https://doi.org/10.31891/2219-9365-2025-83-55
УДК 004.03:004.45:004.75

СТЕЦЮК Юрій
Хмельнитцький національнитй університет
https://orcid.org/0000-0003-0312-2276
e-mail: yuriy.stetsuk@khmnu.edu.ua

МЕТОД РАНДОМНОЇ ПЕРЕДАЧІ КЕРУВАННЯ МІЖ МЕРЕЖЕВИМИ ВУЗЛАМИ З ОПТИМІЗАЦІЄЮ ЦЕНТРАЛІЗАЦІЇ РЕСУРСІВ БЕЗПЕКИ

Проведено аналіз сучасного стану розвитку підсистем безпеки ОС. Приділено увагу принципам побудови централізованих систем безпеки мережевих ОС на основі динамічної передачі керування між мережевими вузлами. Це забезпечує підвищення стійкості до витоків конфіденційної інформації в результаті деструктивних впливів зловмисного програмного забезпечення та комп'ютерних атак. Представлено опис процесу динамічної передачі керування між вузлами комп'ютерної мережі, розглянуто механізми формування централізованих баз безпекових ресурсів. Також розглянуто питання оптимізації безпекових ресурсів, що підлягають централізованому керуванню зі сторони поточного керівного вузла мережі при використанні динамічної передачі керування між вузлами комп'ютерної мережі.

Представлено стратегії формування глобальних баз привілеїв, політик безпеки, мережевих з'єднань при кожному циклі передачі керування від поточного вузла комп'ютерної мережі до наступного.

Проведено кілька серій експериментів з мережею віртуальних машин під керуванням ОС FreeBSD 13.1, результати яких підтвердили теоретичні розрахунки та математичного моделювання.

Виконано порівняльний аналіз ефективності повної централізації ресурсів безпеки та їх часткової централізації при динамічній передачі керування між вузлами комп'ютерної мережі. Переваги запропонованого підходу проявились в суттєвому зменшені часу передачі керування між мережевими вузлами, зменшення поверхні атаки через мінімізацію точок впливу зі сторони зловмисного програмного забезпечення на систему безпеки.

Ключові слова: операційна система, централізована система безпеки, частково- централізована система безпеки

STETSYUK Yuriy
Khmelnytskyi National University

METHOD OF RANDOM CONTROL TRANSFER BETWEEN NETWORK NODES WITH OPTIMIZATION OF SECURITY RESOURCE CENTRALIZATION

The current state of development of OS security subsystems is analyzed. Attention is paid to the principles of building centralized security systems for network OSes based on dynamic control transfer between network nodes. This provides increased resistance to leaks of confidential information as a result of the destructive effects of malicious software and computer attacks. A description of the process of dynamic control transfer between computer network nodes is presented, and mechanisms for forming centralized security resource databases are considered. The issue of optimizing security resources that are subject to centralized control by the current network control node when using dynamic control transfer between computer network nodes is also considered.

Strategies for forming global privilege databases, security policies, and network connections during each control transfer cycle from the current computer network node to the next are presented.

Several series of experiments with a network of virtual machines running the FreeBSD 13.1 OS have been conducted, the results of which were confirmed by theoretical calculations and mathematical modeling.

A comparative analysis of the effectiveness of full centralization of security resources and their partial centralization during dynamic control transfer between computer network nodes was performed. The advantages of the proposed approach were manifested in a significant reduction in the time of control transfer between network nodes, a reduction in the attack surface due to the minimization of points of influence of malicious software on the security system.

Keywords: operating system, centralized security system, partially centralized security system

Стаття надійшла до редакції / Received 11.07.2025
Прийнята до друку / Accepted 14.08.2025

ПОСТАНОВКА ПРОБЛЕМИ У ЗАГАЛЬНОМУ ВИГЛЯДІ ТА ЇЇ ЗВ'ЯЗОК ІЗ ВАЖЛИВИМИ НАУКОВИМИ ЧИ ПРАКТИЧНИМИ ЗАВДАННЯМИ

Технологічні здобутки людства стикаються з новими масштабними викликами - загрозами інформаційній безпеці, які на сьогодні стали постійним фоном цифрового життя. Тепер успішність у будь-якій сфері - чи то наука, чи економіка, чи безпека - визначається рівнем доступу до достовірної інформації, швидкістю її обробки та здатністю захистити її від зовнішнього втручання.

Актуальність задачі забезпечення захисту інформації та підвищення стійкості інформаційних систем до витоку конфіденційної інформації, не зважаючи на масштабні зусилля планетарної економіки, не спадає. Проведений аналіз стану захисту інформаційних систем показує, що ця задача і сьогодні залишається надзвичайно актуальною. Не зважаючи на те, що ця проблема знаходиться в постійному фокусі уваги наукової спільноти та велику кількість запропонованих нею методів та засобів захисту інформаційних систем, вона не може вважатись вирішеною.

ПОСТАНОВКА ПРОБЛЕМИ

Реалізація динамічної передачі керування між вузлами комп'ютерної мережі дає ряд технічних та організаційних переваг, а саме в покращені гнучкості керування мережею, підвищені стійкості до збоїв та протидії зловмисному програмному забезпечені. Однак отримання максимальної ефективності цього методу тісно пов'язано з багатьма чинниками, визначальним з яких є оптимальне структурування та передача новому керівному вузлу критично важливих ресурсів безпеки. Формування оптимальної множини глобальних ресурсів безпеки, які необхідно передавати новому керівному вузлу в момент зміни центру керування є ключовим аспектом успішної реалізації динамічного керування комп'ютерною мережею. Вирішення цієї задачі дозволить уникнути зайвого дублювання, зменшити об'єм даних, що підлягають передачі і, відповідно, знизити час на їх синхронізацію.

Тому основним завданням цього дослідження є вирішення задачі забезпечення максимальної ефективності динамічної передачі керування між мережевими вузлами за рахунок оптимізації ресурсів безпеки, що підлягають централізованому керуванню. Спробі наукового вирішення цієї задачі і присвячена дана робота.

