами роботи антивірусних клієнтів на об'єктах, що захищаються [3]. В його основу організації роботи закладено спосіб, який базується на класичній клієнт-серверній архітектурі, що породжує один з основних його недоліків в організації роботи, який пов'язаний з нерівноправністю у підтримці різних операційних систем.

"Malwarebytes Endpoint Security" [4] - мережне антивірусне програмне забезпечення від компанії "Malwarebytes", що реалізує централізований локальний захист комп'ютерів в мережі, ґрунтується на багаторівневій технології, здатній розірвати ланцюг атаки. В основу закладено спосіб, який використовує централізацію і об'єднує технології виявлення і нейтралізації ЗПЗ в одному додатку.

Рішення мережного захисту від компанії "Cisco" включає технологію "Cisco® Network Admission Control (NAC)", яка базується на використанні двох способів: серверному і архітектурному [5]. Технологія NAC дозволяє адміністраторам мережі аналізувати і контролювати всі пристрої, що підключаються до мережі.

Для антивірусного захисту мережі в антивірусі Kaspersky Administration Kit [6] реалізовано принцип автономної роботи і прийняття рішень без участі адміністратора, якому видаються повідомлення про критичні ситуації. Він базується на централізованому способі організації взаємодії компонентів програмної системи. При цьому адміністратор може обмежити доступ користувачів до сумнівних джерел (зовнішніх носіїв, веб-сайтів певних категорій) та має можливість постфактум оцінити дії антивірусу, відкотити їх назад, або змінити налаштування, щоб в подальшому антивірус аналогічні інциденти обробляв інакше.

Мультиагентний спосіб локалізації бот-мереж в корпоративних комп'ютерних, мережах [7], який включає використання мультиагентної системи для здійснення комунікації між агентами для обміну інформацією між групами агентів для визначення рівня присутності бот-мереж в заданих корпоративних комп'ютерних мережах. Спосіб організації взаємозв'язку компонентів мультиагентної системи є централізованим.

Недоліками відомих способів організації взаємозв'язку компонентів розподілених систем для виявлення ЗПЗ в корпоративних комп'ютерних мережах є використання централізованої архітектури, що контролюється адміністратором. Це призводить до недостатньо високої достовірності виявлення і локалізації ЗПЗ, так як збір інформації про стан мережі, визначення присутності ЗПЗ та його блокування здійснюються для обробки єдиним центром, що може бути сповільнено через передачу зібраних даних цьому центру, обчислювальні ресурси, на яких він розміщений, а також вплив на його роботу адміністратора мережі.

Найбільш близькими аналогом до заявленого способу можна вважати мультиагентний спосіб локалізації бот-мереж у корпоративних комп'ютерних мережах [7], який базується на організації роботи мультиагентної системи з єдиним центром, а також спосіб виявлення метаморфних комп'ютерних вірусів в комп'ютерних системах на основі використання статистичних метрик для визначення еквівалентних функціональних програмних блоків описаний в [8] на основі залучення центром системи, який розміщений на сервері, обчислювальних ресурсів інших комп'ютерних систем локальної мережі для підвищення достовірності виявлення метаморфних вірусів, і спосіб описаний в [6], особливістю якого є організація централізованої роботи антивіруса з прийняття рішень на основі роботи його автономних компонентів.

Відомі мережні рішення показують недостатньо високу достовірність виявлення нового зловмисного програмного забезпечення в локальних комп'ютерних мережах, зокрема, через використання централізованого способу організації взаємодії компонентів розподіленої системи.

В основу корисної моделі поставлена задача підвищення достовірності виявлення зловмисного програмного забезпечення у комп'ютерних системах локальних мереж на основі ефективної взаємодії компонентів децентралізованої розподіленої системи його виявлення.

Поставлена задача вирішується тим, що у способі організації взаємодії компонентів децентралізованих розподілених систем виявлення зловмисного програмного забезпечення на основі рівнів їх безпеки в локальних комп'ютерних мережах, згідно з корисною моделлю, для функціонування таких децентралізованих розподілених систем взаємна координація роботи їх компонентів між собою в локальній мережі враховує рівень достовірності виявлення зловмисного програмного забезпечення в конкретних комп'ютерних системах мережі з використанням заданого порядку взаємодії компонентів системи, який включає визначення станів програмних модулів через рівні їх безпеки, обробку відповідей від кожної компоненти іншими компонентами на відправлені пакети, обробку компонентами системи невизначеностей, пов'язаних з відсутністю відповідей на відправлені пакети в мережі з використанням сканування заданих портів комп'ютерних систем, оцінку стану компонент системи та перевіряння цих оцінок між собою всіма компонентами, визначення стану децентралізованої розподіленої системи на основі обчислення рівня її безпеки, прийняття рішення про подальшу роботу системи в цілому на основі дослідження її стану кожним її компонентом окремо, вплив подій, які активують методи виявлення зловмисного програмного забезпечення, на зміну стану окремих компонентів та системи, здійснення дослідження інших комп'ютерних систем компонентами системи на наявність подібних проявів зловмисного програмного забезпечення та обмін такими отриманими результатами, обробку та оптимізацію статистичних даних, накопичених в системі кожною компонентою окремо, обміном знаннями всередині децентралізованої розподіленої системи, сумісного виконання завдань компонентами системи, зміні конфігурації та архітектури децентралізованої розподіленої системи на основі значення станів кожної компоненти від початку поточного запуску, часу перебування в кожному стані кожної компоненти, рівнів безпеки в кожному стані кожної компоненти, які задано відповідно матрицями W, T, $P_{s,ДРС}$ і для визначення рівня безпеки системи здійснюють його обчислення за формулою 1, яка впливає на динамічну зміну архітектури системи протягом часу її функціонування:

