УДК 004.021
DOI: 10.31673/2409-7292.2022.030617

Лисинчук В.В.

ІДЕНТИФІКАЦІЯ СТАНУ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ ПРИСТРОЇВ IoT НА БАЗІ ОБРОБКИ ЧАСОВИХ РЯДІВ

У статті описується використання часових рядів для математичного опису стану захищеності пристроїв в мережі IoT. Аналізуються методи аналізу даних часових рядів з метою отримання значимих статистик та інших характеристик даних.

Ключові слова: IoT, часовий ряд, методи аналізу даних, аналіз, інформаційна безпека, системи, мережі.

Вступ та постановка проблеми

В концепції інтернету речей (Internet of Things, IoT) моніторинг стану інформаційної безпеки пристроїв обумовлений швидким розвитком різного виду інформаційних загроз, які змушують до постійного розвитку та вдосконалення їх методів пошуку та ідентифікації. Процеси уніфікації, стандартизації та типізації значно полегшують процеси пошуку та ідентифікації загроз інформаційної безпеки. Виникає необхідність розробки універсальних моделей та методів ідентифікації стану інформаційної безпеки систем з малопотужними обчислювальними пристроями. Системи мають використовувати комплексний підхід до аналізу даних, що надходять від внутрішніх та зовнішніх (сторонніх) інформаційних каналів. Це в свою чергу обумовлено тим, що вони отримують інформацію від багатьох пристроїв, збирає великі дані різних форматів від великої кількості джерел з неоднорідними характеристиками.

Метою публікації є розгляд методу ідентифікації стану інформаційної безпеки пристроїв IoT на базі обробки часових рядів.

Виклад основного матеріалу

Сучасний інтернет речей являє собою концепцію організованої обчислювальної мережі фізичних предметів, що взаємодіють один з одним та зовнішнім середовищем. Створення різних систем захисту інформації (СЗІ) та моніторингу стану інформаційної безпеки (ІБ), спрямованих на протидію загроз IoT, зумовлені використанням різних технологій вимірювання, ідентифікації, передачі, зберігання та обробки даних.

Комплексний підхід до організації даних є дуже важливим. Основою в даному питанні є реалізація систем захисту пристроїв, систем контролю взаємодії та систем ідентифікації стану. Тому можна заявити, що реалізація СЗІ та моніторингу стану ІБ інформаційно-телекомунікаційних систем будуються на забезпеченні роботи вище указаних систем [1-2]. Однак подібні системи мають передбачати одночасне використання: нових і старих пристроїв, різних версій прошивок, апаратних і програмних рішень, протоколів сполучення та технологій збору, передачі, зберігання та аналізу даних.

Визначено три основні характеристики – комплексні знання (в результаті отримання інформації про об'єкт, у будь-якому місці та у будь-який час), надійна передача (за допомогою протоколів зв'язку, маршрутизації, шифрування, мережевої безпеки, з високою точністю та реального часу), інтелектуальна обробка (з урахуванням множини обчислень, нечіткого упізнання та інших технологій для аналізу та обробки). Відповідно до цих характеристик структура може бути розділена на три рівні – рівень сприйняття (Perception), мережевий рівень та прикладний рівень [5, 6]. Варто зазначити, що іноді використовується більше ніж трирівнева структура, але в цій статі буде описані проблеми саме трирівнева модель.

Проблема рівня сприйняття. Фізична безпека пристроїв сприйняття та безпека збору інформації – це основні проблеми безпеки на рівні сприйняття. Для більшості пристроїв такого рівня характерні такі проблеми: розгортання в необслуговуваному середовищі, різноманітність, простота, обмеження енергозабезпечення та слабка здатність до захисту безпеки. Ці фактори не дозволяють забезпечити уніфіковану систему захисту та є

вразливими. Через те, що бездротова сенсорна мережа є джерелом інформації, то ІБ на цьому рівні важлива.

Проблеми безпеки на цьому рівні включають: фізичне захоплення сенсорних вузлів, захоплення вузла шлюзу, витік інформації сенсора, загрози цілісності даних, виснаження енергозабезпечення, загрози перевантаження, атаки типу DoS, загрози маршрутизації встановленням в мережу нелегітимних сенсорів, та загрози копіювання вузла.

Проблеми ІБ мережевого рівня. Загрози ІБ теперішніх мереж зв'язку поширюються і на IoT, що побудований на них. До цих проблем відносяться: несанкціонований доступ, перехоплення даних, загрози конфіденційності та цілісності, атаки типу людина посередині, Dos-атаки, віруси, експлойти, мережеві черв'яки, руткіти тощо.

Складнощі забезпечення безпеки на цьому рівні обумовлені двома причинами: гетерогенний характер структури (різноманітність пристроїв та мережевих технологій) і великою кількістю об'єктів. Великі обсяги інформації, що збирає IoT, її різноманітність та неоднорідні характеристики створює ряд проблем. Внаслідок цього на мережевому рівні мають місце складніші проблеми безпеки, до них відносяться можливі проблеми масштабованості мережі, викликані малопередбачуваним обсягом передачі даних від великої кількості вузлів, що призводять до можливості здійснення атак DoS, DDoS.

Окрема увага приділяється вразливості програмного забезпечення (software vulnerabilities), що призводять до порушення ІБ після впровадження. Причинами програмної вразливості можуть бути: помилки розробників складного програмного забезпечення (ПЗ), помилки ядра програми, застосування незахищеного коду неповнота обробки винятків, використання необроблених масивів з можливістю їх переповнення зловмисником, помилки в обробці Big Data, web-уразливості, недостатня продуктивність або масштабованість програмного забезпечення.