АНАЛІЗ ДОСЛІДЖЕНЬ ТА ПУБЛІКАЦІЙ

ОС пройшли кілька технологічних етапів у своєму розвитку. Разом з ними цей шлях подолали і їх системи безпеки, поки дійсно досягли такого рівня, що може гарантувати захист оброблюваної в комп'ютерних системах, що знаходяться під їх керуванням, інформації. І це все в умовах постійного, все зростаючого впливу зловмисного програмного забезпечення, яке в свою чергу безупинно удосконалюється. Розвиваючись під його безперервним тиском системи безпеки ОС на сьогодні являють собою потужні підсистеми керування, які включають велику номенклатуру високо ефективних захисних механізмів. Архітектура систем безпеки ОС, особливо мережевих, пройшла шлях від децентралізованої до складно організованої централізованої, що охоплює своїм керуванням всі ресурси сучасної комп'ютерної системи. Так в [1] представлено модель, що описує взаємодію захисних механізмів у рамках централізованої системи безпеки для ОС мережевого вузла, яка дозволяє побудувати архітектуру безпеки мережевої ОС, в якій мінімізується проблема витоку конфіденційної інформації під час атак на оперативну пам'ять системи. Пошук збалансованої архітектури системи безпеки ОС, що ефективно забезпечує стійкість ОС розглядається в [2].

Багато праць присвячено методам покращення роботи мережевих систем реагування на інциденти (IDS\IPS) та підходам які можуть бути застосовані для підвищення рівня безпеки в мережах в плані протидії витоку конфіденційної інформації [3]. В [4] пропонується система виявлення вторгнень у хмарі (C-IDS), яка адаптується до середовища кожного користувача. Ще одни напрямком досліджень стали IDS з підтримкою машинного навчання (ML) для захисту мереж. В [5] пропонуються засоби підвищення ефективності IDS – систем, що до аналізу та кластеризації мережевого трафіку. Нова метрика для оцінки IDS-систем, яка враховує затримку у виявленні кібератак представлена в [6]. Запропонована в [7] IDS використовує оптимізований блок одновимірної згорткової нейронної мережі та достатню пам'ять для автоматичного вилучення просторових та часових ознак з вихідних даних.

Також, багато уваги приділено стратегіям та механізмам захисту мережевих ОС в частині протидії зловмисному програмному забезпеченню. Самоадаптивна система на основі SVM для забезпечення стійкості мережі до кібератак ботнетів під назвою BotGRABBER розглянута в [8]. Ще одна самоадаптивна система, в якій стійкість забезпечується адаптивною реконфігурацією мережі представлена в [9]. Також запропоновано новий метод виявлення DDoS-ботнетів на основі аналізу їх мережевих характеристик, а в [10] на використанні алгоритмів штучних імунних систем.

Ще одним підходом в забезпечені мережевої безпеки є зберігання даних на основі архітектури напівмережевої ОС [11]. Пропонуються інтегровані системи управління, засновані на моделях загроз та ризиків [12]. Багатообіцяючою є ідея створення безпечної ОС з контрольованою складністю [13]. В [14] започатковано ще один напрямок дослідження в основі якого безсерверна безпека, що володіє високим потенціалом надійних захисних заходів.

Цікавим є рішення доповнення системи безпеки ОС механізмом ізоляції пам'яті, який не дозволяє обійти його, шляхом віртуалізації блоку керування пам'яттю [15]. Також протидії атакам на пам'ять (MCA), що змінюють вміст деяких областей пам'яті, з метою порушення нормальної роботи обчислювальних систем, спричиняючи витік конфіденційних даних або збурення в поточних процесах присвячені роботи [16]. У [17] досліджуються атаки на сучасні гетерогенні вбудовані обчислювальні платформи FPGA-SoC, які містить найсучасніші механізми ізоляції пам'яті та периферійних пристроїв. В [18] запропоновано модель безпеки, яка забезпечує вищий рівень захисту порівняно з такими існуючими підходами, як рекурентні нейронні мережі та метод опорних векторів.

Безпека ОС на різних апаратних платформах є фундаментальною метою поточної оцінки. В [19] запропоновано формальний метод верифікації до криптомодуля ОС RIOT для аналізу програмного забезпечення для коду Frama-C, з метою забезпечення його аспектів безпеки. В [20] досліджуються причини вразливостей, зокрема для систем IoT, IIoT, SCADA та Android для вбудованих систем та пропонується

впровадження ефективних стратегій виявлення шкідливого програмного забезпечення. Робота [21] акцентує на проблемі порушення рандомізації розміщення адресного простору ядра (KASLR) на сучасних мобільних пристроях, пов'язане з проблемою з порушенням KASLR на процесорах ARM. В пошуках архітектур ОС, стійких до витоків інформації в [22] пропонується інструмент для дослідження та моделювання прототипу системи з мікроядерною архітектурою, яка забезпечує високу надійність та масштабованість.

Цікава система автентифікації запропонована в [23], що основана на протоколі, який включає модуль безпеки Linux для NAC із завантаженням підписаних правил та ключів для користувача з USB-ключа безпеки, з контролем мережевих дозволів безпосередньо з ядра Linux.

В [24] піднімається питання мінімізації надання привілеїв доступу до чутливих ресурсів, як способу підвищення стійкості до витоків інформації з метою унеможливлення використання зловмисниками вразливостей файлових систем та драйверів дисків для організації витоку конфіденційної інформації.

Відмічається, що традиційні системи безпеки ОС в основі статичних методів захисту не завжди справляються з дедалі складнішими атаками [25]. З метою подолання проблеми пропонується впровадження проактивної та адаптивної систем безпеки [26]. Ще одним напрямком забезпечення захисту інформації в багато комп'ютерних системах є зміна керівного центру. Метод визначення наступного варіанту централізації без втручання користувача пропонується в [27].

Таким чином задача протидії витоку конфіденційної інформації в мережевих ОС, залишається не повністю вирішеною. Тут і неповна централізація контролю доступу до ресурсів, складність протидії багатовекторним загрозам, відсутність універсального підходу узгодження політик з різних форматів та вузлів, що призводить до прогалин у захисті.