$$R_{b,ДРС,2}=\frac{1}{4}*\left(\Sigma_{s=1}^{m}\left(1-\prod_{p_{s,j}<1}^{n}(1-p_{s,j})\right)*k_s+\Sigma_{s=1}^{m}\Sigma_{w_{s,j}>0}^{n}\left(\frac{t_{s,j}}{\Sigma_{s=1}^{m}t_{s,j}}*\left(\frac{w_{s,j}}{\Sigma_{s=1}^{m}w_{s,j}}\right)+\Sigma_{s=1}^{m}\left((1+k_s)*\frac{\Sigma_{j=1}^{n}w_{s,j}}{\Sigma_{s=1}^{m}\Sigma_{j=1}^{n}w_{s,j}}*\frac{\Sigma_{j=1}^{n}t_{s,j}}{\Sigma_{s=1}^{m}\Sigma_{j=1}^{n}t_{s,j}}\right)\right)\right), \quad (1)$$

де $R_{b,ДРС,2}$ - рівень безпеки децентралізованої розподіленої системи, визначений на другому етапі, b - позначення безпеки, s - номер компоненти (програмного модуля) децентралізованої розподіленої системи, n - кількість компонент системи, m - кількість станів програмного модуля, k_s - коефіцієнт загрози бути ураженим зловмисним програмним забезпеченням s - того стану програмного модуля, значення якого встановлюється з відрізку [0;1] в залежності від того, які функційні навантаження закладено у певний s-ий стан, $p_{s,j}$ - ймовірність бути ураженим зловмисним програмним забезпеченням, значення $p_{s,j}$ -

отримуються на основі результатів функціонування закладених в програмні модулі підсистем виявлення певних типів зловмисного програмного забезпечення, $w_{s,j}$ - кількість перебувань програмного модуля з номером j в стані $s, j = 1,2,\ldots,n, s = 1,2,\ldots m$, $t_{s,j}$ - сумарний час перебування програмного модуля з номером j в стані s.

Для підвищення достовірності та ефективності виявлення нового ЗПЗ пропонується в розподілених системах в локальних мережах використовувати децентралізацію системи і для координації компонентів системи застосувати спосіб взаємодії компонентів розподіленої системи виявлення ЗПЗ, що встановлює порядок здійснення комунікації між компонентами системи та обміну знаннями між ними на основі рівнів безпеки системи, які динамічно визначаються в певні моменти часу. Він застосовуватиметься для вирішення задач верхнього рівня організації взаємодії, тобто тільки для організації взаємодії компонентів системи і представлення її цілісною. Для вирішення проблеми з безпосереднього виявлення ЗПЗ в локальних обчислювальних мережах застосовуватимуться методи, які належатимуть до нижчого рівня системи, що включатимуть архітектурні особливості розподіленої системи і технології виявлення ЗПЗ.

Корисна модель пояснюється кресленнями. На Фіг. 1 представлена узагальнена схема основних компонентів розподіленої системи в локальних комп'ютерних мережах. Архітектура децентралізованої розподіленої системи (ДРС) виявлення ЗПЗ представлена в [9]. Схема застосування методів та їх взаємозв'язку зображена на Фіг. 2.

Розроблений спосіб взаємодії компонентів децентралізованих розподілених систем виявлення ЗПЗ в локальних комп'ютерних мережах встановлює правила взаємодії компонентів розподілених систем через зв'язуючу їх програмну частину, тобто описує порядок роботи зв'язуючого програмного забезпечення, яке виконуватиметься автономно протягом тривалого часу. Застосування способу розпочинається після успішного запуску програмного модуля ДРС в конкретній комп'ютерній системі, тобто з програмного забезпечення програмного модуля (ПМ) ДРС запущена та частина, яка належить і відповідає за запуск тільки в комп'ютерних системах (КС). Ця запущена і активна функція є стартовою, в задачі якої входить активація програмного модуля ДРС в КС і подальший запуск з неї функцій, що реалізують збірку ДРС в мережі. Для розгляду основних кроків методу вважатимемо, що не менше, ніж в двох КС мережі стартова функція виконалась успішно. Крім того, вважатимемо, що програмний модуль вже було інстальовано раніше успішно і цей запуск не є інсталяційним. Тоді, подальші процеси, що протікатимуть в мережі, які пов'язані з функціонуванням системи, представимо такими кроками способу взаємодії компонентів децентралізованої розподіленої системи:

1) визначення станів програмних модулів;

2) обробка відповідей від ПМ КС на відправлені пакети;

3) обробка програмним модулем невизначеностей, пов'язаних з відсутністю відповідей на відправлені пакети;

4) сканування заданого порту КС;

5) оцінка стану програмного модуля та її перевіряння між рештою ПМ ДРС на етапі обміну повідомленнями;

6) визначення стану децентралізованої розподіленої системи;

7) прийняття рішення про подальшу роботу ДРС в цілому на основі дослідження її стану програмними модулями;

8) вилучення активного програмного модуля з ДРС в результаті вимкнення КС;

9) події, які активують методи виявлення зловмисного програмного забезпечення, впливають на зміну стану ПМ ДРС; здійснення дослідження інших КС на наявність подібних активностей та обмін отриманими результатами;.