Слід зазначити складність ПЗ. Ця проблема викликана великою різноманітністю апаратних платформ і операційних систем. Для проєктування ПЗ необхідно емулювати (англ. to emulate) поведінку пристроїв. Це обумовлює необхідність створення імітатора довкілля для серверів. Це зробити складно у зв’язку з обмеженнями в приладах (енергозабезпечення, продуктивність процесора, пам'ять). Необхідно уникнути сильного розходження між емулятором та приладом. Також для налагодженого робочого релізу додатка, необхідно провести повноцінне тестування. Воно включає: тестування навантаження, тестування продуктивності, комплексне тестування взаємодії модулів. Не слід забувати і про бекдори (backdoor - чорний хід) – це ділянки коду, внесені розробником, для подальшої можливості використання для перегляду даних, а у разі ОС віддаленого управління комп'ютером. Вони можуть бути як збоями в програмному забезпеченні, так і встановлені розробником ПЗ для тестування та управління.

Проблеми ІБ прикладного рівня. Інтеграція комп’ютерних технологій, технологій зв’язку та різних промислових галузей привело до широкого поширення IoT.

Крім порушень інформаційної безпеки (загрози повтору, підслуховування, спотворення інформації, розкриття інформації тощо), додатки стикаються з додатковими проблемами безпеки прикладного рівня: при використанні хмарних обчислень, обробці інформації, забезпеченні прав на інтелектуальну власність, захист приватності тощо.

Закордонні фахівці приділяють велику увагу науковим та експериментальним дослідженням у забезпеченні інформаційної безпеки. Наприклад, у роботі [3] говориться про те, що найбільший ризик безпеки можливий на нижньому рівні архітектури - на рівні сприйняття. При цьому зазначається, що високий рівень ризику, характерний деяким загрозам безпеці на інших рівнях архітектури також. У роботі [4] наводяться результати досліджень забезпечення безпеки приватних даних на прикладі «розумного дому».

Пристрої IoT вимагають контролю за їх роботою: оновлення програмного забезпечення, оптимізацію процесів функціонування протягом циклу функціонування тощо. Часто вбудовані системи безпеки не забезпечують надійного рівня безпеки. Це зумовлено

тим, що такі системи мають типові вразливості широковідомі зловмисникам, у яких не виникне труднощів з отриманням доступу до системи та реалізацією даних вразливостей [5].

Зазвичай у процесі реалізації та протягом життєвого циклу інформаційних систем і мереж виникає необхідність вдосконалення систем захисту, контролю та моніторингу ІБ пристроїв IoT, де, як один з напрямків, окрім внутрішніх систем, можуть використовуватися, наприклад, акустичні, електромагнітні та інші, безпосередньо не пов'язані з процесами, що відбуваються, побічні, сторонні канали, що містять інформацію про процеси в мережі IoT [6-8].

Рішення в таких системах зазвичай відштовхуються від того факту, що більшість елементів мають невеликі програмно-апаратні можливості та протягом життєвого циклу, в основному, не змінюють свій обмежений функціонал. Ці пристрої мають обмежений функціонал. Зазвичай вони запрограмовані виконувати певну послідовність дій, в результаті отриманих команд і зовнішніх подій, що відбуваються в інформаційній системі. На основі, цих даних, використовуючи машинне навчання, статистичний аналіз, можна знаючи шаблони функціонування обчислити нормальний стан, де функціонують заздалегідь визначені процеси, а також аномальний стан, пов'язаний з появою нестандартної активності (значень параметрів та характеристик), що свідчать про відхилення від норми.

Постановка задачі

Для формальної постановки та розв'язання задачі в роботі введено позначення, представлені в таблиці 1.

Таблиця 1

Позначення для формалізації задачі

Позначення	Фізичне значення
S	Числові послідовності (часові ряди)
e	Момент часу
N, m	Граничні значення і індекси
H	Множина кортежних характеристик
Z	Множина станів пристрою
C	Множина класів стану
r	Відстань між об’єктами
q	Кількість каналів джерел
u	Зовнішня середа
h	Перехідна характеристика
v	Шумова складова
k	Кількість входів
d	Кількість виходів
i	Канал, що реєструється
j	Канал, що реєструє
X	Вектор
ε	Порогове значення
R	Простір спостереження
I	Кількість груп
μ	Центроїд
w	Ваговий коефіцієнт

В більшості випадків системи та пристрої являються закритими (представляють собою “чорну скриньку”). Це означає, що при розробці різні виробники використовують різні апаратні та програмні платформи, стандартні бібліотеки, де не завжди є можливість аналізувати вихідний код та апаратні прошивки. В більшості випадків пристрої IoT

реалізуються виробниками як закриті системи, де можливість вносити зміни апаратної або модифікації програмної частини відсутні.

Механізми захисту є внутрішніми. При перевірці цілісності у разі колізії запускаються механізми захисту (аж до припинення роботи пристрою). Проте, у процесі роботи, виникають необхідності конфігурації, налаштування, покращення характеристик функціонування системи. Через те, що більшість елементів системи використовують однакові алгоритми функціонування, стає можливим використання реверс-інжинірингу (англ. Reverse-engineering, укр. Зворотне проєктування), для пошуку вразливостей, спроби передачі широкомовних команд, здійснення різноманітних деструктивних впливів. Це в свою чергу веде до переводу пристрою в режим роботи, що супроводжується відхиленням параметрів від гранично допустимих значень, що суттєво впливає на робочий функціонал пристрою (включаючи вбудовані захисні механізми). У зв'язку з цим необхідно використовувати одночасно кілька інформаційних каналів, які можуть бути як сторонніми, так і внутрішніми.

З метою виявлення аномалій в роботі пристроїв IoT необхідно переглядати декілька дискретних станів. Це зумовлено тим, що як і в будь-якій системі, процеси пристроїв IoT протікають в динаміці, одночасно змінюється безліч параметрів, тому для відображення актуальної ситуації одного стану не достатньо.

Запропонований підхід пов'язаний з використанням шаблонів поведінки. Під шаблоном поведінки розуміються синхронізовані за часом множини послідовностей із визначеною частотою знімання значень з внутрішніх і зовнішніх датчиків, що реєструють параметри процесів, що відбуваються в пристрої.

