DOI 10.31891/2307-5732-2024-335-3-59
УДК 004.4

МЕЛЬНИК ВІКТОРІЯ
Хмельницький національний університет
https://orcid.org/0009-0009-7668-4318
e-mail: gutsalykvm@gmail.com
КРИВАК ДЕНИС
Хмельницький національний університет
https://orcid.org/0009-0009-6611-6552
e-mail: denys.kryvak@gmail.com
ВОЗНИЙ КИРИЛО
Хмельницький національний університет
https://orcid.org/0009-0007-2545-565X
e-mail: k.vozniy@gmail.com
ГУРАЛЬНИК ОЛЕКСАНДР
Хмельницький національний університет
https://orcid.org/0009-0009-1175-8726
e-mail: gurualexua@gmail.com
ДРОЗД АНДРІЙ
Хмельницький національний університет
https://orcid.org/0009-0008-1049-1911
e-mail: andriydrozdit@gmail.com

КІБЕРФІЗИЧНІ СИСТЕМИ ДЛЯ АВТОМАТИЗАЦІЇ ПРИМІЩЕНЬ ПІДПРИЄМСТВ

В комерційних, житлових та промислових будівлях для моніторингу та управління механічним чи електричним обладнанням широко використовується система автоматизації будівель, яка є комплексною розподіленою системою управління. Промисловий і технологічний прогрес в частині керуючих компонент стають все більш взаємопов'язаними. Потенційні переваги та, інтеграція породжують та заохочують нові атаки, що значно підвищує ризики для безпеки та захисту їх системи управління. Не всі системи автомазації будівель проєктувались так, щоб мати надійну архітектуру безпеки і покладаються переважно в цьому питанні на фізичну ізоляцію та «безпеку через невідомість». Ці методи неприйнятні для технологій «розумної будівлі». В зв'язку з цим потребує переоцінки безпека та захист поточної системи автоматизації будівель і розробка комплексного рішення, яке забезпечить цілісність, надійність і конфіденційність як на системному, так і на мережевому рівнях. Тому, метою роботи була розробка на системному рівні забезпечення надійної обчислювальної основи для пристроїв і контролерів. Використовуючи бажані функції безпеки, такі як надійна модульна конструкція, невеликий код привілеїв і формальна перевірюваність архітектури мікроядра, потребує опису посилена безпека операційних систем з вбудованим обов'язковим контролем доступу та структурою зв'язку на основі проксі-серверу для контролерів автоматизації будівель, тобто забезпечення функціонування кіберфізичних систем. Це рішення забезпечує зв'язок із дотриманням політики та ізоляцію між критично важливими та некритичними програмами в потенційно ворожому кіберсередовищі.

Розроблено спосіб обробки повідомлень та конфігурування мікроядра. Його реалізація базується на формі розподілу можливостей кінцевих точок, що відбувається в кореневому ініціальному користувацькому процесі. Подано запропоновану методику створення безпечної кіберфізичної системи для автоматизації приміщень підприємств. Вона базується на архітектурі мікроядра. Проведені дослідження з запропонованої кіберфізичною системою на основі мікроядра показали потребу удосконалення операційних систем і їх важливість в контекстсі таких завдань.

Напрямами подальших досліджень є удосконалення функцій операційних систем на основі мікроядра для їх використання в кіберфізичних системах автоматизації будівель в контексті забезпечення безпеки.

Ключові слова: кіберфізичні системи, операційні системи, автоматизація приміщень.