10) обробка та оптимізація статистичних даних, накопичених в системі кожним модулем окремо;

11) обмін знаннями всередині децентралізованої розподіленої системи;

12) сумісне виконання завдань компонентами ДРС;

13) робота ДРС в складі всього одного програмного модуля;

14) поповнення ДРС новими модулями.

Визначення станів програмних модулів ДРС. Визначення стану кожного програмного модуля в КС, на яких встановлена ДРС на етапах початку роботи ПМ при його первинній інсталяції, при щоденному завантаженні КС, в процесі функціонування КС та при завершенні роботи КС. Визначення стану програмного модуля в КС включає перевірку КС, її програмного забезпечення та безпосередньо рівня активності самого ПМ. Занесення інформації у внутрішні бази кожного програмного модуля системи. Порівняння результатів сканування КС з попередніми

результатами сканувань за певний період, які зберігаються в базі сканувань. Якщо результати сканування не співпадають з попередніми, то КС блокується і видається відповідне повідомлення на екран. Якщо основні параметри сканувань співпадають, тоді ПМ продовжує роботу. Підготовка і формування пакета повідомлення про свій стан кожним модулем системи. Занесення повідомлення пакета у базу повідомлень ПМ (відправлених та отриманих). Надсилання пакетів повідомлень про свій стан на всю решту частин ДРС згідно з заданим реєстром програмних модулів системи і комп'ютерних систем, в яких вони розміщені. Збереження інформації про події за номерами та адресами програмних модулів ДРС, які занесені в базу програмних модулів решти КС при інсталяції.

Обробка відповідей від ПМ КС на відправлені пакети. Отримання відповідей на надіслане повідомлення про стан програмного модуля. Якщо надісланий пакет було доставлено успішно, то про це надсилається відповідна відповідь, причому вона обов'язково надходить окремо після відправленого першого пакета від всіх компонентів системи. Очікування відповіді програмним модулем системи відбувається за заданим інтервалом часу, який розраховується при первинній інсталяції системи та враховує технічні можливості мережі по швидкості передачі пакетів, а також можуть бути введені певні критерії, виконання яких вказує на необхідність очікування повідомлення на надісланий пакет, а не перехід на наступний крок. Врахування вимкнених КС, які містять ПМ ДРС і в яких ПМ не активні, для оцінки стану цілісності ДРС та її структури в певні моменти часу. Отримання відповідей від кожного, зареєстрованого в ДРС, програмного модуля з решти КС про успішне отримання ними пакета з повідомленням про свій стан від усіх ПМ ДРС. Проведення аналізу отриманих відповідей від ПМ решти КС та аналізу від яких КС отримано відповіді, а від яких не отримано. Обробка відповіді про успішне отримання пакету з повідомленням про стан програмного модуля, який надіслав це повідомлення всім решті ПМ з ДРС, від певної кількості ПМ та визначення тих, від яких не отримано ніякої відповіді. Обробка події, яка полягає у не отриманні відповіді від жодного ПМ системи у встановлені часові вимоги. Занесення отриманої інформації до бази повідомлень модуля. Формування або підтвердження цілісності ДРС з активних програмних модулів. На кожен пакет, який формується після ввімкнення КС і надсилається на решту КС, обов'язковою є відповідь про його отримання та включення ПМ до реєстру активних ПМ ДРС.

Обробка програмним модулем невизначеностей, пов'язаних з відсутністю відповідей на відправлені пакети. Якщо відповідь на відправлений пакет із заданого програмного модуля певної КС не отримано протягом певного інтервалу часу або визначення такого факту за іншими критеріями, тоді здійснити сканування необхідного порту заданої комп'ютерної системи. Якщо отримано відповідь, що пакет не доставлено через збої в системі передачі, тоді повторити надсилання пакета на визначену КС.

Сканування заданого порту КС. Номер порту, через який буде здійснюватись обмін повідомленнями між програмними модулями визначено при налаштуванні під час встановлення програмного модуля в КС. Також, додатково може бути встановлено ще декілька портів як резервних для підвищення живучості системи. Якщо сканування порту успішне, тобто порт є доступним, тоді робиться відмітка в базі відправлених та отриманих повідомлень, і ПМ по цій КС переходить в стан очікування пакета від неї. Інакше, тобто якщо сканування неуспішне і показує, що порт недоступний, тобто, що порт закрито, тоді робиться відмітка в базі відправлених та отриманих повідомлень, і ПМ по цій КС переходить в стан очікування пакета від неї. Якщо досліджувана КС вимкнена, то в цьому випадку здійснити почергово сканування резервних портів. Якщо по скануванню відповідь негативна, тоді ПМ переходить до стану очікування результатів від такої КС, інакше робиться перевіряння такого результату з іншими активними ПМ, які вже сформували ДРС.

