



DOI 10.28925/2663-4023.2026.35.1375

УДК 004.056.53

Ніщенко Дмитро Олександрович

доктор філософії, старший викладач кафедри Технологій цифрового розвитку
Державний університет інформаційно-комунікаційних технологій, Київ, Україна
ORCID: 0009-0006-1781-4109
d.nishchenko@duikt.edu.ua

Сало Наталія Анатоліївна

кандидат технічних наук, старший викладач кафедри Аеронавігаційного обслуговування
Українська державна льотна академія, Кропивницький, Україна
ORCID: 0000-0003-1422-9853
nataliia.salo@gmail.com

Нестеренко Катерина Сергіївна

доктор технічних наук, професор, директор Навчально-наукового інституту інформаційних технологій
Державний університет інформаційно-комунікаційних технологій, Київ
ORCID: 0000-0001-7672-7386
k.nesterenko@duikt.edu.ua

Мамченко Сергій Миколайович

доктор педагогічних наук, професор кафедри комп'ютерних систем, мереж та кібербезпеки
Національний університет біоресурсів та природокористування України, Київ, Україна
Київський столичний університет імені Бориса Грінченка, Київ, Україна
ORCID: 0009-0006-8743-5606
s.mamchenko@nubip.edu.ua

**УДОСКОНАЛЕННЯ МЕТОДУ ВИЯВЛЕННЯ АТАК НА MQTT-СЕГМЕНТ ІoT-МЕРЕЖІ
РОЗУМНОГО БУДИНКУ З АДАПТАЦІЄЮ ДО НОВОЇ МЕРЕЖІ ЗА ЛЕГІТИМНИМ ТРАФІКОМ**

Анотація. Протокол MQTT є основним засобом обміну повідомленнями між сенсорами, виконавчими пристроями та шлюзом розумного будинку, тому атаки на брокер MQTT безпосередньо впливають на доступність і цілісність IoT-системи. Відомі методи виявлення таких атак на основі машинного навчання демонструють точність 99–100 % за умови, що навчання й перевірку виконано на трафіку одного стенда, тоді як на практиці детектор застосовується в мережі, трафік якої не використовувався для навчання, і має працювати за заданого рівня хибних тривог. У статті удосконалено метод виявлення атак на MQTT-сегмент IoT-мережі, який ґрунтується на віконних протокольних ознаках трафіку брокера, незалежних від мережевих ідентифікаторів і абсолютного часу, та на нормуванні ознак і порога рішення за легітимним трафіком цільової мережі без використання її атакуючих міток: ознаки нормуються медіаною й міжквартильним розмахом легітимних калібрувальних вікон, а поріг визначається як порядкова статистика їх оцінок для заданого рівня хибних тривог. Наведено методіку дослідження на чотирьох незалежних наборах даних (MQTTset, MQTT-IoT-IDS2020, MQTTEEB-D, MQTT_UAD): відтворення відомих підходів за умовами їх авторів, перевірку пакетних ознак за хронологічного поділу та в іншій мережі, оцінювання удосконаленого методу для 12 пар «мережа навчання – цільова мережа» за рівня хибних тривог 1 %. Середня повнота виявлення атак удосконаленим методом становить 0,710 (медіана 0,905) проти 0,532 для калібрування лише порога, 0,549 для детектора, навченого лише на легітимному трафіку цільової мережі, і не більше 0,124 для пакетних ознак; результат зберігається для розміру вікна 5–30 с і обсягу калібрування 10–50 % часу. Метод реалізовано мовою C як модуль IoT-шлюзу: обробка трафіку триває 180–400 нс на пакет, модель займає 21 КБ.

Ключові слова: кібербезпека; Інтернет речей; MQTT; виявлення вторгнень; розумний будинок; IoT-шлюз; перенесення між мережами; калібрування порога.



ВСТУП

Постановка проблеми. Системи розумного будинку об'єднують сенсори, виконавчі пристрої та шлюз, які обмінюються повідомленнями переважно за протоколом MQTT за схемою «видавець – брокер – підписник». Брокер MQTT на шлюзі є єдиною точкою, через яку проходять показання сенсорів і команди керування, тому підбір облікових даних, перевантаження брокера потоком повідомлень, повільні атаки на сеанси та надсилання пошкоджених пакетів безпосередньо порушують доступність і цілісність усієї IoT-системи [20]. Від захищеності каналу обміну повідомленнями залежить і достовірність сенсорних даних, на основі яких здійснюється керування розумним будинком [21, 22]. Для виявлення таких атак застосовують мережеві системи виявлення вторгнень на основі машинного навчання, які аналізують трафік брокера.

Практична особливість застосування таких систем полягає в тому, що модель навчають на трафіку одного стенда, де є розмічені атаки, а встановлюють у іншу мережу з іншими пристроями, брокером і поведінкою легітимних клієнтів. Для нової мережі зазвичай доступний лише записаний легітимний трафік, а розмічених атак немає. Крім того, детектор має працювати за заданого рівня хибних тривог, оскільки часті хибні спрацювання роблять систему непридатною для експлуатації. Тому важливо, щоб показники якості детектора визначалися саме в таких умовах, а метод виявлення забезпечував перенесення в нову мережу без повторного навчання на її атаках.

Аналіз останніх досліджень і публікацій. Для дослідження атак на MQTT створено низку відкритих наборів даних. Набір MQTTset [20] містить трафік емульованої мережі розумного будинку з атаками перевантаження брокера, підбору облікових даних, SlowITe та надсилання пошкоджених пакетів. Набір MQTT-IoT-IDS2020 [11] містить трафік 12 емульованих сенсорів зі скануванням мережі, підбором паролів SSH і підбором облікових даних MQTT. Пізніше опубліковано набори DoS/DDoS-MQTT-IoT [2] і CICIoMT2024 з MQTT-атаками на медичні IoT-пристрої [9], а також набори з реальних і реалістичних стендів MQTTEEB-D [4] і MQTT_UAD [6].

На цих наборах запропоновано детектори на основі дерев рішень, випадкового лісу, градієнтного бустингу та глибоких нейронних мереж. Для MQTTset за випадкового поділу пакетів у співвідношенні 70/30 отримано точність 99,4 % [20]; для MQTT-IoT-IDS2020 на ознаках односпрямованих і двоспрямованих потоків – до 99,98 % [11], з глибокою нейронною мережею та п'ятикратною перехресною перевіркою – понад 99 % [13]; для мереж розумного будинку на пакетних ознаках MQTT-IoT-IDS2020 – 100 % [1]. Високі показники на цих наборах наведено також для трансформерних моделей [19]. Перетворення пакетних ознак MQTTset з додаванням адреси джерела підвищує точність дерева рішень з 92,57 % до 98,56 % [12]. Сеансові протокольні ознаки MQTT, сформовані з трафіку кількох об'єднаних наборів, використано для інтерпретованої моделі TabNet за стратифікованого випадкового поділу 80/20 [14].

Окремий напрям досліджень стосується умов оцінювання мережевих детекторів. Систематизовано типові пастки машинного навчання в комп'ютерній безпеці, зокрема перетин навчальних і тестових даних та ознаки, що ідентифікують середовище запису [5], сформульовано вимоги до прагматичного оцінювання мережевих детекторів [3], а для еталонних наборів показано вплив рішень щодо їх дизайну на отримуваних показники [10]. Для IoT-трафіку встановлено, що випадковий поділ потоків одних і тих самих сеансів між навчальною й тестовою вибірками підвищує повноту виявлення до 14 в. п., а ідентифікатори вузлів і мітки часу можуть заміщувати поведінкові ознаки атак [18].