Шаблон поведінки формується на основі інформації про функціонування пристрою IoT і його компонентів: загрузку і потреба ресурсів, дані про електромагнітні та звукові спектри, частоти та амплітуди, вібрації, температури тощо [9].

У визначені дискретні моменти часу в інформаційній системі реєструються значення вимірювальних пристроїв. Пристрої в момент часу $t = 1, \dots, N$ видають числову послідовність $\{S_0(t), S_1(t), \dots, S_m(t)\}$. Синхронізовані за часом і одержувані від різних контрольних елементів значення в дискретні моменти часу визначають кортежі характеристик $H = \{S(t) \mid t = 1, \dots, N\}$.

Задачу ідентифікації стану ІБ пристрою IoT можна представити таким чином.

Нехай Z – множина станів пристрою, C – множина класів станів, що містять як нормальні (безпечні) стани, де виконуються заздалегідь зумовлені процеси з показниками, що перебувають у нормі, так і аномальні (небезпечні) стани в яких існують відхилення від передбачуваних значень. Вибрано метрику відстані між об'єктами $r(z, z')$. Є кінцева навчальна вибірка відомих станів $\{z_1, \dots, z_l\} \in Z$, яку необхідно розбити на підмножини $c_0, c_1, \dots, c_p$, за метрикою відстані r і знайти алгоритм $a : Z \to C$, що відображає множину Z до множини C. Під метрикою мається на увазі функція або формула, що визначає відстань між будь-якими точками та класами в метричному просторі [10]. Таким чином, мета полягає в тому, щоб обробити інформацію сторонніх та внутрішніх джерел, на основі якої проводити ідентифікацію стану ІБ пристрою.

З обчислювальної точки зори пристрої IoT не мають великих ресурсів та володіють обмеженим набором виконуваних команд. Це дозволяє розглядати обмежений набір станів та їх переходів. Процеси приймання, обробки та передачі повідомлень, внутрішні ситуації, пов'язані з реалізацією обчислювальних алгоритмів, команди управління, що надходять, впливають на пристрій IoT. Ці процеси характеризуються перехідними характеристиками *h(t)* і станом зовнішнього середовища *u(t)*. В результаті отримуємо систему, яка має *k* входів і *d* виходів, де на вхід подається керуюча команда і значення змінних зовнішнього середовища, що визначають стан пристрою, а на виході елемента з'являються сигнали *S(t)* (про загрузку ресурсів, акустичні, електромагнітні тощо), що реєструються різними датчиками. Дані які надходять від зовнішніх каналів залежать від параметра шумової

складової *(v)t*, пов'язаного з властивостями вимірювального приладу, характеристиками сигналу, що одержується тощо.

У загальному випадку модель стану IoT-пристрою на основі інформації вимірюваних сигналів визначається наступним типом співвідношень:

$$\sum_{i=1}^{q}\sum_{j=1}^{d}\int_0^t u_i(t)h_{ij}(t-\tau)d_\tau = \sum_{j=1}^{d}\int_0^t f\left(s_j(t), \nu_j(t)\right), \quad (1)$$

де: q – кількість каналів джерел; h – перехідні характеристики *i*-го каналу для *j*-го каналу, що реєструє, одержувані каналом значення датчика; f – функція виміряних значень.

Ідентифікація стану пристрою IoT відбувається на основі даних у дискретні моменти часу $t_0, t_1, \ldots, t_n$ векторів числових послідовностей, що реєструються у процесі функціонування пристрою. Значення *(X)t* визначаються дискретною функцією, що відображає дані від датчиків, що містять суміш корисного сигналу *(S)t* і шуму, вираженого параметром *(v)t*:

$$X(t) = F[S(t), \nu(t)], \quad (2)$$

де вектор X є результатом змішаних взаємно незалежних сигналів *(S)t*, які мають спотворення шумової складової, яке складає *(v)t*.

Особливості процесів реєстрації даних дозволяють говорити про часовий ряд у поданні вектора X. Через обробку даних, що надходять від пристроїв, ідентифікується стан IoT. Стани визначаються множиною часових рядів у різних ситуаціях функціонування пристроїв IoT та поділяються на дві підмножини:

безпечні, де виконуються наперед визначені процеси;

небезпечні, де є відхилення від параметрів у заданих режимах роботи.

На основі навчальної вибірки визначаються початкові центроїди. Потім, у міру надходження аналізованих значень, відбувається обчислення відстані до найближчого центру кластера, аналіз та включення нового об'єкта в об'єднання однорідних елементів та обчислення нового центроїду з урахуванням обробленої інформації. Вектори $X_1, X_2, \ldots, X_n$ представляють числові значення часових рядів, що відображають поведінку процесу. На основі їх значень визначається множина станів Z. C – множина класів, де підмножини поділяються на небезпечні C_1 та на безпечні C_2 стану. Існує цільова залежність – відображення $Z \to C$, значення якої відомі тільки на об'єктах кінцевої навчальної вибірки $X = \{(x_{11}, \ldots, x_{n1}), (x_{12}, \ldots, x_{n2}), (x_{1m}, \ldots, x_{nm})\}$. Необхідно побудувати алгоритм обробки $X_i - a$, здатний класифікувати вектор, що подається на вхід. Над станами Z, які характеризуються векторами синхронізованих тимчасових рядів X отриманих від пристроїв, проводиться спостереження. Визначається, до якого класу C та його підмножин C_1 та C_2 відноситься досліджуваний стан z_j. Значення векторів X, які містять шаблонні послідовності від датчиків у різних умовах функціонування, віднесених до класів множин C_1 та C_2 є навчальною вибіркою.

За значеннями синхронізованого часового ряду декількох датчиків $x = (x_{1i}, \ldots, x_{ni})$ вектору ознак $X = X_1, X_2, \ldots, X_n$ проводиться ідентифікація класу C, відповідного стану z_j. Розподіл значень випадкового вектора X мають різні параметри. Вирішальне правило $r'(x)$ для алгоритму a ставить у відповідність спостереженню x одне з множин C_1 або C_2. Воно визначається функцією *f(x)*, що породжує розбиття простору на дві області, що не перетинаються:

$$r'(x) = \begin{Bmatrix} C_1, \text{при } f(x) \geq \varepsilon \\ C_2 \text{при } f(x) < \varepsilon \end{Bmatrix}, \quad (3)$$

де ε – порогове значення.