Оцінка стану програмного модуля та її перевіряння між рештою ПМ ДРС на етапі обміну повідомленнями. Обробка подій для ПМ, який надсилав пакети на всі КС: за відправленим пакетом відповіді не отримано; після сканування порту відповіді у вигляді результату не отримано або отримано, що відкритий. Тоді ПМ здійснює перевіряння свого результату з результатами решти КС. Для цього формується і задається запит, у вигляді пакета, решті програмних модулів, крім досліджуваного, які повинні надіслати підтвердження про свою роботу, про стан того досліджуваного програмного модуля. Для здійснення дослідження за вказівкою одного вибраного ПМ ДРС, решта ПМ надсилають йому по одному пакету про свій стан та обробляють відповіді від нього. Надсилання результатів дослідження вибраного програмного модуля запитуваному модулю. Обробка результатів дослідження вибраного програмного модуля. Якщо програмні модулі отримали таку ж відповідь від досліджуваного модуля, як і програмний модуль, що активізував цю подію перевірки, тоді всі ПМ ДРС вважають, що досліджуваний модуль поки не активний і продовжують очікувати пакета від нього, коли КС

ввімкнеться. Досліджувана КС може бути виключена, тоді всі ПМ отримають однакову відповідь. Якщо програмні модулі отримали різні відповіді від досліджуваного модуля або певна частина отримала, а інша не отримала, або інші відповіді, як і програмний модуль, що активізував цю подію перевірки, тоді всі програмні модулі повідомляють адміністратору про цю подію, видають повідомлення на екран своєї КС, записують в свій реєстр позаштатних ситуацій та зменшують ДРС на один програмний модуль.

Визначення стану дегентривізованої розподіленої системи. Після певного часу, встановленого адміністратором, програмні модулі визначають стан ДРС з певною періодичністю за часом або за настанням критичних подій в КС. Стан ДРС виразимо через рівень його безпеки. Кожен ПМ окремо після старту визначає свій власний стан і в подальшій роботі змінює його в залежності від виконуваних функцій. Підготовка і формування пакета з повідомленням про свій стан кожним ПМ системи і надсилання його від кожного ПМ решті активних ПМ. Проведення підтвердження про успішну доставку пакета кожним ПМ від решти ПМ.

Для визначення стану безпеки ДРС використаємо дані в поточний момент часу з її програмних модулів: стан кожного ПМ від початку поточного запуску, часу перебування в кожному стані кожного ПМ, рівні безпеки в кожному стані кожного ПМ. Результати задамо відповідно матрицями: W, T, $P_{s,ДРС}$. Обчислення за формулами (1) і (2) стану ДРС кожним ПМ на основі отриманих даних зі всіх активних ПМ ДРС здійснюємо в два етапи. На першому етапі рівень безпеки ДРС визначимо за формулою 1:

$$R_{b,ДРС,1} = \frac{\sum_{l=1}^{n}(1 - \sum_{s=1}^{m} k_{s,l} * p_{s,l})}{n}, \quad (1)$$

де $R_{b,ДРС,1}$ - рівень безпеки ДРС, визначений на першому етапі, b - позначення безпеки, l - номер програмного модуля ДРС, n - кількість програмних модулів ДРС, $k_{s,l}$ - коефіцієнт загрози (таблиця, Фіг. 3) бути ураженим ЗПЗ s - того стану ПМ, значення якого встановлюється з відрізку $[0;1]$ в залежності від того, які функційні навантаження закладено у певний s-ий стан, $p_{s,l}$ - ймовірність бути ураженим ЗПЗ, m - кількість станів ПМ.

За формулою 2 ДРС здійснює визначення свого центру в поточний момент, а також на основі цього значення здійснюється виділення критичних ПМ.

$$g(R_{b,ДРС,1},k,s,s_{c,ДРС}) = \begin{cases} 0, \text{якщо виконується умова 1} \\ 1, \text{якщо виконується умова 2} \\ 2, \text{якщо виконується умова 3} \end{cases}, \quad (2)$$

де $g(R_{b,ДРС,1},k,s,s_{c,ДРС})$ - функція визначення подальших кроків для ДРС, $R_{b,ДРС,1}$ - рівень безпеки ДРС, який отримано на першому етапі за формулою 1, k - кількість активних ПМ із загальної кількості n, s - номер стану, $s = 1,2,...,m, m$ - кількість станів ПМ, $s_{c,ДРС}$ - середнє значення для ДРС на основі сукупності станів її ПМ. Умови для задання функції g представимо в Таблиці (Фіг. 4). Загальна кількість таких випадків може бути 64, так як є чотири випадки для рівня безпеки, два випадки для кількості ПМ, які входять до центру ДРС в поточний момент часу, вісім для віднесення центру до одного зі станів за рахунок дослідження його відхилення.

При виконанні умови 1, тобто якщо $g(R_{b,ДРС,1},k,s,s_{c,ДРС}) = 0$, то ДРС продовжує роботу в режимі, коли її ПМ працюють у тих станах, в яких були. При цьому жодних дій по обробці ситуацій в певних відібраних КС не проводиться.

При виконанні умови 2, тобто якщо $g(R_{b,ДРС,1},k,s,s_{c,ДРС}) = 1$, то ДРС продовжує роботу в режимі коли її ПМ працюють у тих станах, в яких були. А також, ДРС зразу відмічає програмні модулі, для яких потрібне додаткове уточнення стосовно задач, які виконують в поточний момент часу.

При виконанні умови 3, тобто якщо $g(R_{b,ДРС,1},k,s,s_{c,ДРС}) = 2$, то ДРС переходить до другого етапу уточнення свого стану на основі залучення часових характеристик станів всіх ПМ.