ФОРМУЛЮВАННЯ ЦІЛЕЙ СТАТТІ

Дана стаття спрямована на вирішення проблем стійкості ОС до витоків конфіденційної інформації, що обробляється в комп'ютерних системах. Її метою є розробка моделей процесу рандомної динамічної передачі керування між вузлами мережі, їх дослідження в напрямку оптимізації ресурсів безпеки мережевої ОС, що підлягають централізованому управлінню, що в свою чергу, дозволить зменшити витрати часу ОС на передачу керування між вузлами і, тим самим, забезпечити безперервність та стабільність керування мережею, що в кінцевому рахунку, призведе до підвищення стійкості ОС до витоку інформації.

ВИКЛАД ОСНОВНОГО МАТЕРІАЛУ

Передача керування від одного мережевого вузла до іншого, випадково обраного є досить складно організованим та ресурсо затратним процесом. Одним із питань вирішуваних при цьому є визначення об'ємів передачі ресурсів безпеки, або степені їх централізації. Для забезпечення стабільного й безпечного функціонування комп'ютерної мережі з динамічною рандомною передачею керування між її вузлами, безперервності управління новому керівному вузлу необхідно передавати ряд критичних системних даних, які забезпечують цілісність політик безпеки. В тому числі і політики маркування конфіденційної інформації, мережеві політики (робота Firewall, ACL), політики доступу до спільних ресурсів, порядок дій з передачі керування між вузлами, локальні журнали подій, параметри конфігурування обладнання, локальні кеші сесій, фонові служби безпеки, списки авторизованих локальних процесів, низькорівневі драйвери та інші. Для зменшення витрат часу на передачу керування між вузлами мережі, пропонується частину не критичних ресурсів безпеки, залишити в управлінні локальних вузлів. З метою описання процесу динамічної передачі керування між вузлами комп'ютерної мережі з розділенням централізованого та локального управління ресурсами, введемо позначення для ключових аспектів. Множину вузлів мережі позначимо як: $N = \{N_1, N_2, \dots, N_n\}$, де $N_1, N_2, \dots, N_n$ – вузли мережі. Множину ресурсів безпеки вузлів визначимо як: $R_{BR} = \{R_1, R_2, \dots, R_k\}$, де $R_1, R_2, \dots, R_k$ – безпекові ресурси (бази привілеїв, політики безпеки, база з'єднань і таке інше).

Ресурси, що підлягають централізованому керуванню визначимо як її підмножину $R_{centr} \in R_{BR}$. Тоді підмножину ресурсів, що можуть залишатись на локальному управлінні можна визначити як R_{local}, як таку, що не є пересічною з підмножиною R_{centr}. Для опису процесу динамічної передачі керування між вузлами комп'ютерної мережі також введемо необхідні позначення. Поточний керівний вузол в момент часу t визначимо як $CSMM_t$. При цьому $CSMM_t$ приналежить до множини вузлів мережі N: $CSMM_t \in N$. Тоді імовірність передачі керування від поточного мережевого вузла $CSMM_t$ до наступного $CSMM_{t+1}$, що прийме керування мережею в момент часу $t+1$ можна визначити так:

$$P\left(CSMM_{t+1} = N_j \,\middle|\, CSMM_t = N_i\right), \tag{1}$$

де $CSMM_t$ та $CSMM_{t+1}$ – керівні вузли комп'ютерної мережі в моменти часу t та $t+1$ відповідно з множини мережевих вузлів N; N_i – поточний керівний вузол в момент часу t; N_j – індекс наступного керівного

вузла з множини N в момент часу $t+1$, при цьому на передачу керування накладається обмеження $j \neq i$ (рис. 1 позиція 1).

В силу вище приведеного, можна зробити висновок, що має місце рівномірний розподіл імовірності стати керівним вузлом мережі в процесі динамічної передачі керування для кожного активного вузла мережі, тобто кожен вузол має однаковий шанс стати центральним керівним вузлом мережі. Це можна представити формулою:

$$P\left(CSMM_{t+1} = N_j\right) = \frac{1}{n-1}, \forall j \neq i, \quad (2)$$

де N_j – індекс наступного керівного вузла $CSMM_{t+1}$ з множини N в момент часу $t+1$; n – загальне число активних мережевих вузлів у множині N; $n-1$ – враховує, що поточний керівний вузол не може знову стати керівним вузлом мережі ($j \neq i$).

Вибраний закон розподілу імовірності мережевого вузла стати центральним керівним вузлом мережі дозволяє гарантувати випадковість та рівномірність передачі керування, дозволяє уникнути перевантаження мережевого вузла, підвищує стійкість до атак через ускладнення для зловмисника передбачити, хто буде наступним керівним вузлом, підтримує ідею часткової централізації - жоден вузол не залишається постійним центром. Процес передачі ініціюється поточним керівним вузлом $CSMM_t$ шляхом посилки керівного токена $T_{Governance}$ рандомно вибраному активному мережевому вузлу (рис. 1 позиція 1). Після отримання керівного токена (рис. 1 позиція 2) новий керівний вузол $CSMM_{t+1}$ відправляє запит (рис. 1 позиція 3) до всіх активних мережевих вузлів для передачі йому ресурсів безпеки, які потребують централізованого керування: глобальної бази привілеїв - має бути одна узгоджена копія для всіх вузлів, політик маркування конфіденційних даних – вони впливають на логіку доступу, мережевих політик – потребують єдиної стратегії безпеки, графіка передачі керування між вузлами – його порушення може викликати колізії в управлінні мережею, політик обробки інцидентів безпеки – також потребують централізованого реагування та аналізу. На практиці це означає, шо такі ресурси ОС як /etc/security/policies.conf, /etc/firewall.rules, global_privileges.db є їх мінімально можливою множиною, що потребує централізованого керування. Таке розмежування управління дає змогу зменшити трафік, прискорити передачу керування між вузлами і водночас зберегти високу стійкість до витоків конфіденційної інформації.