MELNYK VIKTORIIA, KRYVAK DENYS, VOZNYI KYRYLO, HURALNYK OLEKSANDR, DROZD ANDRIY
Khmelnytskyi National University

CYBERPHYSICAL SYSTEMS FOR AUTOMATION OF ENTERPRISE PREMISES

A building automation system, which is a complex distributed control system, is widely used in commercial, residential and industrial buildings to monitor and control mechanical or electrical equipment. Industrial and technological progress in the part of control components are becoming more and more interconnected. The potential advantages and, integration generate and encourage new attacks, which significantly increases the risks to the security and protection of their management system. Not all building automation systems are designed to have a robust security architecture and rely mostly on physical isolation and "security through obscurity" in this regard. These methods are not suitable for "smart building" technologies. In this regard, the security and protection of the current building automation system needs to be reassessed and the development of a comprehensive solution that will ensure integrity, reliability and confidentiality at both the system and network levels. Therefore, the goal of the work was the development at the system level of providing a reliable computing basis for devices and controllers. Using desirable security features such as robust modular design, small privilege code, and formal verifiability of the microkernel architecture, enhanced security of operating systems with built-in mandatory access control and a proxy-based communication framework for building automation controllers needs to be described, i.e. ensuring functioning of cyber-physical systems. This solution provides policy-compliant communication and isolation between critical and non-critical applications in a potentially hostile cyber environment.

A method of processing messages and configuring the microkernel has been developed. Its implementation is based on a form of endpoint capability allocation that occurs in the root initialized user process. The proposed method of creating a safe cyber-physical system for the automation of enterprise premises is presented. It is based on microkernel architecture. Conducted research on the proposed microkernel-based cyber-physical system showed the need to improve operating systems and their importance in the context of such tasks.

The direction of further research is to improve the functions of microkernel-based operating systems for their use in cyber-physical building automation systems in the context of security.

Keywords: cyber-physical systems, operating systems, room automation.

Вступ

В комерційних, житлових та промислових будівлях для моніторингу та управління механічним чи електричним обладнанням широко використовується система автоматизації будівель (САБ). Вона є комплексною розподіленою системою управління. У зв'язку зі зростаючим промисловим і технологічним прогресом керуючі компоненти САБ стають все більш взаємопов'язаними. Поряд з потенційними перевагами, інтеграція також породжує та заохочує нові вектори атак, що значно підвищує ризики для безпеки та захисту системи управління. Історично склалося так, що САБ не має архітектури безпеки і покладається на фізичну ізоляцію та «безпеку через невідомість». Ці методи неприйнятні для технологій «розумної будівлі». Галузь потребує [1, 2] переоцінки безпеки та захисту поточної системи автоматизації будівель і розробки комплексного рішення, яке забезпечить цілісність, надійність і конфіденційність як на системному, так і на мережевому рівнях.

Тому, метою роботи є розробка на системному рівні забезпечення надійної обчислювальної основи для пристроїв і контролерів. Використовуючи бажані функції безпеки [3], такі як надійна модульна конструкція, невеликий код привілеїв і формальну перевірюваність архітектури мікроядра, потребує опису посилена безпека операційних систем з вбудованим обов'язковим контролем доступу та структурою зв'язку на основі проксі-серверу для контролерів автоматизації будівель, тобто забезпечення функціонування кіберфізичних систем. Це рішення забезпечує зв'язок із дотриманням політики та ізоляцію між критично важливими та некритичними програмами в потенційно ворожому кіберсередовищі.

Актуальність роботи полягає в необхідності розробити метод створення безпечних кіберфізичних систем для автоматизації приміщень підприємств з використанням операційних систем на основі мікроядра.

Аналіз відомих безпекових проблемних завдань в системах автоматизації будівель

Традиційно [4, 5] САБ складаються з декількох автономних підсистем, специфічних для конкретних застосунків. Кожна підсистема надає лише одну послугу, таку як контроль температури, вентиляція тощо. Пристрої в кожній підсистемі пов'язані між собою через мережі автоматизації будівель, які раніше були відокремлені від ІТ-систем у навколишньому середовищі. Управління логікою управління здійснюється за допомогою центральних серверів управління, які називаються системами управління будівлею, які з'єднуються з програмованими логічними контролерами (ПЛК) за схемою «ведучий-підлеглий». Різні підсистеми об'єднуються в різні домени управління (наприклад, домен безпеки, домен безпеки) і управляються окремо.