Якщо ймовірності бути ураженим ЗПЗ впливатимуть не тільки на програмні модулі або їх вплив на ці модулі несуттєвий, то це не дозволяє визначити стан ДРС як критичний. Такий випадок можливий, коли дослідження на першому етапі через визначення середнього значення і його подальшого використання було невисоким через невеликий час роботи від останнього запуску ПМ чи через необхідність його усереднення на вісім станів. Але може виявитись, що багато програмних модулів тривалий час перебувають чи перебували в одному і тому ж стані, а використання критеріїв першого етапу їх не виділяє. Тому, щоб врахувати такі граничні

особливості, виділимо ймовірності бути ураженим ЗПЗ для ДРС в певних визначених станах і здійснюємо оцінку таких випадків на другому етапі дослідження. $t_{s,j}$

Для другого етапу визначення стану ДРС узагальнена формула 3 визначення рівня безпеки матиме вигляд:

$$R_{b,ДРС,2} = \frac{1}{4} * \left(\sum_{s=1}^{m} \left(1 - \prod_{\substack{j=1 \\ p_{s,j}<1}}^{n} (1 - p_{s,j}) \right) * k_s + \sum_{\substack{j=1 \\ t_{s,j}>0 \\ w_{s,j}>0}}^{m} \left(\frac{t_{s,j}}{\sum_{s=1}^{m} t_{s,j}} * \frac{w_{s,j}}{\sum_{s=1}^{m} w_{s,j}} + \sum_{s=1}^{m} \left((1 + k_s) * \frac{\sum_{j=1}^{n} w_{s,j}}{\sum_{s=1}^{m} \sum_{j=1}^{n} w_{s,j}} * \frac{\sum_{j=1}^{n} t_{s,j}}{\sum_{s=1}^{m} \sum_{j=1}^{n} t_{s,j}} \right) \right) \right), \quad (3)$$

де $R_{b,ДРС,2}$ - рівень безпеки ДРС, визначений на другому етапі, b - позначення безпеки, s - номер програмного модуля ДРС, n - кількість програмних модулів ДРС, m - кількість станів ПМ, k_s - коефіцієнт загрози бути ураженим ЗПЗ s - того стану ПМ, значення якого встановлюється з відрізку [0; 1] в залежності від того, які функційні навантаження закладено у певний s-ий стан, $p_{s,j}$ - ймовірність бути ураженим ЗПЗ, $w_{s,j}$ - кількість перебувань ПМ з номером j в стані $s, i = 1,2,\ldots,n, s = 1,2,\ldots m$, $t_{s,j}$ - сумарний час перебування ПМ з номером j в стані s n - кількість ПМ ДРС. Значення $p_{s,j}$ отримуються на основі результатів функціонування закладених в програмні модулі підсистем виявлення певних типів ЗПЗ.

Обмін пакетами з повідомленнями про стан ДРС від кожного ПМ з рештою ПМ системи. Аналіз і обробка отриманих результатів кожними ПМ ДРС. Якщо всі ПМ обчислили стан ДРС однаково, тоді система продовжує роботу. Результати перевіріння про стан ДРС співпадають, тоді надсилається відповідне повідомлення від кожного програмного модуля всім решті модулів. Ця інформація заноситься у внутрішній реєстр подій. Якщо виявиться, що хоча б один з програмних модулів відповість всім решті, що він отримав від певного модуля результат відмінний від свого і їх, тоді він вказує в повідомленні для всіх програмних модулів номер того програмного модуля і його результат, який відмінний від загального результату. В цій ситуації всі програмні модулі, крім того, в якого відмінні від інших результати, відправляють йому команди для блокування КС та виведення повідомлення про причину блокування на екран, а самі блокують надходження будь-яких пакетів від нього і вилучають його з реєстру модулів системи. ДРС продовжить роботу, але кожен ПМ відобразить на екран КС повідомлення про номер та стан ПМ, якого вилучено. Якщо виявиться, що програмний модуль отримав від усіх однакове значення стану системи, але воно не співпадає з розрахованим ним і при цьому він надіслав своє значення всім, тоді цей програмний модуль блокує КС та видає відвідне ситуації повідомлення на екран. Якщо від частини ПМ ДРС пакети на надійшли, тоді здійснення запуску процедури визначення причини не отримання повідомлень. Після обробки таких подій продовження роботи ПМ.

Прийняття рішення про подальшу роботу ДРС в цілому на основі дослідження її стану програмними модулями. Визначення рівня кожного системи на основі рівнів її ПМ, розподіл їх за групами ризику бути ураженими, тобто рівнями визначених загроз, та прийняття рішення про подальшу роботу системи на основі визначення результатів за функцією 3 і Рис. 4. Якщо стан ДРС, визначено як такий, що ступінь безпеки складає 0-30 %, тоді здійснити блокування ПМ всіх КС і повідомити адміністратору. Така подія може відбутись також за умови наявності в системі великої кількості ПМ, які перебувають на рівнях 2 або 3 тривалий час. Якщо стан ДРС, визначений як такий, що рівень безпеки складає 30-75 %, тоді здійснити блокування тільки тих КС, ПМ яких вказує на віднесення КС до рівня 0-30 %, повідомити адміністратору і перерахувати стан системи на ту кількість КС які залишились. Якщо стан ДРС, визначений як такий, що рівень безпеки складає 75-100 %, тоді дослідити ті КС, в яких рівень безпеки складає менше 75 % тривалий час. Якщо перевищений ліміт часу вичерпано на подолання загрози, тоді здійснити блокування ПМ тієї КС, повідомити адміністратору і продовжити роботу без вилученого ПМ. Якщо стан ДРС, визначений як такий, що рівень безпеки складає 75-100 % і після дослідження тих КС, в яких рівень безпеки складає менше 75 % тривалий час, їх не виявлено, то продовжити роботу.