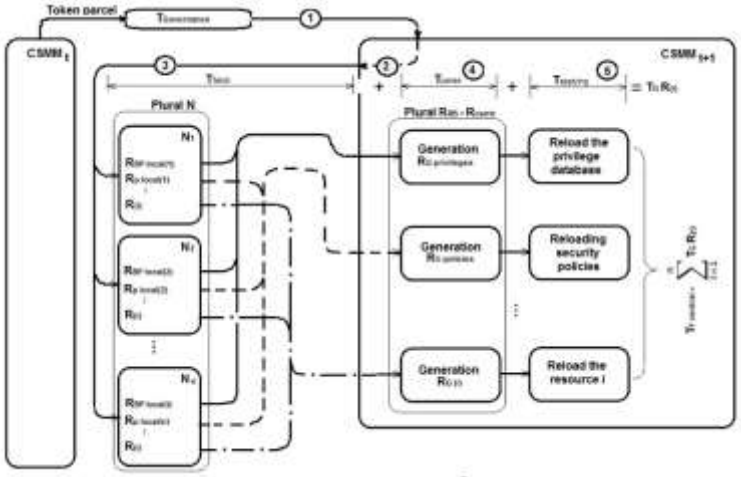

Рис. 1: Модель процесу динамічної передачі керування мережевими вузлами з повною централізацією ресурсів безпеки системи безпеки мережевої ОС

Так актуальна централізована база привілеїв global_privileges.db для наступного керівного вузла $CSMM_{t+1}$ отримується шляхом об'єднання локальних баз привілеїв активних мережевих вузлів (рис. 1 позиція 4). Для представлення цього механізму введемо наступні позначення: $R_{BP\ local}$ – база привілеїв деякого локального вузла. Тоді глобальну базу привілеїв $R_{G\ privileges}$ можна представити так:

$$R_{G\ privileges} = \bigcup_{i \in A} R_{BP\ local}(i), \quad (3)$$

де $R_{BP\ local}(i)$ – локальна база привілеїв i-го мережевого вузла, де i належить до множини A активних на момент передачі керування вузлів мережі.

Розглянемо формування глобальної бази політик безпеки global_policies.conf яка є ще одним централізованим ресурсом, що оновлюється при кожному циклі передачі керування між вузлами. Для зручності описання введемо наступні позначення: $R_{P_i(x)}$ – політика безпеки x i-го вузла мережі як ресурс

безпеки; $R_{P\ local}$ – політики безпеки деякого локального вузла. Її можна визначити як об'єднання всіх політик безпеки даного локального вузла комп'ютерної мережі:

$$R_{P\ local} = \cup_{i=1}^{n}\{R_{P_i(x)}\}, \quad (4)$$

де $R_{P_i(x)}$- політика безпеки x i-го вузла; n – число мережевих вузлів.

Наступним кроком є отримання саме глобальної бази політик безпеки комп'ютерної мережі. При цьому слід зазначити, що при реалізації функції об'єднання всіх локальних баз політик безпеки можливе застосування кількох механізмів, що реалізують різні правила доступу, списки дозволених дій, тощо і, з яких необхідно вибрати найбільш оптимальний, який забезпечить максимальну стійкість системи до витоку інформації. Це може бути консенсусний підхід, оснований на включені до глобальної бази політик безпеки $R_{G\ policies}$ всіх політик безпеки локальних активних вузлів. Ще один - зважене об'єднання, що враховує рівень довіри до вузлів. Оскільки мова іде за формування централізованих баз ресурсів безпеки системи безпеки мережевої ОС, то найбільш доцільним і оправданим бачиться застосування механізму, основаного на перетині політик безпеки локальних баз мережевих вузлів, оскільки він досить простий в реалізації і в той же час забезпечує режим найсуворіших правил застосування політик безпеки зі сторони центрального керівного вузла, що в свою чергу гарантує підвищену стійкість ОС до витоків конфіденційної інформації, що обробляється в системі під її керуванням.

Тоді глобальну базу політик безпеки $R_{G\ policies}$ можна представити так:

$$R_{G\ policies} = \cap_{i\in A}\{R_{p\ local}(i)\}, \quad (5)$$

де $R_{p\ local}(i)$ – локальна база політик безпеки i-го вузла комп'ютерної мережі, де i належить до множини активних вузлів мережі A в момент передачі керування.

Для формування глобальної бази з'єднань $R_{G\ firewall}$ скористаємось тим же механізмом, що і при формуванні глобальної бази політик безпеки $R_{G\ policies}$, оскільки управління з'єднаннями також в силу своєї важливості, потребує найсуворіших правил контролю. Тоді $R_{G\ firewall}$ визначиться так:

$$R_{G\ firewall} = \cap_{i\in A}\{R_{FW\ local}(i)\}, \quad (6)$$

де $R_{FW\ local}(i)$ – локальна база з'єднань i-го вузла комп'ютерної мережі, де i належить до множини активних вузлів мережі A в момент передачі керування.

Обґрунтування ефективності часткової централізації ресурсів безпеки. Для цього виконаємо порівняльний розрахунок витрат при динамічній передачі керування між вузлами комп'ютерної мережі для ситуацій з повною централізацією всіх ресурсів безпеки системи безпеки мережевої ОС та централізації лише баз привілеїв, політик безпеки та мережевих з'єднань.

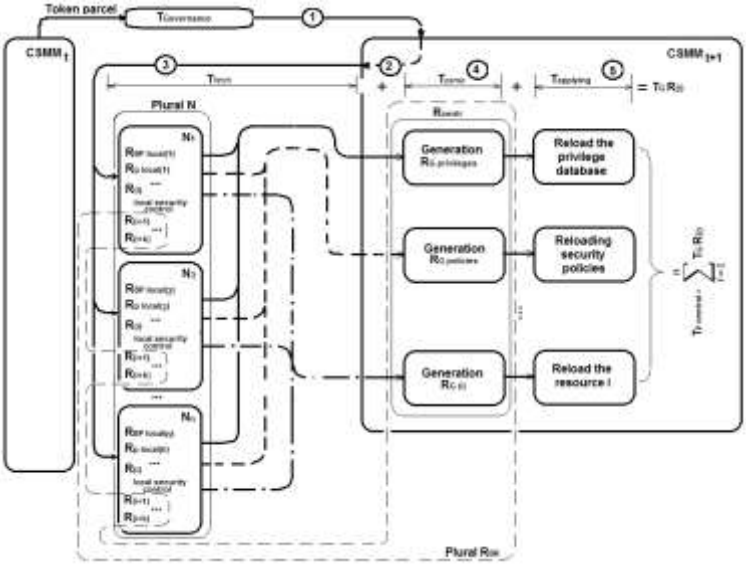

Рис. 2: Модель процесу динамічної передачі керування мережевими вузлами з частковою централізацією ресурсів безпеки системи безпеки мережевої ОС

Для математичного представлення загальних витрат на передачу керування між вузлами мережі скористаємось раніше введеними позначеннями: R_{BR} – множина безпекових ресурсів мережі; $R_{centr} \in R_{BR}$ – підмножина ресурсів, що підлягають централізованому керуванню для ситуації часткової централізації, а також введемо позначення порівняльних критеріїв: T_{fetch} - час на опитування всіх активних вузлів мережі та отримання від них копій ресурсу R_i для формування його актуальної глобальної бази; T_{parse} – час, який новий керівний вузол витрачає на розбір, уніфікацію або підтвердження відповідності отриманих даних (перевірка цілісності бази, перевірка узгодженості політик безпеки, побудова узагальненої глобальної версії ресурсу); $T_{aplying}$ – час на перезапуск нової глобальної версії ресурсу на новому керівному вузлу (активації його в службах або підсистемах ОС, передача підтвердження іншим вузлам мережі у разі необхідності).