Зі швидким комерційним поширенням [6, 7] кіберфізичних систем (КФС) або широко відомих технологій Інтернету речей (IoT) у промисловості, розробники та дослідники прагнули прийняти концепцію «розумних будівель», яка використовує різні датчики для кращого розуміння життєвого контексту. Це надає підходи з підтримкою IP для керування навколишнім середовищем та з'єднує існуючі системи керування, щоб вони краще реагували на індивідуальні потреби. Будівлі зазнають трансформації, щоб краще обслуговувати клієнтів і мешканців за допомогою передової автоматизації та об'єднання в мережу. Це, в свою чергу вимагає інтеграції все більш складних обчислювальних і мережевих можливостей в САБ, надійність [8] і стабільність яких має вирішальне значення для підтримки повсякденного життя і забезпечення як безпеки, так і для нормальної роботи, під час надзвичайних ситуацій. Ці досягнення полегшили життя мешканцям і допомогли забудовникам знайти нові способи, наприклад, зменшити споживання енергії. Для досягнення мети енергозбереження необхідна тісна інтеграція між датчиками, органами управління, Без інтелектуальної складової частини [9] це реалізувати неможливо. Сьогодні сучасний САБ об'єднує кілька підсистем на рівні пристроїв. Ця інтеграція означає не тільки взаємозв'язок між собою існуючих систем, але й можливість дистанційного керування та збору даних. У зв'язку з цими змінами хмарні рішення машинного навчання та системи управління, доступні в Інтернеті, починають набувати широкого поширення в САБ [9]. Хоча, такі переваги, як гнучка реакція та каскадне управління, не викликають сумнівів, як і інші сфери, які охоплюють перспективу CPS, такі як розумні мережі, розподілена робототехніка, автономні транспортні засоби та системи авіоніки, розвиток САБ як системи зараз стикається з численними перешкодами на шляху прогресу, особливо взаємозв'язком безпеки-захисту та проблемами надійності, пов'язаними із загрозою кібератак [10, 11].

Інтеграція між мережею управління та ІТ-мережею значно збільшує поверхню атаки САБ, пропонуючи зловмисникам більше можливостей для атаки, ніж будь-коли раніше. Кібератаки не тільки збільшують споживання енергії шляхом втручання в ретельно складені плани, але й можуть змінити функціональність компонентів керування, тим самим впливаючи на загальну безпеку будівель. Відсутність належної сумісності та стандартів безпечної архітектури створює потенційні небезпеки та загрози для контролю, особливо для критично важливих для безпеки об'єктів.

САБ вже підключені до інтернету. Існує більше 20 тисяч систем Tridium Niagara (одна з найпопулярніших платформ для управління будівлями), підключених до Інтернету [12]. На сьогодні в ній було виявлено атаки у понад 50 000 будівлях, які піддаються впливу Інтернету навмисно або через неправильну конфігурацію [10, 11]. Крім того, САБ широко використовує застарілі низькорівневі протоколи, які відправляють дані у вигляді відкритого тексту і не мають належних механізмів аутентифікації. В роботах

[12, 13] показано, як зловмисники можуть легко отримувати доступ до керуючих пакетів, модифікувати програмований логічний контролер довільно та використовувати ретельно розроблені низькорівневі дані, зібрані за допомогою ПЛК, для впровадження програмного забезпечення високого рівня. Багато комерційних готових САБ, таких як MetaSys і Niagara [14, 15], засновані на застарілих операційних системах Windows. Відкриття першої кіберзброї, Stuxnet [6], свідчить про те, що спеціально створене зловмисне програмне забезпечення може бути легко запущено проти мереж, таких як САБ, для їх виведення з ладу, щоб компоментувати установи, які займаються високою безпекою.