Вилучення активного програмного з ДРС в результаті вимкнення КС. Якщо одна з КС вимикається, тоді про це її програмний модуль повідомляє решту модулів і тільки після цього відбувається вимкнення.

Події, які активують методи виявлення зловмисного програмного забезпечення, впливають на зміну стану ПМ ДРС. Здійснення дослідження інших КС на наявність подібних активностей та обмін отриманими результатами. При переході ПМ на рівень 2 застосовується метод виявлення файлового зловмисного програмного забезпечення, на основі агентного підходу і нечіткого висновку та методу, на основі розподілу доступу в мережі і залучення додаткових обчислювальних компонентів мережі. При переході програмного модуля на рівень 3, застосовується метод виявлення бот-мереж або метод виявлення експлоїтів.

Обробка та оптимізація статистичних даних, накопичених в системі кожним модулем окремо. При невеликій навантаженості КС і відсутності інших завдань, які пов'язані з необхідністю перебування програмного модуля на рівні 2 або 3, та тривале за часом перебування в стані 1 ПМ переходить до рівня 4 та здійснює дослідження накопичених статистичних даних. Зокрема, він відслідковує та аналізує порядок запуску ПМ, в порівнянні з іншими програмами КС та час його старту за певний період часу. Обробка таких даних включає обчислення основних статистичних параметрів: визначення середнього значення, дисперсії та середнього квадратичного відхилення. Здійснення оптимізації накопичених статистичних даних в базах програмних модулів. Якщо відбулось обчислення і виявлено значне відхилення, тоді оптимізація даних не здійснюється, блокується КС та видається повідомлення адміністратору.

Обмін знаннями всередині розподіленої багаторівневої системи. Отримані результати одним програмним модулем ДРС, які стосуються виявлення і локалізації ЗПЗ, формуються в пакет і надсилаються іншим ПМ в мережі, які застосовують ці результати для перевірки своїх КС.

Сумісне виконання завдань компонентами ДРС. Колективне виконання завдань, пов'язаних з виявленням ЗПЗ шляхом здійснення збільшення обчислювальних ресурсів для програмного модуля шляхом відправки частини задач на інші КС для дослідження ЗПЗ, в якому є підозрілі поведінки. Зокрема, залучення в ПМ інших КС емуляторів роботи процесора для спонукання до проявів ЗПЗ. Підготовка і надсилання іншим ПМ результатів, які отримані.

Робота ДРС в складі всього одного програмного модуля. Якщо ДРС залишається в складі всього одного ПМ, у зв'язку з коректним завершенням роботи інших ПМ, тоді вона переходить до обмеженого використання своїх можливостей і може переходити з першого рівня на другий, в якому обмежені можливості. Після наступного нового запуску ДРС у цьому ПМ здійснюється обов'язкова перевірка його стану за умови, якщо він переходив до рівня 2, залишаючись одним в системі.

Таким чином, кожен останній працюючий ПМ при новому запуску ДРС так перевіряється.

Поповнення ДРС новими модулями. Комп'ютерні системи, які ввімкнуться пізніше, перебудують систему, розширивши її. Кожен програмний модуль КС здійснить розсилку пакетів іншим ПМ системи.

Використання корисної моделі дозволяє організовувати підтримку цілісності системи та здійснення передачі знань, отриманих окремими структурними компонентами децентралізованої розподіленої системи програмними модулями іншим компонентам. Розроблений спосіб є основою для розробки зв'язуючої частини програмного забезпечення децентралізованої розподіленої системи виявлення ЗПЗ в локальних комп'ютерних мережах на основі рівня її безпеки.

Джерела інформації:

1. Машинне навчання та знання людини в динамічній рівновазі. URL: https://eset.ua/ua/products/for_business/security/endpoint_security (дата звернення: 26.11.2018)

2. Доктор Веб. URL: https://curenet.drweb.ru/ (дата звернення: 26.11.2018)

3. Обзор Symantec Endpoint Protection 12. URL: https://www.anti-malware.ru/reviews/Symantec_Endpoint_Protection_12_2 (дата звернення: 26.11.2018)

4. Malwarebytes Endpoint Security. URL: https://ru.malware bytes.com/business/ endpoint security/ (дата звернення: 26.11.2018)

5. Решение по безопасности беспроводных сетей на базе Cisco Network Admission Control. URL: https://www.cisco.com/web/RU/products/hw/wireless/secure/cnac.html (дата звернення: 26.11.2018)

6. Антивирусная защита сети Kaspersky Administration Kit. URL: https://support.kaspersky.ru/learning/c wses/ki_102.80/intro/sectionl (дата звернення: 26.11.2018)

7. Патент України на корисну модель № 108238, МПК G06F 21/55 Мультиагентний спосіб локалізації бот-мереж у корпоративних комп'ютерних мережах / Поморова О.В., Савенко О.С., Крищук А.Ф., Лисенко С.М., Бобровнікова К.Ю., Нічепорук А.О.; власник - Хмельницький національний університет. - № u201600127; заявл. 04.01.2016; опубл. 11.07.2016, Бюл.№ 13/2016.

8. Патент України на корисну модель № 118456, МПК G06F 21/55 Спосіб виявлення метаморфних вірусів на основі статистичних метрик для визначення еквівалентних функціональних програмних блоків / Савенко О.С., Лисенко С.М., Бобровнікова К.Ю., Нічепорук А.О.,. Савенко Б.О.; власник - Хмельницький національний університет. - № u201701743; заявл. 23.02.2017; опубл. 10.08.2017, Бюл. № 15/2017.