Тоді загальний час актуалізації глобальної версії деякого ресурсу безпеки R_i мережевої ОС можна представити як суму всіх часових періодів його формування:

$$T_G\,(R_i) = T_{fetch}(R_i) + T_{parse}(R_i) + T_{applying}(R_i), \tag{7}$$

де T_{fetch} - час на запит локальних копій ресурсу R_i з усіх активних вузлів мережі; R_i – час формування глобальної версії ресурсу R_i; $T_{applying}$ – час на перезапуск нової версії глобального ресурсу R_i.

Виконання формули (7) проілюстровано на рис. 1 (позиції 3, 4, 5) та рис. 2 (позиції 3, 4, 5), на яких представлені процеси підготовки глобальних баз ресурсів безпеки для наступного керівного $CSMM_{t+1}$ системи безпеки ОС з повною (рис. 1) та частковою (рис. 2) централізацією ресурсів.

Тепер для ситуації з повною централізацією всіх ресурсів безпеки мережевої ОС час передачі керування новому керівному вузлу $CSMM_{t+1}$ $T_{F\,central}$ можна представити як суму всіх часових періодів, необхідних для формування всіх глобальних версій всіх ресурсів безпеки ОС відповідно до схеми (рис. 1). З урахуваням сказаного та (формули 7) вона прийме вигляд:

$$T_{F\,central} = \sum_{i=1}^{n}(T_{fetch}(R_i) + T_{parse}(R_i) + T_{applying}(R_i)), \tag{8}$$

де T_{fetch} - час на запит локальних копій ресурсу R_i з усіх активних вузлів мережі; T_{parse}– час формування глобальної версії ресурсу R_i; $T_{applying}$ – час на перезапуск нової версії глобального ресурсу R_i; n – загальне число ресурсів безпеки ОС у множині R_{BP}.

Час передачі керування новому керівному вузлу $CSMM_{t+1}$ з частковою централізацією ресурсів безпеки $T_{P\,central}$ можна розрахувати скориставшись формулою (8) з тою лише різницею, що загальне число ресурсів безпеки ОС n буде відповідати числу елементів множини R_{centr}, яка є підмножиною множини R_{BP}, як це показано на рис. 2. При такій схемі роботи частина ресурсів безпеки залишаються у локальному управлінні мережевого вузла, відповідно, не потребують збору та централізації даних про них при передачі керування новому центральному вузлу $CSMM_{t+1}$, що значно зменшує витрати часу на саму передачу керування. Для їх мінімізації важливо, щоб число елементів множини R_{centr} було якомога меншим, але з іншої сторони таким, щоб забезпечувався необхідний рівень стійкості ОС до витоку конфіденційної інформації.

Тоді відношення часу передачі керування новому керівному вузлу з повною централізацією ресурсів безпеки $T_{F\,central}$ та часу передачі керування з частковою централізаціє ресурсів безпеки $T_{P\,central}$ більше одиниці покаже на кращу ефективність системи безпеки ОС з частковою централізацією (формула 9):

$$E_{central} = \frac{T_{F\,central}}{T_{P\,central}} > 1 \tag{9}$$

Для представлення величини ефективності $E_{P\,central}$ у відсотках формулу (9) представимо у такому виді (формула 10):

$$E_{P\,central} = \frac{T_{F\,central} - T_{P\,central}}{T_{F\,central}} \times 100\%, \tag{10}$$

де $T_{F\,central}$ - час передачі керування новому керівному вузлу з повною централізацією ресурсів безпеки; $T_{P\,central}$ - час передачі керування з частковою централізаціє ресурсів безпеки.

З метою підтвердження ефективності системи безпеки мережевої ОС з частковою централізацією ресурсів безпеки при динамічній передачі керування між мережевими вузлами було проведено кілька серій експериментів. Для чого було розгорнуто тестове середовище на базі віртуальної комп'ютерної мережі під управлінням гіпервізора Virtual Box. В якості мережевої ОС на кожній віртуальній машині встановлена ОС FreeBSD 13.1

В результаті проведених експериментів з симуляцією динамічної передачі керування між вузлами мережі були отримані дані, приведені в таблиці 1.

Таблиця 1.

Усереднені дані експериментів

Назва ресурсу безпеки	T_{fetch} сек	T_{parse}, сек	$T_{applying}$, сек	$T_G\,(R_i)$ сек	Централізація ресурсу
База привілегій	1,03	2,27	2,01	5,31	Так
Політики безпеки	5,22	1,59	2,73	9,54	Так
Мережеві з'єднання	3,44	1,28	3,49	8,21	Так
Аудит (Audit Logs)	3,432	1,144	1,562	6,14	Ні
Права доступу	0,6943	0,88	1,2314	2,81	Ні
Налаштування антивірусу	2,5545	1,2183	1,3755	5,15	Ні

Підставивши дані у формулу (8) для розрахунку часу передачі керування між вузлами комп'ютерної мережі для ситуації з повною централізацією ресурсів безпеки (з урахуванням $R_i \in R_{BP}$) отримаємо:

$$T_{F\ central} = \sum_{i=1}^{n} (T_{fetch}(R_i) + T_{parse}(R_i) + T_{applying}(R_i)) = (5{,}31 + 9{,}54 + 8{,}21 + 6{,}14 + 2{,}81 + 5{,}15) = 37{,}16 \text{ сек.}$$