Нещодавні резонансні атаки продемонстрували можливі загрози та потенційні наслідки кібератак у середовищах будівель. Відомі випадки, коли зловмисники створили перевірку концепції зловмисного програмного забезпечення, яке продемонструвало, як зловмисники можуть використовувати системи для з'єднання мереж із зовнішнім світом [16]. Такі атаки не тільки можуть допомогти зловмисникам отримати контроль над САБ, але й можуть стати сходинкою для подальшого проникнення в інші критично важливі інфраструктури, такі як електроенергія, вода, транспорт, охорона здоров'я тощо. Завдання забезпечення автоматизації будівель має кілька вимірів. По-перше, будівлі неоднорідні. Будівлі проектуються для різних цілей, а тому мають різні вимоги. Наприклад, система автоматизації стадіону може бути зосереджена на тому, як керувати освітленням, температурою та посиленою вентиляцією, щоб підтримувати низьку концентрацію вуглекислого газу. З іншого боку, головною проблемою для об'єкта біоізоляції або хімічного заводу може бути мінімізація повітрообміну між різними зонами, щоб зменшити ризик перехресного забруднення. Архітектура САБ наступного покоління повинна враховувати деталі різних сценаріїв використання та розуміти відмінності вимог, що містяться в ньому. По-друге, САБ є ієрархічною структурою [17]. Різні підієрархії мають різні погляди на систему. Те, де і як кожна підієрархія застосовує різні правила, має значний вплив на загальну безпеку, безпеку та надійність будівлі. Життєво важливо мати глобальний погляд на систему зверху вниз з відповідною абстракцією. Формалізація різних вимог безпеки та захисту та аналіз загроз з глобальної точки зору системи, а також розробка політик та відповідних механізмів безпеки для їх підтримки повинні стати ключовим кроком у розробці САБ. Також, не менш важливою є сумісність [18] САБ. Високоінтегровані підсистеми не тільки відкрито пов'язані між собою за допомогою мереж, але й взаємодіють [19] за допомогою фізичних особливостей, якими вони керують, наприклад, відкриті двері змінюють характер повітряних потоків. При проектуванні БАС часто неможливо повністю ізолювати одну систему управління від іншої [20].

Беручи до уваги великі майбутні зміни в САБ наступного покоління та потенційні ризики [21], які вони спричинять, дуже важливо переосмислити та переоцінити те, як системи автоматизації будівель проектуються та організовуються разом. Для того, щоб забезпечити необхідні гарантії безпеки в розподіленому середовищі [22], безпека не може вважатись другорядною ідеєю, а повинна бути одним з найважливіших міркувань усього процесу проектування та реалізації. Оцінка безпеки включає в себе не тільки механізми безпечного мережевого зв'язку, аутентифікації, але також повинна включати вбудовані платформи, архітектуру системи, операційні системи тощо. Крім того, з практичних міркувань необхідно враховувати застарілі підсистеми та стандартні протоколи промислового управління в існуючих САБ.

Таким чином, напрям роботи можна узагальнити за трьома категоріями: дослідження безпеки та захищеності в системах автоматизації будівель; безпечні операційні системи для вбудованих пристроїв; безпечна віртуалізація для вбудованих систем. Проведено аналіз предметної області для дослідження. В результаті встановлено, що кіберфізичні системи, які активно використовують а автоматизації будівель, потребують розроблення засобів і методів забезпечення їх безпеки від зовнішніх атак, оскільки використовувані операційні системи не орієнтовані на виконання саме таких завдань. За основу для напряму досліджень в частині забезпечення безпеки може бути розглянуто розроблення мікроядра операційної системи.

Стратегія формування безпеки в системах автоматизації будівель

Важливим місцем будь-якої кіберфізичної системи, що є основою систем автоматизації будівель, є саме операційні системи. Тому, для прототипу в якості платформи розробки було обрано два мікроядра: MINIX 3 і seL4. Завдяки дослідженням архітектура виділяється як основна обчислювальна платформа контролерів будівель. Одним із прикладів є контролер автоматизації будівель.