Перенесення детекторів між мережами досліджено для класичних наборів CIC-IDS2017/2018, де точність у перехресній перевірці між наборами наближається до випадкового вгадування [7], а для MQTT-атак – у парі «власний стенд – MQTT-IoT-IDS2020» з точністю понад 80 % [17]. Найбільш повне дослідження для IoT охоплює вісім мереж, зокрема MQTTset і MQTT-IoT-IDS2020 [8]: на транспортних ознаках перших пакетів двоспрямованих потоків об'ємні атаки переносяться між мережами, тоді як для MQTTset як цільової мережі падіння F1-міри становить 80–90 %, що автори пов'язують зі специфічними для протоколу заголовками MQTT; там же оцінено пам'ять і затримку моделей на периферійних пристроях. Для розгортання на шлюзах запропоновано легковагові детектори з оцінюванням на одноплатному комп'ютері Raspberry Pi [16]. Узагальнення умов оцінювання в цих роботах наведено в табл. 1.

Таким чином, високі показники відомих методів отримано переважно за випадкового поділу записів одного стенда, а дослідження перенесення між мережами спираються на транспортні ознаки без урахування прикладного рівня MQTT. Невирішеними залишаються такі частини загальної проблеми: метод, ознаки якого відображають поведінку протоколу MQTT і не залежать від адрес, портів і часу запису конкретного стенда; правило налаштування детектора під нову мережу лише за її легітимним трафіком із забезпеченням заданого рівня хибних тривог; оцінювання на кількох незалежних стендах в умовах



перенесення з вимірюванням обчислювальних витрат на шлюзі.

Наукова новизна роботи полягає в тому, що удосконалено метод виявлення атак на MQTT-сегмент IoT-мережі, який, на відміну від відомих, ґрунтується на віконних протокольних ознаках трафіку брокера, незалежних від мережевих ідентифікаторів і абсолютного часу, та на нормуванні цих ознак і порога рішення за легітимним трафіком цільової мережі без використання її атакувальних міток, що забезпечує підвищення повноти виявлення атак у новій мережі за заданого рівня хибних тривог.

Таблиця 1

Умови оцінювання відомих методів виявлення атак на MQTT і удосконаленого методу

Робота	Набір даних	Ознаки	Поділ даних	Перевірка в іншій мережі	Фіксований рівень хибних тривог	Реалізація на шлюзі
[20]	MQTTset	пакетні	випадковий 70/30	ні	ні	ні
[11]	MQTT-IoT-IDS2020	пакетні, потоківі	випадковий 75/25, 5 частин	ні	ні	ні
[13]	MQTT-IoT-IDS2020	пакетні, потоківі	5 частин	ні	ні	ні
[1]	MQTT-IoT-IDS2020	пакетні	випадковий 60/40	ні	ні	ні
[12]	MQTTset	пакетні, адреса джерела	70/30 авторів набору	ні	ні	ні
[14]	3 набори разом	сеансові MQTT	стратифікований 80/20	ні	ні	ні
[17]	власний стенд	пакетні	стратифікований	1 пара	ні	ні
[8]	8 IoT-наборів	транспортні, перші пакети	70/30 і між мережами	так	pAUC при 1 %, без калібрування	оцінено пам'ять і затримку
Ця робота	4 MQTT-набори	віконні протокольні	хронологічний і між мережами	12 пар	1 %, калібрування за легітимним трафіком	модуль мовою C, виміряно

Мета статті. Метою статті є підвищення повноти виявлення атак на MQTT-сегмент IoT-мережі розумного будинку в новій мережі за заданого рівня хибних тривог шляхом удосконалення методу виявлення на основі віконних протокольних ознак і калібрування за легітимним трафіком цільової мережі. Для досягнення мети поставлено такі задачі: проаналізувати умови оцінювання відомих методів і відтворити їх результати; визначити показники пакетних ознак за хронологічного поділу й у новій мережі; розробити віконні протокольні ознаки, правила нормування й калібрування порога; експериментально оцінити удосконалений метод на чотирьох незалежних наборах даних і його стійкість до параметрів; реалізувати метод мовою C для IoT-шлюзу й оцінити обчислювальні витрати.

МЕТОДИКА ДОСЛІДЖЕННЯ

Набори даних і узгодження міток

Дослідження виконано на чотирьох відкритих наборах даних, записаних на незалежних стендах (табл. 2). Набори відрізняються способом отримання трафіку (емуляція чи реальні пристрої), складом легітимних клієнтів і реалізацією атак, що відповідає умовам застосування детектора в новій мережі.

Таблиця 2

Набори даних, використані в дослідженні

Набір	Стенд	Пакетів MQTT, тис.	Сім'ї атак	Вікон $W = 10$ с
MQTTset [20]	емуляція, сенсори розумного будинку	12 081	підбір облікових даних, DoS/flood, SlowITe, malformed	64 966
MQTT-IoT-IDS2020 [11]	емуляція, 12 сенсорів	10 968	підбір облікових даних	3 789
MQTTEEB-D [4]	реальний стенд на Raspberry Pi 4	223	підбір облікових даних, DoS/flood, SlowITe, malformed	5 078
MQTT_UAD [6]	реалістичний стенд (домашня мережа)	64	DoS, інші (MitM, вторгнення)	1 034



/Для всіх наборів сформовано єдину пакетну таблицю трафіку брокера (порт 1883). Для MQTTset, MQTTEEB-D і MQTT_UAD використано файли CSV наборів, для MQTT-IoT-IDS2020 – файли PCAP, оброблені власним програмним розбором заголовків Ethernet, IPv4, TCP і MQTT. Спільними для всіх наборів є шість полів пакета: довжина корисного навантаження TCP, прапорці TCP, тип і молодші біти фіксованого заголовка MQTT, прапорці та інтервал keepAlive пакета CONNECT. Мітки узгоджено до сімей: норма, підбір облікових даних, DoS/flood, SlowITe, malformed та інші атаки. У наборах MQTTset і MQTT-IoT-IDS2020 записи атак отримано окремо від легітимного трафіку, тому для віконного аналізу кожен запис атаки накладався зі зсувом у часі на легітимний потік того самого набору (15 повторів у випадкових місцях відповідної частини).

Протокол оцінювання

Спочатку відтворювалися відомі підходи за умовами їх авторів (поділ даних, ознаки, моделі), що дає змогу порівняти отримані значення з опублікованими. Далі ті самі пакетні ознаки оцінювалися за хронологічного поділу (перші 70 % часу кожного сценарію – навчання, решта – перевірка) і в умовах перенесення «мережа навчання A – цільова мережа B » для 12 упорядкованих пар наборів. Удосконалений метод оцінювався для тих самих пар: модель навчалася на навчальній частині мережі A , параметри нормування й поріг визначалися за легітимними вікнами перших 30 % часу мережі B , а показники обчислювалися на решті даних мережі B , атакуювальні мітки якої під час налаштування не використовувалися.