На слідуючому кроці виконаємо розрахунок часу передачі керування між вузлами мережі для ситуації з частковою централізацією ресурсів (формула (8) з урахуванням $R_i \in R_{central}$) отримаємо:

$$T_{P\ central} = \sum_{i=1}^{n} (T_{fetch}(R_i) + T_{parse}(R_i) + T_{applying}(R_i)) = (5{,}31 + 9{,}54 + 8{,}21) = 23{,}06 \text{ сек.}$$

Скориставшись формулою (10) виконаємо розрахунок ефективності часткової централізації ресурсів безпеки при динамічній передачі керування між вузлами мережі в порівняні з повною централізацією:

$$E_{P\ central} = \frac{T_{F\ central} - T_{P\ central}}{T_{F\ central}} \times 100\% = \frac{37{,}16 - 23{,}06}{37{,}16} \times 100\% = 37{,}9\%.$$

Розрахунки по результатах проведених експериментів підтвердили ефективність методу оптимізації централізації ресурсів безпеки мережевої ОС при рандомній динамічній передачі керування між вузлами комп'ютерної мережі (Рис. 3).

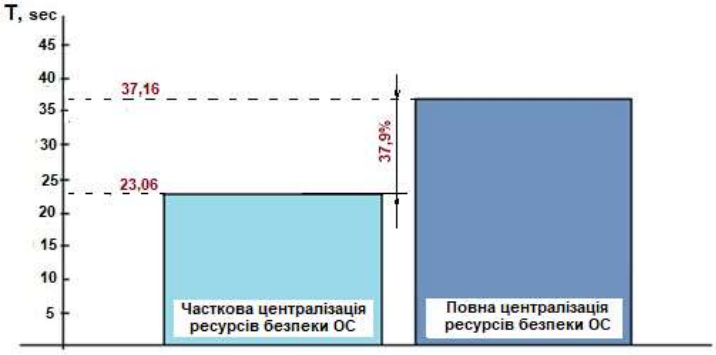

Рис. 3. Графік ефективності часткової централізації ресурсів безпеки при динамічній перелачі керування

Зменшення часу на передачу керування між вузлами комп'ютерної мережі при частковій централізації ресурсів безпеки склало, як видно з розрахунків ≈ 38% (рис. 3), що досягнуто за рахунок виключення з централізованої реплікації другорядних ресурсів безпеки (log-файли, локальні драйвери пристроїв тощо), зменшення обсягу даних для синхронізації, зменшення витрат на перевірку, нормалізацію та логування даних.

ВИСНОВКИ З ДАНОГО ДОСЛІДЖЕННЯ І ПЕРСПЕКТИВИ ПОДАЛЬШИХ РОЗВІДОК У ДАНОМУ НАПРЯМІ

Результатом запропонованого підходу є підтвердження суттєвого скорочення часу (рис. 3) при динамічній рандомній передачі керування між мережевими вузлами, що в свою чергу, зменшує ймовірність втрати керування мережею. Крім того, такий підхід дозволяє краще масштабувати систему, оскільки збільшення кількості вузлів непропорційно впливає на об'єм даних, що передаються новому керівному вузлу, завдяки оптимізації множини критичних ресурсів безпеки. Ще однією перевагою оптимізації множини глобальних ресурсів безпеки є зменшення поверхні атаки, оскільки зменшується число точок, через які зловмисник міг би вплинути на систему безпеки під час передачі керування. Це особливо важливо в умовах, коли атаки можуть бути скоординованими та спрямованими на моменти максимальної вразливості системи, а саме в моменти зміни керівного вузла. Такий підхід в побудові централізованих систем безпеки ОС дозволяє

зберігати баланс між гнучкістю динамічного керування та необхідною жорсткістю контролю над її критичними аспектами.

Література

1. Liu G. Improved Biba model based on trusted computing /Liu G., Zhang J., Liu J., Zhan Y.// Security and Communication Networks. -2015. -Vol. 8. -P. 2793 – 2797. https://doi.org/10.1002/sec.1201

2. Stetsyuk Y. A model of a centralized security system, as an information technology for the synthesis of an OS architecture protected against the leakage of confidential information /Y. Stetsyuk, M. Stetsyuk, V. Kyrylo, V. Paiuk, M. Kvassay// 1st International Workshop on Advanced Applied Information Technologies (AdvAIT-2024) vol. 3899, Khmelnytskyi Ukraine and Zilina Slovakia, -2024, -P. 224-233.

3. Leventopoulos S. Retaliating against cyber-attacks: a decision-taking framework for policy-makers and enforcers of international and cybersecurity law /S. Leventopoulos, K. Pipyros, D. Gritzalis// Int. Cybersecur. Law Rev, -2024 -№5 -P.237–262. doi:10.1365/s43439-024-00113-5.

4. Hajimirzaei B. Intrusion detection for cloud computing using neural networks and artificial bee colony optimization algorithm /B. Hajimirzaei, N.J. Navimipour// ICT Express, -2019 -№5(1) -P.56–59. doi:10.1016/j.icte.2018.01.014

5. Govindaram A.J. FLBC-IDS: a federated learning and blockchain-based intrusion detection system for secure IoT environments/A.J. Govindaram// Multimed Tools Appl -2025. -№84. –P.17229–17251. doi:10.1007/s11042-024-19777-6.

6. Liao D. GE-IDS: an intrusion detection system based on grayscale and entropy / D. Liao, R. Zhou, H. Li, // Peer-to-Peer Netw. Appl. 15 (2022) 1521–1534. doi:10.1007/s12083-022-01300-z.

7. Lopez-Vizcaino M. Time Aware F-Score for Cybersecurity Early Detection Evaluation / M. Lopez-Vizcaino, F.J. Novoa, D. Fernandez, F. Cacheda // BASEL: Mdpi -2024. –Vol. 14(2). P. 574. doi:10.3390/app14020574

8. Lysenko S. BotGRABBER: SVM-Based Self-Adaptive System for the Network Resilience Against the Botnets' Cyberattacks / S. Lysenko, K. Bobrovnikova, O. Savenko, A. Kryshchuk // Communications in Computer and Information Science. -2019. -Vol. 1039. –P.127 – 143. ISSN: 1865-0929.