Запропонований підхід ідентифікації стану відрізняється використанням шаблонних послідовностей. Ці послідовності вигідно відрізняються тим, що використовують шаблонні послідовності, що описують сформовані в минулому умови функціонування пристрою, що містять синхронізовані часові ряди, що показують отримані від різних датчиків і сенсорів числові значення під час виконання процесів. Це дозволяє визначати стан ІБ, не збільшуючи обсяг бази даних інформації, що зберігається. Це якісно виділяє цей підхід, через технічні обмеження пристроїв IoT, що були описані вище.

Використання запропонованого підходу на початковому етапі передбачає "налаштування" пристрою в заздалегідь заданих режимах роботи, де відбувається перед обробка, пов'язана з обчисленням кластерних областей та порогових значень, що може відбуватися на основі вибірки. На етапі функціонування на основі функції розбиття простору здійснюється співвідношення поточного стану до визначених на початковому етапі.

Подальший аналіз відхилень відбувається з урахуванням порівнянь з еталонними значеннями центрів кластерів і класів, обчисленими за умов, заданих для формування навчальної вибірки.

Результати експерименту

В рамках експерименту щодо реалізації запропонованого підходу було виконано з'єднання типу "мережевий міст". Схема проведення експерименту наведено на рис. 1.

Для моніторингу стану інформаційної безпеки виявлення змін обчислювального середовища, процесів, що реалізують функціонал на користь зловмисника, є актуальним проблемним питанням. У зв'язку з цим виникає необхідність ідентифікувати стан пристрою як безпечного або небезпечного.

Метою проведення експерименту було виявлення стану, який визначається алгоритмом обробки даних, обчислювального вузла на основі оцифрованих показників завантаження обчислювальних ресурсів [11-13]. Як послідовність, що формує поведінковий шаблон, були виділені синхронізовані за часом відсоткові показники монітора системного завантаження.

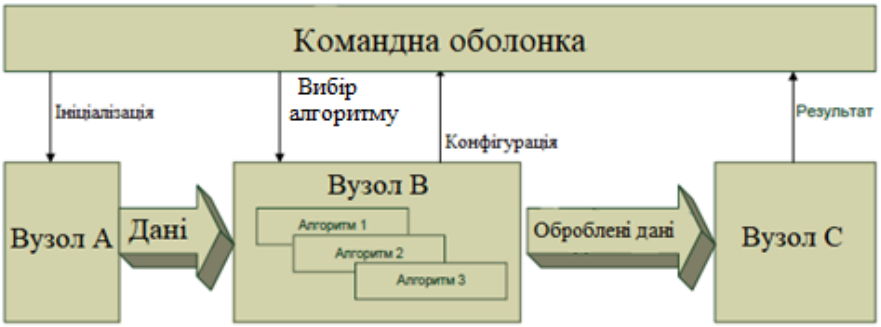


Рис. 1 – Схема проведення експерименту

Від вузла А на вузол В через вузол С передавалися файли, що містять поля таблиці баз даних. У вузлу С здійснювалося перемикання алгоритмів обробки. В першому випадку передача інформації здійснювалася через вузол С без обробки (стан Z_1), у другому (стан Z_2) – проводилася фільтрація по заздалегідь заданому полю таблиці, в третьому (стан Z_3) – виконувалися обчислення і до переданої таблиці додавалися додаткові поля. Умовно вважаючи, що Z_1 – безпечний стан, а Z_2 і Z_3 – небезпечні, ідентифікація стану ІБ визначалась через сигнальні послідовності трас системного монітора вузла С.

Синхронізовані у часі послідовності значень відсоткового використання ресурсів центрального процесора, мережевих пакетів, споживаних ресурсів пам'яті представлені у вигляді вихідних векторів.

При проведенні експерименту була отримана вибірка шаблонів сигнальних трас для станів, що розглядаються, яка була розділена на "навчальну" і "тестову" (рис. 2-4).

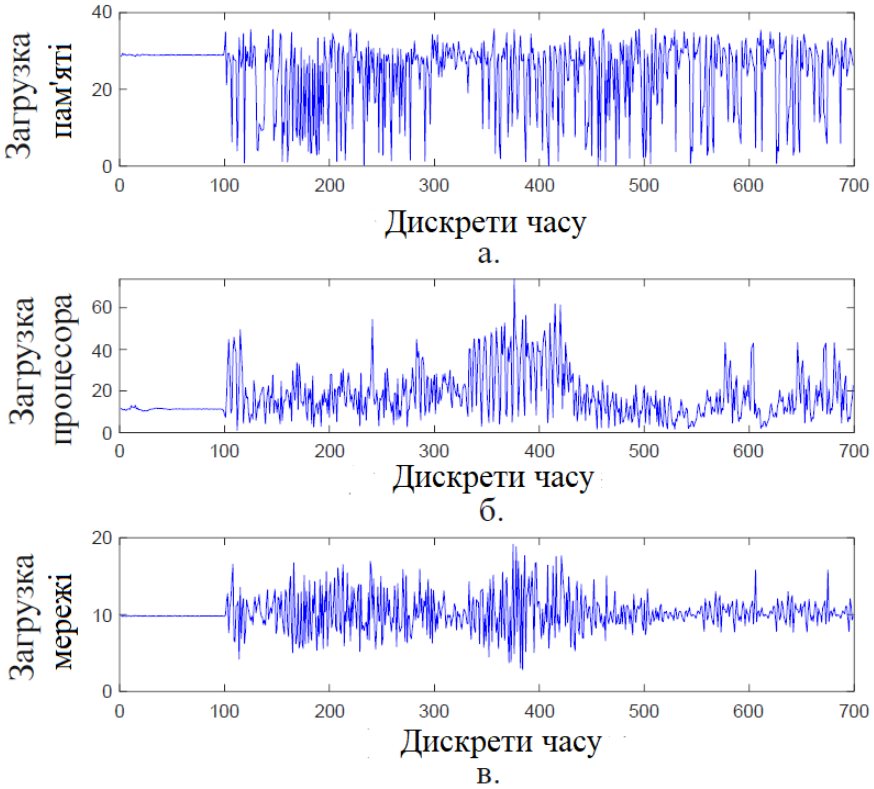


Рис. 2 – Приклад вибірки відсоткового завантаження ресурсів (згори донизу відповідно – пам'ять, процесор, мережа) від дискретів часу (часові звіти від 0 до 600) для стану Z_1