Основними показниками є повнота виявлення атак (TPR) і частка хибних тривог (FPR) на вікнах цільової мережі:

$$TPR = \frac{|\{w \in \mathcal{A}_B: s(w) > \tau\}|}{|\mathcal{A}_B|}, \quad FPR = \frac{|\{w \in \mathcal{N}_B: s(w) > \tau\}|}{|\mathcal{N}_B|}, \quad (1)$$

де $\mathcal{A}_B$ і $\mathcal{N}_B$ – множини атакуювальних і легітимних вікон цільової мережі; $s(w)$ – оцінка вікна w ; τ – поріг рішення. Цільовий рівень хибних тривог становить $\alpha = 0,01$. Для кожної пари виконано 5 запусків з різними початковими значеннями генератора випадкових чисел; довірчі інтервали 95 % отримано бутстрепом (1000 повторів). Для відтворення відомих підходів додатково наведено точність і макросереднену F1-міру.

РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Удосконалений метод виявлення атак

Метод складається з двох етапів (рис. 1): підготовки моделі на мережі-джерелі A з розміченими атаками та адаптації до цільової мережі B лише за її легітимним трафіком. Кожен пакет трафіку брокера описується кортежем

$$p_i = (t_i, \ell_i, f_i, m_i, h_i, k_i, c_i), \quad (2)$$

де t_i – час; ℓ_i – довжина корисного навантаження TCP; f_i – прапорці TCP; m_i і h_i – тип і молодші біти фіксованого заголовка MQTT; k_i і c_i – інтервал keepAlive і прапорці пакета CONNECT. Адреси й порти до кортежу не входять. Пакети групуються в часові вікна тривалістю W :

$$w_n = \{p_i: \lfloor (t_i - t_0)/W \rfloor = n\}, \quad (3)$$

де t_0 – час першого пакета; n – номер вікна. Для кожного вікна обчислюється вектор $\varphi(w) \in \mathbb{R}^{32}$ (табл. 3), що описує поведінку протоколу на брокері: інтенсивність, частки типів керуючих пакетів, прапорці TCP і PUBLISH, параметри CONNECT і статистики довжин. Частки ознак обчислюються відносно кількості пакетів відповідного типу, наприклад частка пакетів CONNECT і відношення кількості CONNECT до PUBLISH:

$$\varphi_{\text{con}}(w) = \frac{1}{|w|} \sum_{p_i \in w} \mathbf{1}[m_i = 1], \quad \varphi_{\text{cpp}}(w) = \frac{N_{\text{CONNECT}}(w)}{N_{\text{PUBLISH}}(w) + 1}. \quad (4)$$

Віконні протокольні ознаки

Група ознак	Кількість	Ознаки
Інтенсивність	1	кількість пакетів за секунду
Типи пакетів MQTT	10	частки CONNECT, CONNACK, PUBLISH, PUBACK, SUBSCRIBE, PINGREQ, DISCONNECT, інших типів, сегментів без заголовка MQTT, некоректних типів (0 і 15)
Прапорці TCP	3	частки SYN, FIN, RST
Співвідношення	1	$N_{\text{CONNECT}} / (N_{\text{PUBLISH}} + 1)$
Прапорці PUBLISH	4	частки DUP, QoS 1, QoS 2, RETAIN
Пакети CONNECT	7	частки keepAlive > 600 с і keepAlive = 0, прапорців імені, пароля, will, clean session; $\ln(1 + \max k)$
Довжини	6	середнє, стандартне відхилення, 90-й перцентиль і максимум довжини; середнє й максимум довжини PUBLISH

Вікно маркується за кількістю атакуючих пакетів $a(w)$:

$$y(w) = \begin{cases} \text{норма,} & a(w) = 0, \\ \text{сім'я атаки,} & a(w) \geq 5 \text{ і } a(w)/|w| \geq 0,05, \\ \text{вилучається,} & \text{в інших випадках.} \end{cases} \quad (5)$$

де сім'я атаки визначається за найбільшою кількістю пакетів у вікні. Правило виключає вікна, у яких атакуючий клієнт надсилає поодинокі службові пакети, наприклад після завершення атаки.

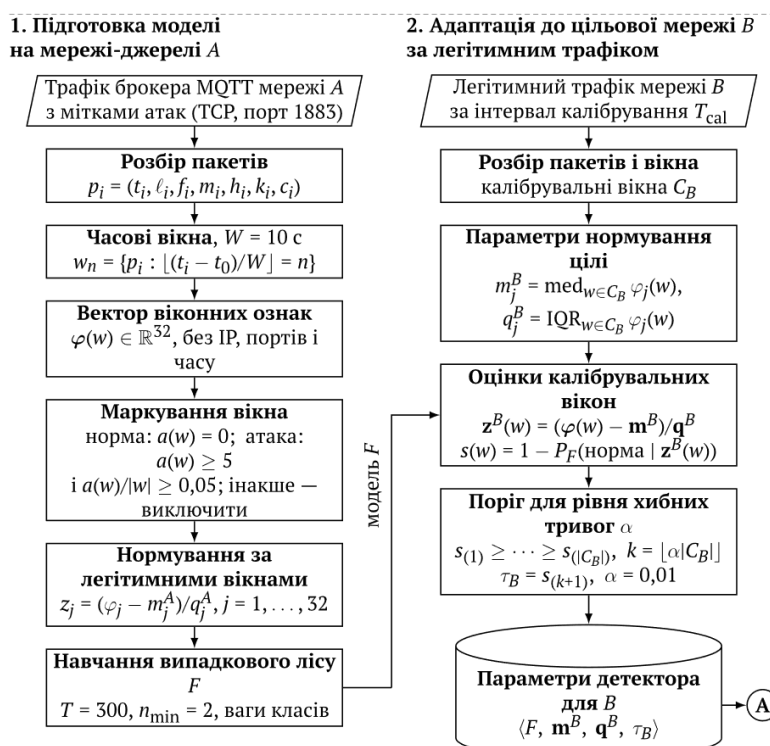


Рис. 1. Блок-схема удосконаленого методу: підготовка моделі на мережі-джерелі та адаптація до цільової мережі

Ознаки нормуються за легітимними вікнами: під час навчання – за легітимними вікнами мережі-джерела, під час роботи – за калібрувальними легітимними вікнами C_B цільової мережі:

$$z_j(w) = \frac{\varphi_j(w) - m_j^B}{q_j^B}, \quad m_j^B = \text{med}_{w \in C_B} \varphi_j(w), \quad q_j^B = \text{IQR}_{w \in C_B} \varphi_j(w), \quad (6)$$

де med – медіана; IQR – міжквартильний розмах (за нульового розмаху використовується стандартне



відхилення, а за його відсутності – одиниця). Нормування за легітимним трафіком переводить ознаки в одиниці «відхилення від звичайної поведінки мережі», що усуває відмінності між стендами в інтенсивності й складі легітимних клієнтів. Класифікатором є випадковий ліс з T дерев, навчений на нормованих вікнах мережі-джерела з урахуванням ваг класів. Оцінка атаки для вікна обчислюється як

$$s(w) = 1 - \frac{1}{T} \sum_{t=1}^T P_t(\text{норма} | z(w)), \quad (7)$$

де P_t – частка легітимних вікон у листі дерева t , до якого потрапляє вектор $z(w)$. Поріг рішення визначається за оцінками калібрувальних вікон, упорядкованими за спаданням $s_{(1)} \geq s_{(2)} \geq \dots \geq s_{|C_B|}$:

$$\tau_B = s_{(k+1)}, \quad k = \lfloor \alpha |C_B| \rfloor, \quad \hat{y}(w) = \text{атака} \Leftrightarrow s(w) > \tau_B. \quad (8)$$

Правило (8) враховує однакові значення оцінок і гарантує, що частка калібрувальних вікон з тривогою не перевищує α . Адаптація до нової мережі зводиться до обчислення векторів m^B , q^B і порога τ_B та не потребує повторного навчання моделі й атакуювальних міток цільової мережі.