9. Savenko B. "Malware Detection By Distributed Systems with Partial Centralizatio" /B. Savenko, A. Kashtalian, S. Lysenko and O. Savenko// 2023 IEEE 12th Int. Conference: Technology and Applications (IDAACS), Dortmund, Germany, -2023. -P. 265-270, doi: 10.1109/IDAACS58523.2023.10348773

10. Savenko O. DDoS Botnet Detection Technique Based on the Use of the Semi-Supervised Fuzzy c-Means Clustering / O. Savenko, K. Bobrovnikova, S. Lysenko // CEUR-WS volume 2104 (2018) 688-695.

11. Lysenko S. A Botnet Detection Approach Based on The Clonal Selection Algorithm / S. Lysenko, K. Bobrovnikova O. Savenko // 9th International Conference on Dependable Systems, Services and Technologies (DeSSerT-2018), Kyiv, Ukraine, May 24-27 2018, pp. 424-428.

12. Zhang Y.S. Analysis of OSPU security effect and data assembly verification under semi-network OS architecture/ Y.S. Zhang // Int. J. Inf. Secur -2023. -№22. –P. 1497–1509. doi: 10.1007/s10207-023-00702-1

13. Azura Y.T.Y. An integrated cyber security risk management framework for online banking systems / Y.T.Y. Azura, M.A. Azad, Y. Ahmed // J BANK FINANC TECHNOL -2025. doi: 10.1007/s42786-025-00056-3.

14. P.C. Pathak, M. Nadeem, S.A. Ansar Security assessment of operating system by using decision making algorithms / P.C. Pathak, M. Nadeem, S.A. Ansar // Int. j. inf. tecnol. -2025. -№17. P.3609–3618. doi: 10.1007/s41870-023-01706-9

15. Escaleira P. A systematic review on security mechanisms for serverless computing / P. Escaleira, V.A. Cunha, J.P. Barraca // Cluster Comput -2025. -№28. -P.465. doi: 10.1007/s10586-025-05371-4

16. Ma C. CToMP: a cycle-task-oriented memory protection scheme for unmanned systems / C. Ma, N. Xi, D. Lu // Sci. China Inf. Sci. 67 (2024) 162305. doi:10.1007/s11432-023-3865-0.

17. Herrera I. Survey of Techniques on Data Leakage Protection and Methods to address the Insider threat / I. Herrera Montano, J.J. García Aranda, J. Ramos Diaz // Cluster Comput -2022. -№25. P. 4289–4302. doi:10.1007/s10586-022-03668-2.

18. Bedratyuk L. Predicting Quantum-Chemical Descriptors for Small Molecules by Artificial Neural Networks / L. Bedratyuk, O. Savenko // MATCH Commun. Math. Comput. Chem. -2018. -№79. P.407–414.

19. Meganathan R. Security establishment using deep convolutional network model in cyber-physical systems / R. Meganathan, R. Anand // Multimed Tools Appl 83 (2024) 76201–76221. doi: 10.1007/s11042-024-18535-y

20. Rai N. Analysis of crypto module in RIOT OS using Frama-C / N. Rai, J. Grover // J Supercomput -2024. -№80 –P.18521–18543. doi: 10.1007/s11227-024-06171-0.

21. Kumar A. IoT device security audit tools: a comprehensive analysis and a layered architecture approach for addressing expanded security requirements / A. Kumar, L. Kavisankar, S. Venkatesan // Int. J. Inf. Secur. -2025. -№24. P.13. doi: 10.1007/s10207-024-00930-z

22. Seddigh M. Breaking KASLR on mobile devices without any use of cache memory / M. Seddigh, M. Esfahani, S. Bhattacharya//J Cryptogr Eng -2024.-№14.–P.281–294. doi:10.1007/s13389-023-00344-y.

23. Qian Z. A measurable refinement method of design and verification for micro-kernel operating

systems in communication network / Z. Qian, R. Xia, G. Sun, X. Xing, K. Xia // Digital communications and networks, -2023. -№10. -Vol 9(5), -P.1070-1079. doi: 10.1016/j.dcan.2022.03.024.

24. Cheng S.T. Per-user network access control kernel module with secure multifactor authentication / S.T. Cheng, G.J. Horng, C.W. Hsu // J Supercomput -2024. -№80. –P.970–1008. doi:10.1007/s11227-023-05480-0.

25. Cho C. Mandatory Access Control Method for Windows Embedded OS Security / C. Cho, Y. Seong, Y. Won// Electronics (Basel), -2021. -Vol.10(20), -P.2478. doi: 10.3390/electronics10202478

26. Derasari P. Autonomous Hardware-Based Proactive Defenses with Deep Reinforcement Learning / P. Derasari, G. Venkataramani// J Hardw Syst Secur -2025. doi:10.1007/s41635-025-00163-z.

27. Wu W. Deep learning-based stacked models for cyber-attack detection in industrial internet of things / W.Wu, H.Fouzi, B. Benamar// Neural Comput & Applic -2025. -№32. doi: 10.1007/s00521-025-11418-9

References

1. Liu G. Improved Biba model based on trusted computing / Liu G., Zhang J., Liu J., Zhan Y. // Security and Communication Networks. -2015. -Vol. 8. -P. 2793 – 2797. https://doi.org/10.1002/sec.1201

2. Stetsyuk Y. A model of a centralized security system, as an information technology for the synthesis of an OS architecture protected against the leakage of confidential information / Y. Stetsyuk, M. Stetsyuk, V. Kyrylo, V. Paiuk, M. Kvassay // 1st International Workshop on Advanced Applied Information Technologies (AdvAIT-2024) volume 3899, Khmelnytskyi Ukraine and Zilina Slovakia, -2024, -P. 224-233.

3. Leventopoulos S. Retaliating against cyber-attacks: a decision-taking framework for policy-makers and enforcers of international and cybersecurity law / S. Leventopoulos, K. Pipyros, D. Gritzalis // Int. Cybersecur. Law Rev, -2024 -№5 -P.237–262. doi:10.1365/s43439-024-00113-5.