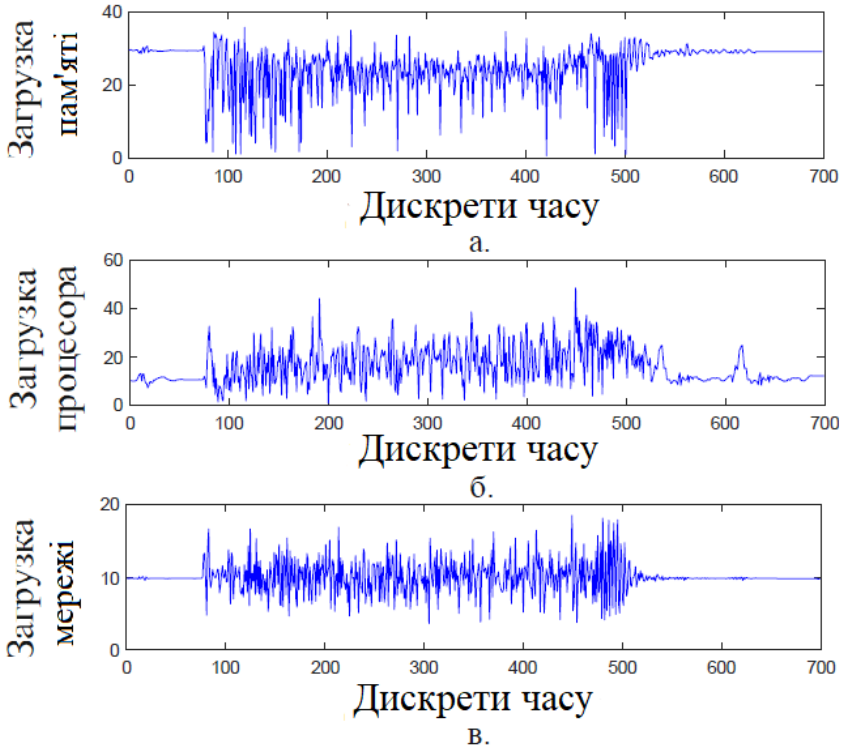


Рис. 3 – Приклад вибірки відсоткового завантаження ресурсів (згори донизу відповідно – пам'ять, процесор, мережа) від дискретів часу (часові і звіти від 0 до 600) для стану Z_2

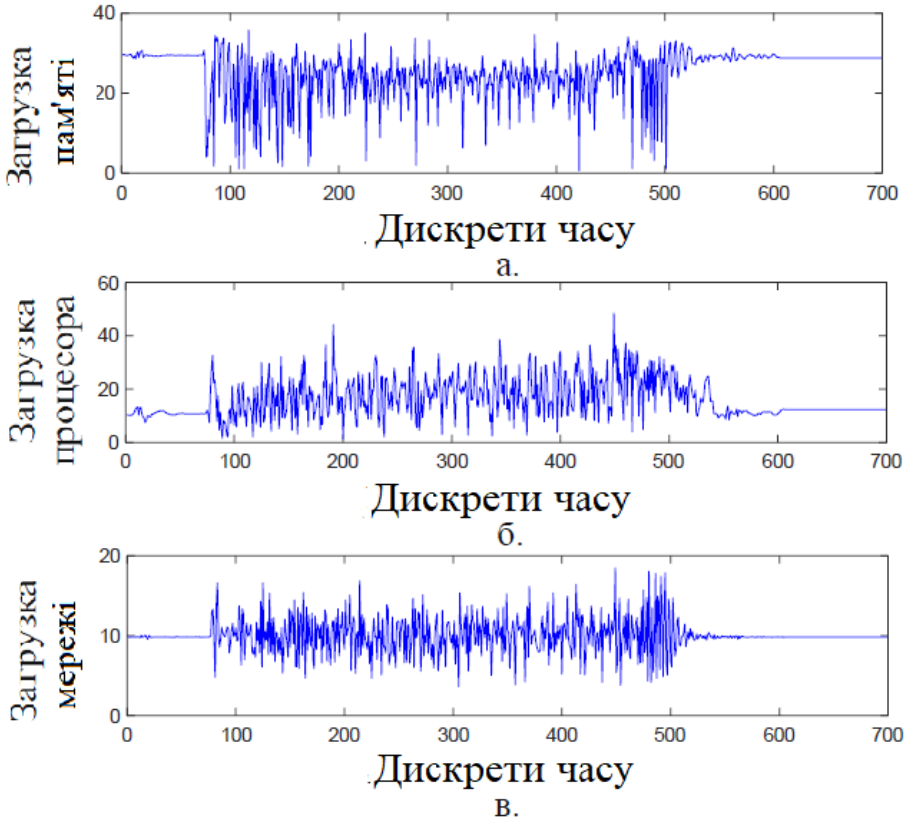


Рис. 4 – Приклад вибірки відсоткового завантаження ресурсів (згори донизу відповідно – пам'ять, процесор, мережа) від дискретів часу (часові звіти від 0 до 600) для стану Z_3

Ідентифікація стану виконувалася на основі методу кластеризації *k*-середніх. Як міра близькості використана Евклідова відстань:

$$r(x, \dot{x}) = \sqrt{\sum_{\rho=1}^{n} (x_\rho - \dot{x}_\rho)^2}\,, \tag{4}$$

де R – простір спостережень; $x, \dot{x} \in R^n$.