Робота детектора на IoT-шлюзі (рис. 2) є потоковою: для кожного пакета оновлюються накопичувачі поточного вікна (лічильники, суми довжин і їх квадратів, максимум, буфер довжин для процентиля), а при переході до нового вікна обчислюються ознаки, нормування, оцінка лісу й рішення (8). Стан вікна займає 296 байт, а обчислювальна складність лінійна за кількістю пакетів.

Відтворення відомих підходів

Результати відтворення наведено в табл. 4. Для MQTTset використано код авторів набору та поділ train70/test30 [20], для MQTT-IoT-IDS2020 – код авторів [11] (10 частин даних, поділ 75/25 і п'ятикратна перехресна перевірка), для підходу [1] – пакетні ознаки зі стратифікованою вибіркою тих самих обсягів, поділ 60/40 і випадкове дублювання прикладів меншого класу в навчальній вибірці, для підходу [12] – дерево рішень на пакетних ознаках MQTTset з адресою джерела й без неї.

Таблиця 4

Опубліковані й відтворені показники відомих підходів

Підхід	Модель	Опубліковано, %	Відтворено, %	Додаткові показники відтворення
[20], MQTTset	випадковий ліс	99,43	99,61	макро-F1 0,773; повнота: flood 0,484, malformed 0,468, підбір 0,738
[20], MQTTset	дерево рішень	97,80	97,98	макро-F1 0,635
[11], односпрям. потоки	випадковий ліс	99,98	99,997	дерево рішень: 99,96 $\rightarrow$ 99,994
[1], підбір облікових даних MQTT	логістична регресія	100	98,91	без портів і TTL: хибні тривоги 8,9 %
[12], MQTTset	дерево рішень	92,57	92,59	з адресою джерела: 97,50 %

Відтворені значення близькі до опублікованих, що підтверджує коректність відтворення. Водночас за точності 99,61 % на MQTTset макроусереднена F1-міра становить 0,773, а повнота для класів flood і malformed не перевищує 0,49, оскільки легітимні пакети становлять 98,6 % тестової вибірки. За узгодженого кодування ознак у навчальній і тестовій вибірках точність становить 99,72 %, макро-F1 – 0,825. Вектори ознак 97,4 % тестових пакетів MQTTset збігаються з векторами навчальної вибірки, тобто за випадкового поділу перевірка значною мірою відбувається на тих самих векторах. Для MQTT-IoT-IDS2020 сканер використовує фіксований порт джерела (35087 у 17 000 потоках сценарію агресивного сканування), а для MQTTset атаки записано з окремих адрес; додавання адреси джерела зменшує частку хибних тривог дерева рішень з 1,34 % до 0,03 %.

Пакетні ознаки за хронологічного поділу і в новій мережі

За хронологічного поділу на тому самому стенді пакетний детектор (випадковий ліс на шести спільних полях) не забезпечує заданого рівня хибних тривог: для MQTTset поріг, визначений на ранній частині запису, на пізнішій дає частку хибних тривог 37,0 %; для MQTTEEB-D однакові вектори ознак трапляються в легітимних і атакуювальних пакетах, і жоден поріг не дає повноти, більшої за нуль, за частки хибних тривог до 1 %; для MQTT-IoT-IDS2020 повнота становить 0,300. У новій мережі з порогом мережі-джерела середня за 12 парами повнота становить 0,392 за частки хибних тривог 41,2 %, а середня площа під ROC-кривою – 0,570. Навіть за найкращого порога, визначеного на легітимному трафіку цільової мережі, середня повнота пакетного детектора становить лише 0,124 (медіана – 0).

3. Поточкова обробка трафіку на IoT-шлюзі

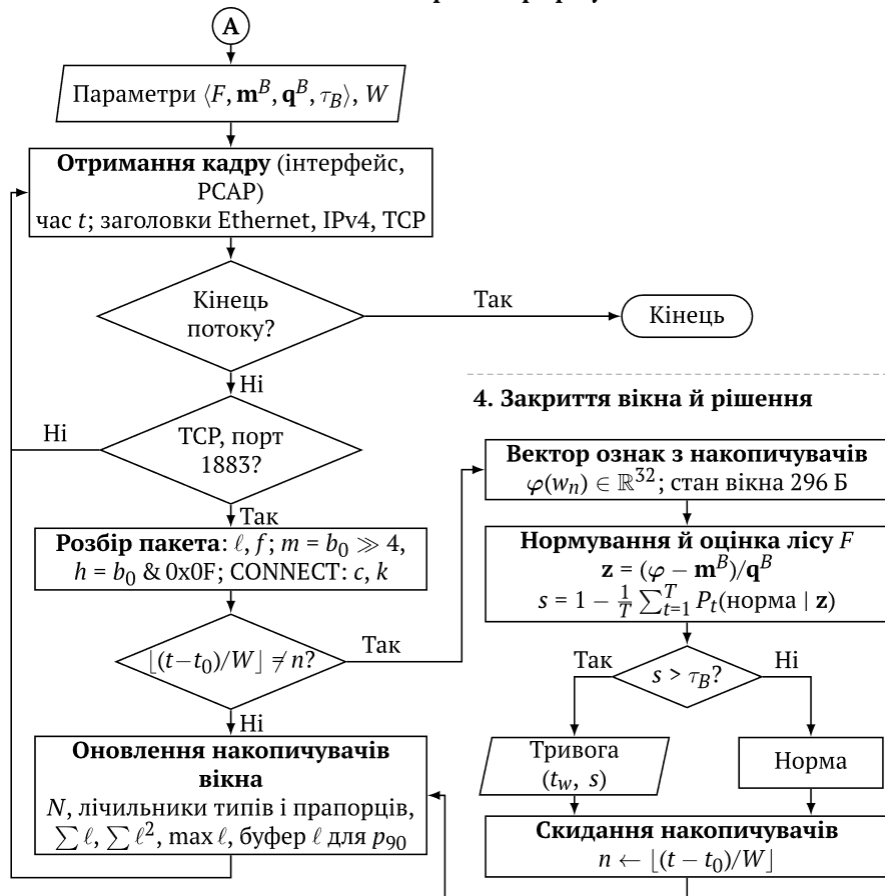


Рис. 2. Блок-схема поточної роботи детектора на IoT-шлюзі

Удосконалений метод у новій мережі

Повноту виявлення для 12 пар мереж за рівня хибних тривог 1 % наведено на рис. 3, середні значення – у табл. 5. Для порівняння розглянуто ті самі віконні ознаки з порогом мережі-джерела, ті самі ознаки з калібруванням лише порога за легітимним трафіком цільової мережі, детектор IsolationForest [15], навчений лише на легітимних вікнах цільової мережі, а також гібрид, що поєднує оцінки удосконаленого методу й IsolationForest (тривога, якщо будь-яка з двох оцінок перевищує свій процентиль; поріг визначається на окремій половині калібрувальних вікон).