4. Hajimirzaei B. Intrusion detection for cloud computing using neural networks and artificial bee colony optimization algorithm / B. Hajimirzaei, N.J. Navimipour // ICT Express, -2019 -№5(1) -P.56–59. doi:10.1016/j.icte.2018.01.014

5. Govindaram A.J. FLBC-IDS: a federated learning and blockchain-based intrusion detection system for secure IoT environments/ A.J. Govindaram // Multimed Tools Appl -2025. -№84. –P.17229–17251. doi:10.1007/s11042-024-19777-6.

6. Liao D. GE-IDS: an intrusion detection system based on grayscale and entropy / D. Liao, R. Zhou, H. Li, // Peer-to-Peer Netw. Appl. 15 (2022) 1521–1534. doi:10.1007/s12083-022-01300-z.

7. Lopez-Vizcaino M. Time Aware F-Score for Cybersecurity Early Detection Evaluation / M. Lopez-Vizcaino, F.J. Novoa, D. Fernandez, F. Cacheda // BASEL: Mdpi -2024. –Vol. 14(2). P. 574. doi:10.3390/app14020574

8. Lysenko S. BotGRABBER: SVM-Based Self-Adaptive System for the Network Resilience Against the Botnets' Cyberattacks / S. Lysenko, K. Bobrovnikova, O. Savenko, A. Kryshchuk // Communications in Computer and Information Science. -2019. -Vol. 1039. –P.127 – 143. ISSN: 1865-0929.

9. Savenko B. "Malware Detection By Distributed Systems with Partial Centralizatio" /B. Savenko, A. Kashtalian, S. Lysenko and O. Savenko// 2023 IEEE 12th Int. Conference: Technology and Applications (IDAACS), Dortmund, Germany, -2023. -P. 265-270, doi: 10.1109/IDAACS58523.2023.10348773

10. Savenko O. DDoS Botnet Detection Technique Based on the Use of the Semi-Supervised Fuzzy c-Means Clustering / O. Savenko, K. Bobrovnikova, S. Lysenko // CEUR-WS volume 2104 (2018) 688-695.

11. Lysenko S. A Botnet Detection Approach Based on The Clonal Selection Algorithm / S. Lysenko, K. Bobrovnikova O. Savenko // 9th International Conference on Dependable Systems, Services and Technologies (DeSSerT-2018), Kyiv, Ukraine, May 24-27 2018, pp. 424-428.

12. Zhang Y.S. Analysis of OSPU security effect and data assembly verification under semi-network OS architecture/ Y.S. Zhang // Int. J. Inf. Secur -2023. -№22. –P. 1497–1509. doi: 10.1007/s10207-023-00702-1

13. Azura Y.T.Y. An integrated cyber security risk management framework for online banking systems / Y.T.Y. Azura, M.A. Azad, Y. Ahmed // J BANK FINANC TECHNOL -2025. doi: 10.1007/s42786-025-00056-3.

14. P.C. Pathak, M. Nadeem, S.A. Ansar Security assessment of operating system by using decision making algorithms / P.C. Pathak, M. Nadeem, S.A. Ansar // Int. j. inf. tecnol. -2025. -№17. P.3609–3618. doi: 10.1007/s41870-023-01706-9

15. Escaleira P. A systematic review on security mechanisms for serverless computing / P. Escaleira, V.A. Cunha, J.P. Barraca // Cluster Comput -2025. -№28. -P.465. doi: 10.1007/s10586-025-05371-4

16. Ma C. CToMP: a cycle-task-oriented memory protection scheme for unmanned systems / C. Ma, N. Xi, D. Lu // Sci. China Inf. Sci. 67 (2024) 162305. doi:10.1007/s11432-023-3865-0.

17. Herrera I. Survey of Techniques on Data Leakage Protection and Methods to address the Insider threat / I. Herrera Montano, J.J. García Aranda, J. Ramos Diaz // Cluster Comput -2022. -№25. P. 4289–4302. doi:10.1007/s10586-022-03668-2.

18. Bedratyuk L. Predicting Quantum-Chemical Descriptors for Small Molecules by Artificial Neural Networks / L. Bedratyuk, O. Savenko // MATCH Commun. Math. Comput. Chem. -2018. -№79. P.407–414.

19. Meganathan R. Security establishment using deep convolutional network model in cyber-physical systems / R. Meganathan, R. Anand // Multimed Tools Appl 83 (2024) 76201–76221. doi: 10.1007/s11042-024-18535-y

20. Rai N. Analysis of crypto module in RIOT OS using Frama-C / N. Rai, J. Grover // J Supercomput -2024. -№80 –P.18521–18543. doi: 10.1007/s11227-024-06171-0.

21. Kumar A. IoT device security audit tools: a comprehensive analysis and a layered architecture approach for addressing expanded security requirements / A. Kumar, L. Kavisankar, S. Venkatesan // Int. J. Inf. Secur. -2025. -№24. P.13. doi: 10.1007/s10207-024-00930-z

22. Seddigh M. Breaking KASLR on mobile devices without any use of cache memory / M. Seddigh, M. Esfahani, S. Bhattacharya//J Cryptogr Eng -2024.-№14.–P.281–294. doi:10.1007/s13389-023-00344-y.

23. Qian Z. A measurable refinement method of design and verification for micro-kernel operating systems in communication network / Z. Qian, R. Xia, G. Sun, X. Xing, K. Xia // Digital communications and networks, -2023. -№10. -Vol 9(5), -P.1070-1079. doi: 10.1016/j.dcan.2022.03.024.

24. Cheng S.T. Per-user network access control kernel module with secure multifactor authentication / S.T. Cheng, G.J. Horng, C.W. Hsu // J Supercomput -2024. -№80. –P.970–1008. doi:10.1007/s11227-023-05480-0.

25. Cho C. Mandatory Access Control Method for Windows Embedded OS Security / C. Cho, Y. Seong, Y. Won// Electronics (Basel), -2021. -Vol.10(20), -P.2478. doi: 10.3390/electronics10202478

26. Derasari P. Autonomous Hardware-Based Proactive Defenses with Deep Reinforcement Learning / P. Derasari, G. Venkataramani// J Hardw Syst Secur -2025. doi:10.1007/s41635-025-00163-z.

27. Wu W. Deep learning-based stacked models for cyber-attack detection in industrial internet of things / W.Wu, H.Fouzi, B. Benamar// Neural Comput & Applic -2025. -№32. doi: 10.1007/s00521-025-11418-9