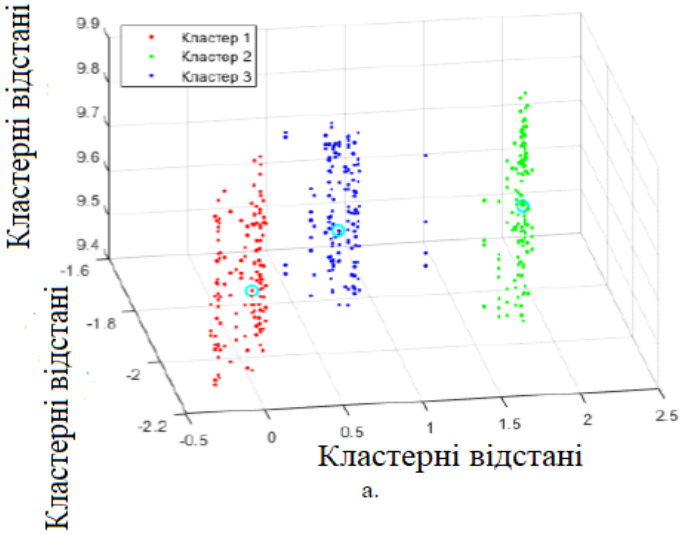
За допомогою значень навчальної вибірки на основі методу *k*-середніх здійснено поділ q спостережень на l груп (або кластерів) $(l \le q), C = \{C_1, C_2, \dots, C_0\}$:

$$min\left[\sum_{i=1}^{l}\sum_{x}(C)_{\in z}\left|x^{(j)} - \mu_i\right|^2\right], \tag{5}$$

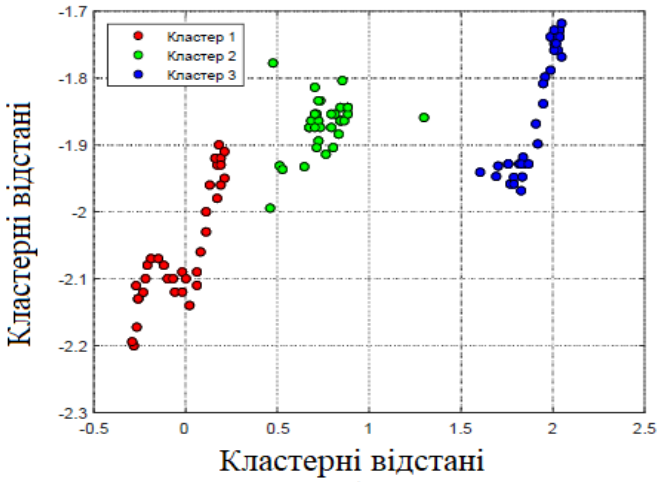
де $x^{(j)} \in R^n$; $\mu_i \in R^n$; μ_i – центроїд для кластера C_i.

Процес ідентифікації стану полягав у тому, що за даними послідовності, що надходить, обчислюються значення, які порівнюються з центроїдами кластерів.

Фіксовані часові ряди послідовностей значень від центрального процесора, мережевих пакетів, споживаних ресурсів пам'яті для різних станів, а також кластери станів досить добре відрізняються один від одного (вирази (4) та (5)), що візуально простежується на областях (рис. 5) і підтверджується аналізом розмірів кластерів та відстаней усередині (рис. 6).



а.



б.

Рис. 5 – Результати кластеризації з урахуванням середнього значення зміни координат

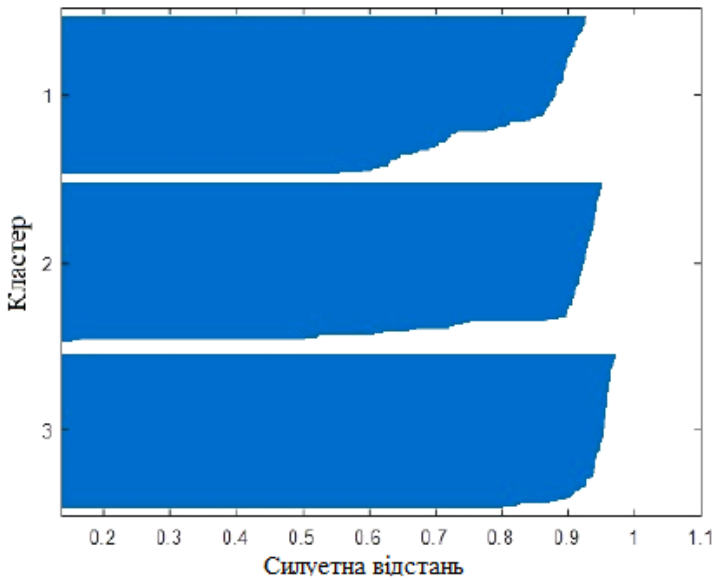


Рис. 6 – Візуалізація оцінки отриманих кластерів

Вимірюючи відстань до центроїдів, вибирається мінімальне значення, на основі якого приймається рішення про належність до кластера, що ідентифікує стан. Загальна точність обраного запропонованого рішення для повної класифікації склала 0,96 (рис. 6-7).