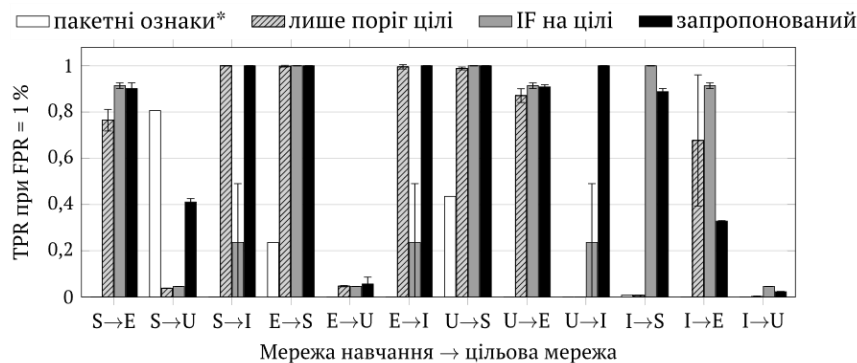


Рис. 3. Повнота виявлення атак за рівня хибних тривог 1 % для пар «мережа навчання – цільова мережа» (S – MQTTset, I – MQTT-IoT-IDS2020, E – MQTTEEB-D, U – MQTT_UAD; * – пакетні ознаки найкращим порогом цільової мережі; відрізки – стандартне відхилення за 5 запусками)

Середні показники для 12 пар «мережа навчання – цільова мережа»
($W = 10$ с, калібрування – 30 % часу)

Підхід	TPR, середнє	TPR, медіана	FPR, середнє
Пакетні ознаки, поріг мережі-джерела	0,392	–	0,412
Пакетні ознаки, найкращий поріг цільової мережі	0,124	0,000	0,010
Віконні ознаки, поріг мережі-джерела	0,819	–	0,705
Віконні ознаки, калібрування лише порога	0,532	0,721	0,004
IsolationForest на легітимному трафіку цільової мережі	0,549	0,575	0,006
Гібрид удосконаленого методу й IsolationForest	0,564	0,637	0,006
Удосконалений метод	0,710	0,905	0,010
Довідково: навчання й перевірка на тому самому стилі	0,929	0,993	0,006

Без калібрування поріг мережі-джерела не переноситься: частка хибних тривог у цільовій мережі становить у середньому 70,5 %, у восьми парах – 100 %. Калібрування лише порога забезпечує частку хибних тривог 0,4 % за середньої повноти 0,532. Нормування ознак і порога за легітимним трафіком цільової мережі підвищує середню повноту до 0,710 (медіана – 0,905) за фактичної частки хибних тривог 1,0 %. Удосконалений метод кращий за калібрування лише порога в 7 парах і гірший в одній (I – E: 0,328 проти 0,677). IsolationForest, який не використовує знань про атаки, кращий на цільових мережах MQTTset і MQTTEEB-D, проте на MQTT-IoT-IDS2020 його повнота становить 0,236 проти 1,000 удосконаленого методу, а на MQTT_UAD з мережі-джерела MQTTset – 0,046 проти 0,411. Гібрид не дав переваги через насичення емпіричної функції розподілу на зменшеному калібрувальному наборі. Найнижчі значення для всіх підходів отримано на MQTT_UAD, де атаки типу MitM і вторгнення відсутні в мережах-джерелах, а атаки DoS представлені лише чотирма тестовими вікнами.

7. Стійкість до параметрів методу

Стійкість результату перевірено для розміру вікна $W \in \{5; 10; 30\}$ с і частки часу калібрування $\{0,1; 0,3; 0,5\}$ (рис. 4). Удосконалений метод має найвищу середню повноту в усіх дев'яти конфігураціях: 0,619–0,677 для $W = 5$ с, 0,704–0,721 для $W = 10$ с і 0,703–0,777 для $W = 30$ с. Для 108 випадків (9 конфігурацій $\times$ 12 пар) він кращий за калібрування лише порога в 60 випадках і гірший у 12, кращий за IsolationForest у 70 випадках і гірший у 23. Перевага найбільша за малого обсягу калібрування: за 10 % часу повнота IsolationForest становить 0,16–0,36, тоді як удосконаленого методу – 0,62–0,78, оскільки знання про атаки з мережі-джерела компенсує нестачу даних цільової мережі.

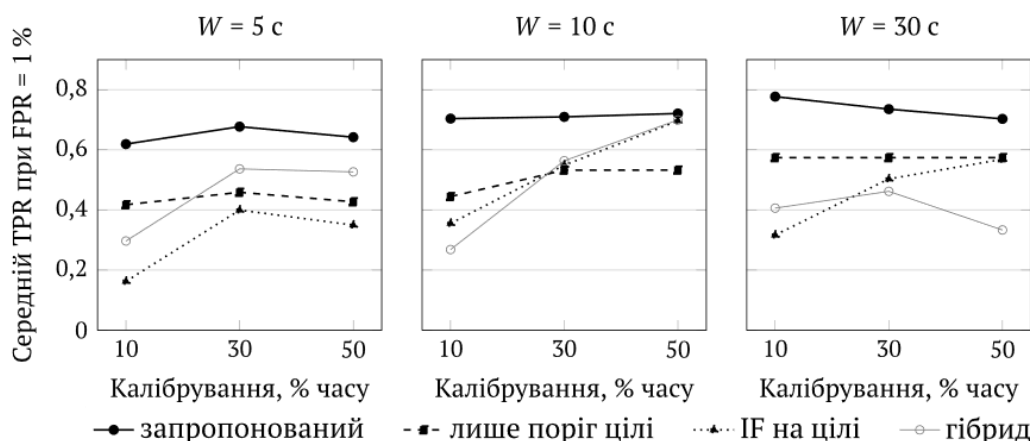


Рис. 4. Середня повнота виявлення за рівня хибних тривог 1 % залежно від частки часу калібрування для різних розмірів вікна

Обмеженням є точність забезпечення рівня хибних тривог за малої кількості калібрувальних вікон: за $W = 30$ с (22 калібрувальні вікна для MQTT-IoT-IDS2020 і 30 для MQTT_UAD) фактична частка хибних тривог становить 1,7–3,6 %, а за $W = 10$ с і калібрування 10 % часу – 2,0 %. Для $W = 10$ с і калібрування 30–50 % часу фактична частка хибних тривог становить 1,0–1,1 %, тому рекомендовано використовувати не менше 100–200 калібрувальних вікон.

**Програмна реалізація для IoT-шлюзу**

Метод реалізовано мовою С як модуль, що читає трафік із файлів PCAP/PCAPNG або мережевого інтерфейсу, розбирає заголовки Ethernet, Linux SLL, IPv4, TCP і MQTT, накопичує стан вікна й виконує нормування, оцінювання й порівняння з порогом (рис. 2). Модель для шлюзу – випадковий ліс з 50 дерев глибиною до 12, експортований у таблиці вузлів; на парі MQTTset – MQTT-IoT-IDS2020 її показники збігаються з показниками моделі з 300 дерев (TPR 1,000 за FPR 0,000). Режим калібрування обчислює m^B , q^B і τ_B безпосередньо на шлюзі за записом легітимного трафіку. Вимірювання виконано на ноутбучі (Windows 11, один потік, компілятор gcc 16.2 з оптимізацією -O2); результати наведено в табл. 6.