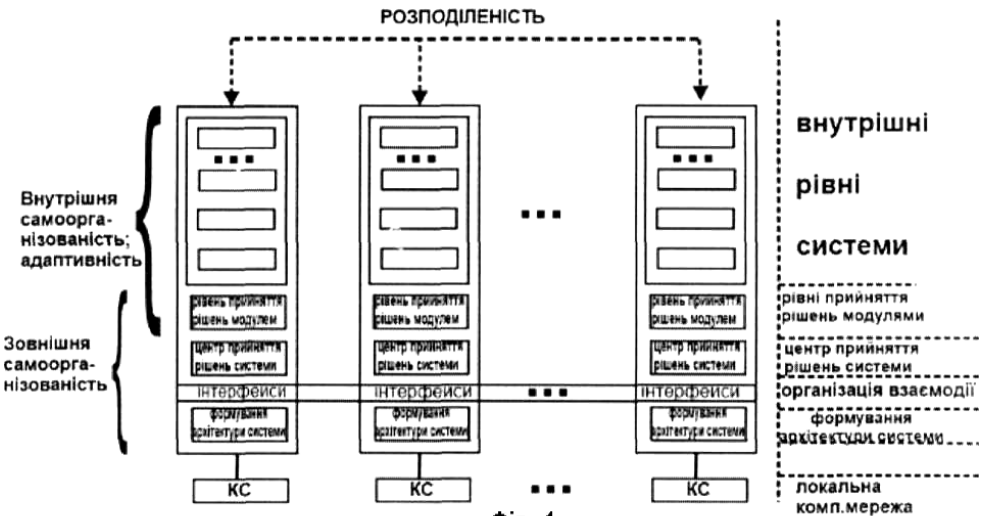
9. Markowsky G. Distributed Sy em for Detecting the Malware in LAN / G. Markowsky, O. Savenko, A. Sachenko // Proceedings of the 2018 IEEEE 13th International Scientific and Technical Conference on Computer Science and Information Technologies (CSIT), CSIT'2018, Lviv, Ukraine, September 11-14, 2018. - PP. 306-309.

ФОРМУЛА КОРИСНОЇ МОДЕЛІ

Спосіб організації взаємодії компонентів децентралізованих розподілених систем виявлення зловмисного програмного забезпечення на основі рівнів їх безпеки в локальних комп'ютерних мережах, який **відрізняється** тим, що для функціонування таких децентралізованих розподілених систем взаємна координація роботи їх компонентів між собою в локальній мережі враховує рівень достовірності виявлення зловмисного програмного забезпечення в конкретних комп'ютерних системах мережі з використанням заданого порядку взаємодії компонентів системи, який включає визначення станів програмних модулів через рівні їх безпеки, обробку відповідей від кожної компоненти іншими компонентами на відправлені пакети, обробку компонентами системи невизначеностей, пов'язаних з відсутністю відповідей на відправлені пакети в мережі з використанням сканування заданих портів комп'ютерних систем, оцінку стану компонент системи та перевіряння цих оцінок між собою всіма компонентами, визначення стану децентралізованої розподіленої системи на основі обчислення рівня її безпеки, прийняття рішення про подальшу роботу системи в цілому на основі дослідження її стану кожним її компонентом окремо, вплив подій, які активують методи виявлення зловмисного програмного забезпечення, на зміну стану окремих компонентів та системи, здійснення дослідження інших комп'ютерних систем компонентами системи на наявність подібних проявів зловмисного програмного забезпечення та обмін такими отриманими результатами, обробку та оптимізацію статистичних даних, накопичених в системі кожною компонентою окремо, обміном знаннями всередині децентралізованої розподіленої системи, сумісного виконання завдань компонентами системи, зміні конфігурації та архітектури децентралізованої розподіленої системи на основі значення станів кожної компоненти від початку поточного запуску, часу перебування в кожному стані кожної компоненти, рівнів безпеки в кожному стані кожної компоненти, які задано відповідно матрицями W, T, $P_{s,ДРС}$ і для визначення рівня безпеки системи здійснюють його обчислення за формулою 1, яка впливає на динаміку зміну архітектури системи протягом часу її функціонування:

$$R_{b,ДРС,2} = \frac{1}{4} * \left(\frac{\sum_{s=1}^{m} \left(1 - \prod_{\substack{j=1 \\ p_{s,j}<1}}^{n} (1 - p_{s,j}) \right) * k_s}{m} + \sum_{\substack{i=1 \\ t_{s,j}>0 \\ w_{s,j}>0}}^{n} \sum_{\substack{j=1 \\ t_{s,j}>0 \\ w_{s,j}>0}}^{m} \left(\frac{t_{s,j}}{\sum_{s=1}^{m} t_{s,j}} * \frac{w_{s,j}}{\sum_{s=1}^{m} w_{s,j}} + \sum_{s=1}^{m} \left((1 + k_s) * \frac{\sum_{j=1}^{n} w_{s,j}}{\sum_{s=1}^{n} \sum_{j=1}^{n} w_{s,j}} * \frac{\sum_{j=1}^{n} t_{s,j}}{\sum_{s=1}^{m} \sum_{j=1}^{n} t_{s,j}} \right) \right) \right), \quad (1)$$