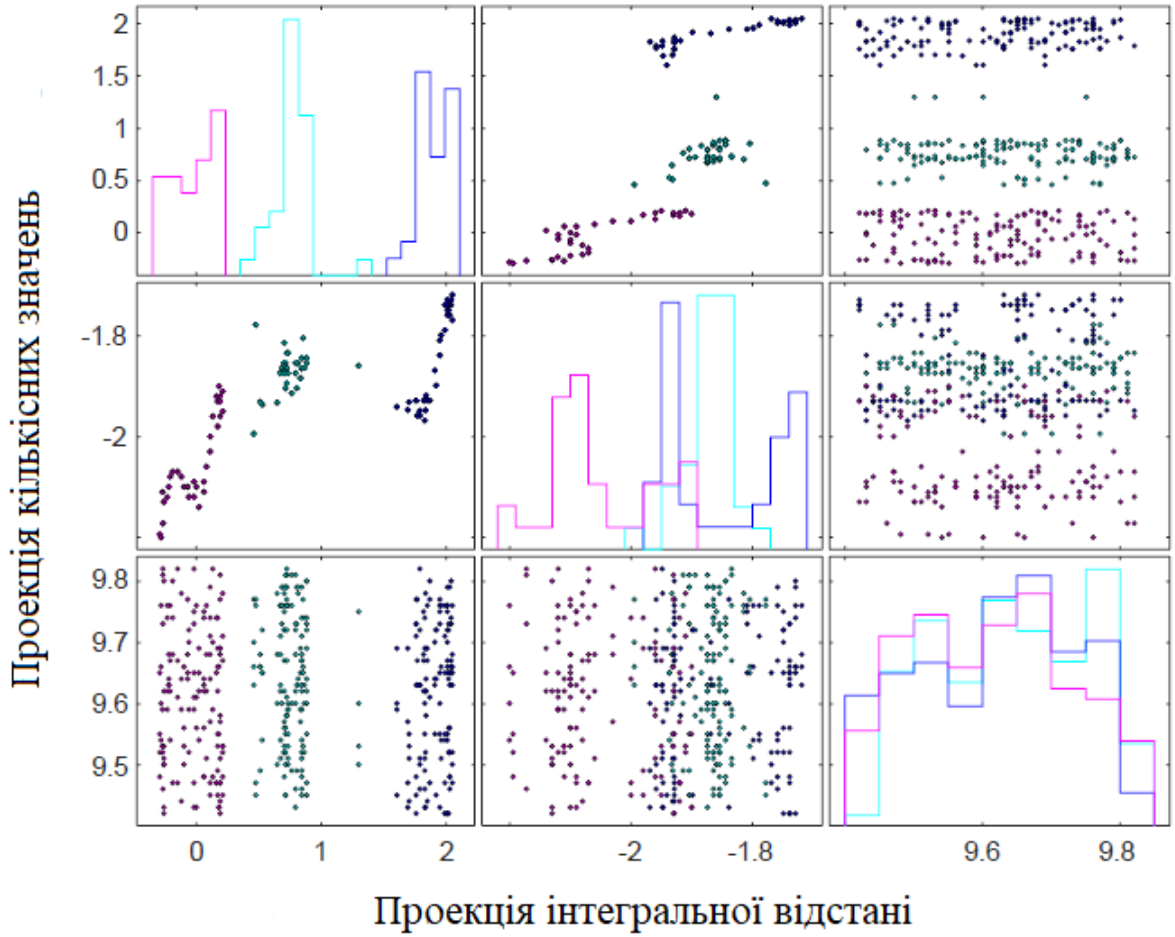


Рис. 7 – Ознаки (рядки зверху вниз – завантаження пам'яті, процесора та мережі), за якими здійснюється ідентифікація станів ІБ пристроїв

Гістограма на рис. 6 показує, що вибрані значення векторів, що ідентифікують стан, розбиті на три кластери однакового розміру, що відповідає характеристикам експериментальних вибірок даних. Більшість точок у кластерах мають великі значення відносної відстані (0,8 або більше), що вказує на те, що кластери добре розділені.

Для візуального аналізу даних на рис. 7 одержані значення векторів відображені у вигляді матриці двовимірних декартових графіків розсіювання, згрупованих за різними вхідними аргументами. На основі даних, що відображаються, та їх гістограм можна визначити якість ознак, що впливають на ідентифікацію станів.

У проведеному обчислювальному експерименті видно, що найбільший внесок у якість ідентифікації стану ІБ вносять сигнали про завантаження пам'яті. Таким чином, запропонований підхід дозволяє визначити клас поточного стану пристрою ІБ.

Надалі, як продовження дослідження, аналізуючи значення для різних завдань оптимізації обсягу обчислень або бази даних, де зберігаються синхронізовані часові ряди, отримані від датчиків і вимірювальних перетворювачів, можна використовувати адитивний критерій, наприклад узагальнений критерій оптимальності

$$\min_{\bar{x}\in X} F\left(\overline{w}, \overline{X}(t)\right) = \min_{\bar{x}\in X} \sum_{i=1}^{S} w_i X_i(t) \quad (6)$$

за умови

$$\sum_{i=1}^{S} w_i = 1\,, \quad (7)$$

де w_i – вагові коефіцієнти, що дозволяють створювати пріоритет найважливішим критеріям з допомогою збільшення їх значень w_i.

Висновки

1. Широке практичне використання концепції IoT обмежується необхідністю вирішення проблеми забезпечення інформаційної безпеки в спектрі захисту від загроз зловмисників.
2. Моніторинг стану інформаційної безпеки пристроїв інформаційно-телекомунікаційних систем та мереж у концепції IoT обумовлює необхідність пошуку та вдосконалення підходів до виявлення різного виду загроз.
3. Процес моніторингу стану пристроїв IoT вимагає аналізу великої кількості показників. Вибір даних для аналізу істотно впливає на якісні показники визначення станів. При цьому небажані як недостатня, так і надмірна кількість інформативних показників.
4. Використання часових рядів для математичного опису стану пристроїв в мережі IoT надає можливість ідентифікувати стан інформаційної безпеки пристрою IoT без збільшення обсягу інформації, що зберігається і обробляється у внутрішніх обчислювальних ресурсах. Ідентифікація стану ІБ пристроїв здійснюється за трьома ознаками: загрузка мережі, загрузка процесору, загрузка пам'яті.
5. Іншим напрямом розвитку подібних рішень, враховуючи відносну обмеженість ресурсів пристроїв є організація зовнішніх центрів обробки даних, на які пристрої IoT можуть надсилати інформацію про своє функціонування для подальшого аналізу стану.