Відповідність програмного модуля еталонній реалізації мовою Python перевірено на 669 вікнах легітимного запису: найбільша відносна розбіжність ознак становить $6 \cdot 10^{-8}$, оцінок моделі – $5,6 \cdot 10^{-17}$, рішення збігаються для всіх вікон. Отже, метод придатний для потокової роботи на IoT-шлюзі: обробка займає сотні наносекунд на пакет, а модель і стан вікна вміщуються в десятки кілобайтів пам'яті.

Таблиця 6

Обчислювальні витрати програмного модуля мовою С (5 запусків)

Показник	MQTT-IoT-IDS2020, легітимний запис	MQTT-IoT-IDS2020, запис атаки
Кількість пакетів	957 233	10 010 556
Час обробки, нс на пакет (з читанням файлу)	272–401	181–209
Пропускна здатність, млн пакетів/с	2,5–3,7	4,8–5,5
Час оцінювання вікна, мкс	0,69–1,36	2,1–2,5
Пікова пам'ять процесу, МБ	3,6–3,8	4,5
Розмір моделі, КБ / стан вікна, Б	20,7 / 296	20,7 / 296
Хибні тривоги після калібрування	0 з 469 вікон	–
Виявлені вікна активної фази атаки	–	128 з 129

Обговорення результатів

Отримані результати показують, що показники, отримані за випадкового поділу записів одного стенда, не характеризують роботу детектора в новій мережі: ті самі пакетні ознаки, що дають точність понад 99 % у таких умовах, у новій мережі за рівня хибних тривог 1 % виявляють у середньому 12,4 % атак. Віконні протокольні ознаки відображають поведінку клієнтів відносно брокера (наприклад, частку відмов у підключенні та інтенсивність пакетів CONNECT під час підбору облікових даних або тривалі інтервали keepAlive під час атаки SlowITe), тому вони менше залежать від конкретного стенда. Нормування за легітимним трафіком цільової мережі враховує відмінності легітимної поведінки між мережами: у MQTT-IoT-IDS2020 сенсори відкривають новий сеанс для кожного повідомлення, а в MQTTset підтримують постійне з'єднання.

До обмежень дослідження належать: накладання записів атак на легітимний трафік для двох наборів, у яких атаки записано окремо; відсутність у мережах-джерелах атак типу MitM і вторгнення, наявних у MQTT_UAD; залежність точності забезпечення рівня хибних тривог від кількості калібрувальних вікон; вимірювання обчислювальних витрат на ноутбучі, а не на вбудованому шлюзі. Ці обмеження визначають напрями подальших досліджень.

ВИСНОВКИ ТА ПЕРСПЕКТИВИ ПОДАЛЬШИХ ДОСЛІДЖЕНЬ

У статті удосконалено метод виявлення атак на MQTT-сегмент IoT-мережі, який, на відміну від відомих, ґрунтується на віконних протокольних ознаках трафіку брокера, незалежних від мережових ідентифікаторів і абсолютного часу, та на нормуванні цих ознак і порога рішення за легітимним трафіком цільової мережі без використання її атакуювальних міток, що забезпечує підвищення повноти виявлення атак у новій мережі за заданого рівня хибних тривог.

Відтворення чотирьох відомих підходів на наборах MQTTset і MQTT-IoT-IDS2020 дало значення, близькі до опублікованих (92,6–99,99 %), однак у новій мережі пакетні ознаки за рівня хибних тривог 1 % забезпечують середню повноту не більше 0,124. Для 12 пар мереж, сформованих із чотирьох незалежних наборів даних, удосконалений метод забезпечує середню повноту виявлення 0,710 (медіана 0,905) за фактичної частки хибних тривог 1,0 % проти 0,532 для калібрування лише порога й 0,549 для детектора, навченого лише на легітимному трафіку цільової мережі. Перевага зберігається для розміру вікна 5–30 с і обсягу калібрування 10–50 % часу та найбільша за малого обсягу легітимних даних цільової мережі. Програмна реалізація мовою С для IoT-шлюзу обробляє трафік за 180–400 нс на пакет за розміру моделі 21 КБ, а налаштування на нову мережу виконується на шлюзі лише за записом її легітимного трафіку.

Практичне значення отриманих результатів полягає в можливості встановлення детектора атак у



нову мережу розумного будинку без розмічених атак цієї мережі із заданим рівнем хибних тривог. Перспективами подальших досліджень є навчання на кількох мережах-джерелах, виявлення типів атак, відсутніх у мережах-джерелах, онлайн-перекалібрування за змін легітимного трафіку та перевірка на вбудованих IoT-шлюзах.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Alasmari, R., & Alhogail, A. A. (2024). Protecting smart-home IoT devices from MQTT attacks: An empirical study of ML-based IDS. *IEEE Access*, 12, 25993–26004. <https://doi.org/10.1109/ACCESS.2024.3367113>
2. Alatram, A., Sikos, L. F., Johnstone, M., Szewczyk, P., & Kang, J. J. (2023). DoS/DDoS-MQTT-IoT: A dataset for evaluating intrusions in IoT networks using the MQTT protocol. *Computer Networks*, 231, Article 109809. <https://doi.org/10.1016/j.comnet.2023.109809>
3. Apruzzese, G., Laskov, P., & Schneider, J. (2023). SoK: Pragmatic assessment of machine learning for network intrusion detection. In *2023 IEEE 8th European Symposium on Security and Privacy (EuroS&P)* (pp. 592–614). IEEE. <https://doi.org/10.1109/EuroSP57164.2023.00042>
4. Aqachtoul, A., Karam, K., Elamrani, A., Najib, M., Rafalia, N., & Bakhouya, M. (2025). MQTTEEB-D: A real-world IoT cybersecurity dataset for AI-powered threat detection in MQTT networks. *Data in Brief*, 62, Article 111897. <https://doi.org/10.1016/j.dib.2025.111897>
5. Arp, D., Quiring, E., Pendlebury, F., Warnecke, A., Pierazzi, F., Wressnegger, C., Cavallaro, L., & Rieck, K. (2022). Dos and don'ts of machine learning in computer security. In *Proceedings of the 31st USENIX Security Symposium* (pp. 3971–3988). USENIX Association.
6. Aveleira-Mata, J., Alaiz-Moretón, H., Bayón-Gutiérrez, M., García-Ordás, M. T., Prieto-Fernández, N., & García-Rodríguez, I. (2025). MQTT_UAD: MQTT Under Attack Dataset. A public dataset for the detection of attacks in IoT networks using MQTT protocol. *Data in Brief*, 63, Article 112167. <https://doi.org/10.1016/j.dib.2025.112167>
7. Cantone, M., Marrocco, C., & Bria, A. (2024). Machine learning in network intrusion detection: A cross-dataset generalization study. *IEEE Access*, 12, 144489–144508. <https://doi.org/10.1109/ACCESS.2024.3472907>
8. Cerasuolo, F., Bovenzi, G., & Pescapè, A. (2026). Cross-network transferability of AI-based network intrusion detection systems in heterogeneous Internet of Things environments. *Computer Networks*, 286, Article 112434. <https://doi.org/10.1016/j.comnet.2026.112434>
9. Dadkhah, S., Neto, E. C. P., Ferreira, R., Molokwu, R. C., Sadeghi, S., & Ghorbani, A. A. (2024). CICIoMT2024: A benchmark dataset for multi-protocol security assessment in IoMT. *Internet of Things*, 28, Article 101351. <https://doi.org/10.1016/j.iot.2024.101351>
10. Flood, R., Engelen, G., Aspinall, D., & Desmet, L. (2024). Bad design smells in benchmark NIDS datasets. In *2024 IEEE 9th European Symposium on Security and Privacy (EuroS&P)* (pp. 658–675). IEEE. <https://doi.org/10.1109/EuroSP60621.2024.00042>
11. Hindy, H., Bayne, E., Bures, M., Atkinson, R., Tachtatzis, C., & Bellekens, X. (2021). Machine learning based IoT intrusion detection system: An MQTT case study (MQTT-IoT-IDS2020 dataset). In *Selected papers from the 12th International Networking Conference* (Lecture Notes in Networks and Systems, Vol. 180, pp. 73–84). Springer. https://doi.org/10.1007/978-3-030-64758-2_6
12. Imran, Zuhairi, M. F., Ali, S. M., Shahid, Z., Alam, M. M., & Su'ud, M. M. (2024). Realtime feature engineering for anomaly detection in IoT based MQTT networks. *IEEE Access*, 12, 25700–25718. <https://doi.org/10.1109/ACCESS.2024.3363889>
13. Khan, M. A., Khan, M. A., Jan, S. U., Ahmad, J., Jamal, S. S., Shah, A. A., Pitropakis, N., & Buchanan, W. J. (2021). A deep learning-based intrusion detection system for MQTT enabled IoT. *Sensors*, 21(21), Article 7016. <https://doi.org/10.3390/s21217016>
14. KouhiRonaghi, A., & Habibi Lashkari, A. (2026). MQTTFlowLyzer: Interpretable TabNet-based flow-level MQTT intrusion detection for IoT. *The Journal of Supercomputing*, 82(5), Article 334. <https://doi.org/10.1007/s11227-026-08485-7>
15. Liu, F. T., Ting, K. M., & Zhou, Z.-H. (2008). Isolation forest. In *2008 Eighth IEEE International Conference on Data Mining* (pp. 413–422). IEEE. <https://doi.org/10.1109/ICDM.2008.17>
16. Nguyen, X.-H., Nguyen, X.-D., Huynh, H.-H., & Le, K.-H. (2022). Realguard: A lightweight network intrusion detection system for IoT gateways. *Sensors*, 22(2), Article 432. <https://doi.org/10.3390/s22020432>
17. Omotosho, A., Qendah, Y., & Hammer, C. (2023). IDS-MA: Intrusion detection system for IoT MQTT