де $R_{b,ДРС,2}$ - рівень безпеки децентралізованої розподіленої системи, визначений на другому етапі, b - позначення безпеки, s - номер компоненти (програмного модуля) децентралізованої розподіленої системи, n - кількість компонент системи, m - кількість станів програмного модуля, k_s - коефіцієнт загрози бути ураженим зловмисним програмним забезпеченням s-того стану програмного модуля, значення якого встановлюється з відрізку [0;1] в залежності від того, які функційні навантаження закладено у певний s-ий стан, $p_{s,j}$ - ймовірність бути ураженим зловмисним програмним забезпеченням, значення $p_{s,j}$ - отримуються на основі результатів функціонування закладених в програмні модулі підсистем виявлення певних типів зловмисного програмного забезпечення, $w_{s,j}$ - кількість перебувань програмного модуля з номером j в стані $s, j = 1,2,\ldots,n, s = 1,2,\ldots m,$ $t_{s,j}$ - сумарний час перебування програмного модуля з номером j в стані s.



Фіг. 1

1. Спосіб виявлення бот-мереж	...	1. Спосіб виявлення бот-мереж
2. Спосіб виявлення експлойтів	...	2. Спосіб виявлення експлойтів
3. Спосіб виявлення файлового ЗПЗ на основі агентного підходу і нечіткого висновку	...	3. Спосіб виявлення файлового ЗПЗ на основі агентного підходу і нечіткого висновку
4. Спосіб виявлення файлового ЗПЗ на основі розподілу в мережі і залучення додаткових обчислювальних компонентів мережі	...	4. Спосіб виявлення файлового ЗПЗ на основі розподілу в мережі і залучення додаткових обчислювальних компонентів мережі
Спосіб взаємодії компонетнів ДРС		
1	...	n
Комп'ютерні системи		

Фіг. 2

Рівні загроз станів	Стани ПМ, s для $m = 8$	Коефіцієнти загроз	Унормовані значення коефіцієнтів загроз, $k_{s,j}$
0	1, 8	0	0
1	2, 3	0,2	1/15
2	4	0,4	2/15
3	5	0,6	1/5
4	6	0,8	4/15
5	7	1	1/3

Фіг. 3

Умови	Значення умов для функції g		
для функції g	Значення рівня безпеки ДРС $R_{b,\text{ДРС},1}$	Умови, які пов'язані з кількістю ПМ	Умови, які пов'язані з середньоквадратичним відхиленням
Умова 1	$R_{b,\text{ДРС},1} > 0{,}75$	$\frac{k}{n} > 0$,	$\min(\frac{S_{c,\text{ДРС}}}{s} - 1) < 1$ для $s = 1$ або для $s = 8$
	$R_{b,\text{ДРС},1} > 0{,}75$	$\frac{k}{n} > 0$,	$\min(\frac{S_{c,\text{ДРС}}}{s} - 1) < 1$ для $s = 2$ або для $s = 3$ або s
	$R_{b,\text{ДРС},1} > 0{,}75$	$\frac{k}{n} < 0$,	$\min(\frac{S_{c,\text{ДРС}}}{s} - 1) < 1$ для $s = 5$ або для $s = 6$ або s
	$0{,}5 < R_{b,\text{ДРС},1} <$	$\frac{k}{n} < 0$,	$\min(\frac{S_{c,\text{ДРС}}}{s} - 1) < 1$ для $s = 2$ або для $s = 3$
Умова 2	$0{,}5 < R_{b,\text{ДРС},1} <$	$\frac{k}{n} > 0$,	$\min(\frac{S_{c,\text{ДРС}}}{s} - 1) < 1$ для $s = 2$ або для $s = 3$
	$0{,}5 < R_{b,\text{ДРС},1} <$	$\frac{k}{n} < 0$,	$\min(\frac{S_{c,\text{ДРС}}}{s} - 1) < 1$ для $s = 2$ або для $s = 3$ або s
	$0{,}25 < R_{b,\text{ДРС},1}$	$\frac{k}{n} > 0$,	$\min(\frac{S_{c,\text{ДРС}}}{s} - 1) < 1$ для $s = 1$ або для $s = 8$
Умова 3	$R_{b,\text{ДРС},1} > 0{,}75$	$\frac{k}{n} > 0$,	$\min(\frac{S_{c,\text{ДРС}}}{s} - 1) < 1$ для $s = 6$ або для $s = 7$
	$0{,}5 < R_{b,\text{ДРС},1} <$	$\frac{k}{n} > 0$,	$\min(\frac{S_{c,\text{ДРС}}}{s} - 1) < 1$ для $s = 5$ або для $s = 6$ або д
	$0{,}25 < R_{b,\text{ДРС},1}$	$\frac{k}{n} > 0$,	$\min(\frac{S_{c,\text{ДРС}}}{s} - 1) < 1$ для $s = 4$ або для $s = 5$ або д
	$0{,}25 < R_{b,\text{ДРС},1}$	$\frac{k}{n} < 0$,	$\min(\frac{S_{c,\text{ДРС}}}{s} - 1) < 1$ для $s = 5$ або для $s = 6$ або д
	$R_{b,\text{ДРС},1} < 0{,}25$	-	-
	решта випадків		

Фіг. 4

Комп'ютерна верстка Л. Бурлак

Міністерство економічного розвитку і торгівлі України, вул. М. Грушевського, 12/2, м. Київ, 01008, Україна

ДП "Український інститут інтелектуальної власності", вул. Глазунова, 1, м. Київ – 42, 01601