Перелік посилань

1. Farwell J. P., Rohozinski R. Stuxnet and the Future of Cyber War // Survival. 2011. Vol. 53. № 9. P. 23-40.
2. Yeung D. Y., Ding Y. Host-based intrusion detection using dynamic and static behavioral models // Pattern recognition. 2003. Vol. 36. P. 229-243.
3. Zhang Baoquan, Zou Zongfeng, Liu Mingzheng, Evaluation on security system of internet of things based on Fuzzy-AHP method, E-Business and E-Government (ICEE), 2011 International Conference on 2011, pp. 1 – 5.
4. Schurgot, M.R.; Shinberg, D.A.; Greenwald, L.G., Experiments with security and privacy in networks, World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2015 IEEE 16th International Symposium on, 2015, pp. 1-6.
5. Зикратов И. А., Зикратова Т. В., Лебедев И. С. Доверительная модель информационной безопасности мультиагентных робототехнических систем с децентрализованным управлением // Научно-технический вестник информационных технологий, механики и оптики. 2014. № 2 (90). С. 47-52.
6. Gao D., Reiter M., Song D. Beyond output voting: Detecting compromised replicas using HMM-based behavioral distance // IEEE Transactions on Dependable and Secure Computing. 2009. Vol. 6. № 2. P. 96-110.
7. Макаренко С. И., Олейников А. Я, Черницкая Т. Е. Модели интероперабельности информационных систем // Системы управления, связи и безопасности. 2019. № 4. С. 215-245.
8. Bevir M. K., O'Sullivan V. T., Wyatt D. G. Computation of electromagnetic flowmeter characteristics from magnetic field data // Journal of Physics D Applied Physics. 1981. Vol. 14. № 3. P. 373-388.
9. Semenov V. V., Lebedev I. S., Sukhoparov M. E., Salakhutdinova K. I. Application of an Autonomous Object Behavior Model to Classify the Cybersecurity State. Internet of Things, Smart Spaces, and Next Generation Networks and Systems, 2019, pp. 104-112.
10. Сошникова Л. А., Тамашевич В. Н., Усбе Г., Шефер М. Многомерный статистический анализ в экономике: учебное пособие для вузов. – М.: ЮНИТИ – Дана, 1999. 598 с.

11. Golub T. R. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring // Science. 1999. Vol. 286. P. 531-537.

12. Anderberg M. R. Cluster Analysis for Applications. – Academic Press, New York, 1976. – 376 p.

13. Dembele D., Kastner P. Fuzzy C-means method for clustering microarray data // Bioinformatics. 2003. Vol. 19. № 8. P. 973-980.

14. Fritz H, Garcıa-Escudero LA, Mayo-Iscar A. Tclust: An R package for atrimming approach to cluster analysis. J Stat Softw 2012;47(12):1–26.http://dx.doi.org/10.18637/jss.v047.i12.

15. Ertöz L, Steinbach M, Kumar V. Finding clusters of different sizes, shapes,and densities in noisy, high dimensional data. In: Proceedings of the 2003SIAM international conference on data mining. SIAM; 2003, p. 47–58.http://dx.doi.org/10.1137/1.9781611972733.5.

16. Wehrens R, Buydens L. Self-and super-organizing maps in R: The kohonenpackage. J Stat Softw 2007;21(5):1–19. http://dx.doi.org/10.18637/jss.v021.i05.

17. Ng AY, Jordan MI, Weiss Y. On spectral clustering: Analysis and analgorithm. In: Proceedings of the 14th International Conference on NeuralInformation Processing Systems: Natural and Synthetic, NIPS'01. Cam-bridge, MA, USA: MIT Press; 2001, p. 849–56. http://dx.doi.org/10.5555/2980539.2980649.

18. John CR, Watson D, Barnes MR, Pitzalis C, Lewis MJ. Spectrum: Fastdensity-aware spectral clustering for single and multi-omic data. Bioin-formatics 2020;36(4):1159–66. http://dx.doi.org/10.1093/bioinformatics/btz704.

19. Aggarwal CC, Yu PS. Finding generalized projected clusters in high di-mensional spaces. In: Proceedings of the 2000 ACM SIGMOD internationalconference on management of data. 2000, p. 70–81.

20. Liu, Tianmou & Yu, Han & Blair, Rachael. (2022). Out-of-bag stability estimation for k -means clustering. Statistical Analysis and Data Mining: The ASA Data Science Journal. 10.1002/sam.11593.

21. Kalia, Khushboo & Dixit, Saurav & Kumar, Kaushal & Gera, Rajat & Epifantsev, Kirill & John, Vinod & Taskaeva, Natalia. (2022). Improving MapReduce heterogeneous performance using KNN fair share scheduling. Robotics and Autonomous Systems. 157. 104228. 10.1016/j.robot.2022.104228.

22. Sriphum, Wiwat & Wills, Gary & Green, Nicolas. (2021). Floptics: A Novel Automated Gating Technique for Flow Cytometry Data. International Journal of Organizational and Collective Intelligence. 12. 10.4018/IJOCI.301561.

23. Sukhdev Singh Ghuman, "Clustering Techniques - A Review," International Journal of Computer Scienceand Mobile Computing, Vol. 5, Pp. 524-530, 2016

24. Pradeep Rai and Shubha Singh, "A Survey of Clustering Techniques," International Journal of Computer Applications, Vol. 7, Pp. 1-5, 2010

25. Kavitha V.and Punithavalli M., "Clustering Time Series Data Stream - A Literature Survey," International Journal of Computer Science and Information Security, Vol. 8, pp. 289-294, 2010.

26. Saroj and Tripti Chaudhary, "Study on Various Clustering Techniques," International Journal of ComputerScience and Information Technologies, Vol. 6, pp. 3031-3033, 2015.

27. Vijayalaksmi S.and Punithavalli M., "A Fast Approach to Clustering Datasets using DBSCAN and Applications," Vol. 60, pp. 1-7, 2012.

Надійшла: 16.08.2022 Рецензент: д.т.н., професор Савченко В.А.