- attacks using centralized and federated learning. In 2023 *IEEE 47th Annual Computers, Software, and Applications Conference (COMPSAC)* (pp. 678–688). IEEE. <https://doi.org/10.1109/COMPSAC57700.2023.00093>
18. Roy, U. S., & Minu, M. J. (2026). *Unmasking shortcut learning in IoT intrusion detection: A forensic, multi-paradigm evaluation of feature dependence and data leakage* [Preprint]. arXiv. <https://doi.org/10.48550/arXiv.2609.28725>
 19. Ullah, S., Ahmad, J., Khan, M. A., Alshehri, M. S., Boulila, W., Koubaa, A., Jan, S. U., & Iqbal Ch, M. M. (2023). TNN-IDS: Transformer neural network-based intrusion detection system for MQTT-enabled IoT networks. *Computer Networks*, 237, Article 110072. <https://doi.org/10.1016/j.comnet.2023.110072>
 20. Vaccari, I., Chiola, G., Aiello, M., Mongelli, M., & Cambiaso, E. (2020). MQTTset, a new dataset for machine learning techniques on MQTT. *Sensors*, 20(22), Article 6578. <https://doi.org/10.3390/s20226578>
 21. Ніщенко, Д. О., & Волошук, О. Б. (2025). Адаптивне очищення різнорідних сенсорних даних у системах розумного будинку на основі класифікації шумів. *Кібербезпека: освіта, наука, техніка*, 4(28), 740–750. <https://doi.org/10.28925/2663-4023.2025.28.844>
 22. Жебка, В. В., Аронов, А. О., Ніщенко, Д. О., & Ананченко, О. Є. (2026). Удосконалення методу багатоцільової оптимізації керування IoT-системою розумного будинку з урахуванням кіберризиків. *Кібербезпека: освіта, наука, техніка*, 2(34), 820–830. <https://doi.org/10.28925/2663-4023.2026.34.1389>

**Nishchemenko Dmytro**

Doctor of Philosophy, Senior Lecturer, Department of Digital Development Technologies,
State University of Information and Communication Technologies, Kyiv, Ukraine
ORCID: 0009-0006-1781-4109
d.nishchemenko@duikt.edu.ua

Salo Nataliia

PhD in Engineering, Senior Lecturer, Department of Air Navigation Services
Ukrainian State Flight Academy, Kropyvnytskyi, Ukraine
ORCID: 0000-0003-1422-9853
nataliiaa.salo@gmail.com

Nesterenko Kateryna

State University of Information and Communication Technologies, Kyiv
ORCID: 0000-0001-7672-7386
k.nesterenko@duikt.edu.ua

Mamchenko Serhij

Doctor of Technical Sciences, Professor, Department of Computer Systems, Networks and Cybersecurity
National University of Life and Environmental Sciences of Ukraine, Kyiv, Ukraine
Borys Grinchenko Kyiv Metropolitan University, Kyiv, Ukraine
ORCID: 0009-0006-8743-5606
s.mamchenko@nubip.edu.ua

IMPROVING THE METHOD OF ATTACK DETECTION IN THE MQTT SEGMENT OF A SMART HOME IoT NETWORK WITH ADAPTATION TO A NEW NETWORK USING LEGITIMATE TRAFFIC

Abstract. The MQTT protocol is the main means of message exchange between sensors, actuators and the gateway of a smart home, so attacks on the MQTT broker directly affect the availability and integrity of the IoT system. Known machine-learning methods for detecting such attacks report an accuracy of 99–100% when training and testing are performed on traffic from the same testbed, whereas in practice a detector is deployed in a network whose traffic was not used for training and must operate at a given false alarm rate. The paper improves the method of attack detection in the MQTT segment of an IoT network: it relies on window-level protocol features of broker traffic that do not depend on network identifiers and absolute time, and on normalising the features and the decision threshold by the legitimate traffic of the target network without using its attack labels: the features are normalised by the median and the interquartile range of legitimate calibration windows, and the threshold is set as an order statistic of their scores for the given false alarm rate. The research methodology uses four independent datasets (MQTTset, MQTT-IoT-IDS2020, MQTTEEB-D, MQTT_UAD): reproduction of known approaches under their authors' conditions, evaluation of packet features with a chronological split and in another network, and evaluation of the improved method for 12 "training network – target network" pairs at a false alarm rate of 1%. The improved method achieves a mean attack detection rate of 0.710 (median 0.905) versus 0.532 for threshold-only calibration, 0.549 for a detector trained only on legitimate traffic of the target network and at most 0.124 for packet features; the result holds for window sizes of 5–30 s and calibration on 10–50% of the time. The method is implemented in C as an IoT gateway module that processes traffic in 180–400 ns per packet with a 21 KB model.

Keywords: cybersecurity; Internet of Things; MQTT; intrusion detection; smart home; IoT gateway; cross-network transfer; threshold calibration.

REFERENCES (TRANSLATED AND TRANSLITERATED)

1. Alasmari, R., & Alhogail, A. A. (2024). Protecting smart-home IoT devices from MQTT attacks: An empirical study of ML-based IDS. *IEEE Access*, 12, 25993–26004. <https://doi.org/10.1109/ACCESS.2024.3367113>



2. Alatram, A., Sikos, L. F., Johnstone, M., Szewczyk, P., & Kang, J. J. (2023). DoS/DDoS-MQTT-IoT: A dataset for evaluating intrusions in IoT networks using the MQTT protocol. *Computer Networks*, 231, Article 109809. <https://doi.org/10.1016/j.comnet.2023.109809>
3. Apruzzese, G., Laskov, P., & Schneider, J. (2023). SoK: Pragmatic assessment of machine learning for network intrusion detection. In *2023 IEEE 8th European Symposium on Security and Privacy (EuroS&P)* (pp. 592–614). IEEE. <https://doi.org/10.1109/EuroSP57164.2023.00042>
4. Aqachtoul, A., Karam, K., Elamrani, A., Najib, M., Rafalia, N., & Bakhouya, M. (2025). MQTTEEB-D: A real-world IoT cybersecurity dataset for AI-powered threat detection in MQTT networks. *Data in Brief*, 62, Article 111897. <https://doi.org/10.1016/j.dib.2025.111897>
5. Arp, D., Quiring, E., Pendlebury, F., Warnecke, A., Pierazzi, F., Wressnegger, C., Cavallaro, L., & Rieck, K. (2022). Dos and don'ts of machine learning in computer security. In *Proceedings of the 31st USENIX Security Symposium* (pp. 3971–3988). USENIX Association.
6. Aveleira-Mata, J., Alaiz-Moretón, H., Bayón-Gutiérrez, M., García-Ordás, M. T., Prieto-Fernández, N., & García-Rodríguez, I. (2025). MQTT_UAD: MQTT Under Attack Dataset. A public dataset for the detection of attacks in IoT networks using MQTT protocol. *Data in Brief*, 63, Article 112167. <https://doi.org/10.1016/j.dib.2025.112167>
7. Cantone, M., Marrocco, C., & Bria, A. (2024). Machine learning in network intrusion detection: A cross-dataset generalization study. *IEEE Access*, 12, 144489–144508. <https://doi.org/10.1109/ACCESS.2024.3472907>
8. Cerasuolo, F., Bovenzi, G., & Pescapè, A. (2026). Cross-network transferability of AI-based network intrusion detection systems in heterogeneous Internet of Things environments. *Computer Networks*, 286, Article 112434. <https://doi.org/10.1016/j.comnet.2026.112434>
9. Dadkhah, S., Neto, E. C. P., Ferreira, R., Molokwu, R. C., Sadeghi, S., & Ghorbani, A. A. (2024). CICIoMT2024: A benchmark dataset for multi-protocol security assessment in IoMT. *Internet of Things*, 28, Article 101351. <https://doi.org/10.1016/j.iot.2024.101351>
10. Flood, R., Engelen, G., Aspinall, D., & Desmet, L. (2024). Bad design smells in benchmark NIDS datasets. In *2024 IEEE 9th European Symposium on Security and Privacy (EuroS&P)* (pp. 658–675). IEEE. <https://doi.org/10.1109/EuroSP60621.2024.00042>
11. Hindy, H., Bayne, E., Bures, M., Atkinson, R., Tachtatzis, C., & Bellekens, X. (2021). Machine learning based IoT intrusion detection system: An MQTT case study (MQTT-IoT-IDS2020 dataset). In *Selected papers from the 12th International Networking Conference* (Lecture Notes in Networks and Systems, Vol. 180, pp. 73–84). Springer. https://doi.org/10.1007/978-3-030-64758-2_6
12. Imran, Zuhairi, M. F., Ali, S. M., Shahid, Z., Alam, M. M., & Su'ud, M. M. (2024). Realtime feature engineering for anomaly detection in IoT based MQTT networks. *IEEE Access*, 12, 25700–25718. <https://doi.org/10.1109/ACCESS.2024.3363889>
13. Khan, M. A., Khan, M. A., Jan, S. U., Ahmad, J., Jamal, S. S., Shah, A. A., Pitropakis, N., & Buchanan, W. J. (2021). A deep learning-based intrusion detection system for MQTT enabled IoT. *Sensors*, 21(21), Article 7016. <https://doi.org/10.3390/s21217016>
14. KouhiRonaghi, A., & Habibi Lashkari, A. (2026). MQTTFlowLyzer: Interpretable TabNet-based flow-level MQTT intrusion detection for IoT. *The Journal of Supercomputing*, 82(5), Article 334. <https://doi.org/10.1007/s11227-026-08485-7>
15. Liu, F. T., Ting, K. M., & Zhou, Z.-H. (2008). Isolation forest. In *2008 Eighth IEEE International Conference on Data Mining* (pp. 413–422). IEEE. <https://doi.org/10.1109/ICDM.2008.17>
16. Nguyen, X.-H., Nguyen, X.-D., Huynh, H.-H., & Le, K.-H. (2022). Realguard: A lightweight network intrusion detection system for IoT gateways. *Sensors*, 22(2), Article 432. <https://doi.org/10.3390/s22020432>
17. Omotosho, A., Qendah, Y., & Hammer, C. (2023). IDS-MA: Intrusion detection system for IoT MQTT attacks using centralized and federated learning. In *2023 IEEE 47th Annual Computers, Software, and Applications Conference (COMPSAC)* (pp. 678–688). IEEE. <https://doi.org/10.1109/COMPSAC57700.2023.00093>
18. Roy, U. S., & Minu, M. J. (2026). Unmasking shortcut learning in IoT intrusion detection: A forensic, multi-paradigm evaluation of feature dependence and data leakage [Preprint]. arXiv. <https://doi.org/10.48550/arXiv.2609.28725>
19. Ullah, S., Ahmad, J., Khan, M. A., Alshehri, M. S., Boulila, W., Koubaa, A., Jan, S. U., & Iqbal Ch, M. M. (2023). TNN-IDS: Transformer neural network-based intrusion detection system for MQTT-enabled IoT networks. *Computer Networks*, 237, Article 110072. <https://doi.org/10.1016/j.comnet.2023.110072>
20. Vaccari, I., Chiola, G., Aiello, M., Mongelli, M., & Cambiaso, E. (2020). MQTTset, a new dataset for machine learning techniques on MQTT. *Sensors*, 20(22), Article 6578. <https://doi.org/10.3390/s20226578>



21. Nishchemenko, D., & Voloshchuk, O. (2025). Adaptive cleaning of heterogeneous sensor data in smart home systems based on noise classification. *Cybersecurity: Education, Science, Technique*, 4(28), 740–750. <https://doi.org/10.28925/2663-4023.2025.28.844>
22. Zhebka, V. V., Aronov, A. O., Nishchemenko, D. O., & Ananchenko, O. Ye. (2026). Improvement of the multi-objective optimization method for controlling a smart home IoT system, considering cyber risk. *Cybersecurity: Education, Science, Technique*, 2(34), 820–830. <https://doi.org/10.28925/2663-4023.2026.34.1389>

Отримано редакцією журналу / Received: 06.09.26

Прорецензовано / Revised: 18.09.26

Схвалено до друку / Accepted: 26.10.26



This work is licensed under Creative Commons Attribution-noncommercial-sharealike 4.0 International License.