MINIX – це відома ОС на основі мікроядер, яка розроблена як приклад мікроядрового підходу. Останньою версією є MINIX 3, яка націлена на вбудовані пристрої з акцентом на високу надійність. З точки зору користувача, MINIX 3 дуже схожа на традиційну систему в стилі UNIX. Насправді, більшість програм для роботи з користувачем були портовані в MINIX 3. Однак архітектура MINIX 3 повністю відрізняється від традиційних систем, які керуються типовим мікроядром. Ядро MINIX 3 складається всього з 6 000 рядків коду. Код ядра містить апаратні абстракції, переривання, блоки управління процесами, таймери та примітиви IPC. ОС побудована в трьох рівнях: рівень драйверів пристроїв, рівень системних служб; рівень застосунків користувача. Окрім драйверів пристроїв, MINIX 3 складається з кількох серверів, що працюють як ізольовані процеси в просторі користувача, включаючи менеджер процесів, віртуальну файлову систему, менеджер віртуальної пам'яті, службу системної інформації, диспетчер пристроїв тощо. MINIX 3 є найбільш усталеною ОС на основі мікроядра з відкритим вихідним кодом з найбільшою підтримкою драйверів пристроїв. Тому, використовуємо його як одну з платформ для розробки прототипу.

Мікроядро seL4 є найновішим представником сімейства мікроядер L4. Дотримуючись філософії ядра

L4, ядро seL4 підтримує абстракції для віртуальних адресних просторів, потоків і міжпроцесного зв'язку з високою продуктивністю. Найголовніше, що seL4 є першим математично перевіреним програмним ядром. Впроваджена та перевірена формальна верифікація доводить, що виконуваний машинний код, складений із більше, ніж 10 000 рядків коду seL4, є функціонально правильним щодо його високорівневої специфікації за допомогою доведення теорем, що означає, що код ядра вільний від вразливостей, таких як переповнення буфера, невизначена поведінка тощо.

На відміну від MINIX 3 і традиційної Unix-подібної системи, seL4 використовує модель безпеки, засновану на можливостях. З точки зору ядра, ресурси розглядаються як різні типи об'єктів ядра. Права власності або доступу до об'єкта ядра, наприклад, невикористовувані області пам'яті, таблиці сторінок, блоки керування завданнями, кінцеві точки IPC тощо, обліковуються за можливостями. Керування доступом на основі можливостей безпосередньо пов'язане з керуванням віртуальною пам'яттю через MMU. Можливість — це непідробний токен, який представляє явні повноваження власника та безпосередньо керується ядром. Ця модель забезпечує гнучкий механізм для обґрунтування та забезпечення дотримання політики контролю.

У MINIX 3 примітив IPC ядра — це синхронна передача повідомлень. Синхронна передача повідомлень використовує механізм у стилі рандеву. При виклику примітивів IPC процес виклику буде призупинено до тих пір, поки повідомлення не буде скопійовано від відправника до одержувача. Повідомлення є 64-байтовими буферами фіксованого розміру, які включають 4-байтовий ідентифікатор кінцевої точки, 4-байтове поле типу повідомлення та 56-байтове корисне навантаження. Кінцева точка призначення має бути явно вказана для надсилання або отримання повідомлення. Кінцева точка ідентифікує процес унікально серед операційної системи. Він складається з номера слота процесу, об'єднаного з номером генерації для адресації IPC, який зберігається в блоці керування процесом. Є 3 системних виклики: ipc_send() ipc_receive() і ipc_sendrec(). Системні виклики блокуються до тих пір, поки повідомлення не буде доставлено в процес, що приймає. Системні виклики блокують до отримання повідомлення від цільового процесу. Це забезпечує атомарна операція для надсилання та отримання зв'язку в обидві сторони. У поточній версії, синхронна передача повідомлень зарезервована для драйверів пристроїв і компонентів системного сервера з розробленими протоколами зв'язку.

Примітиви MINIX 3 IPC є ефективним засобом для впровадження обов'язкового контролю доступу для ізоляції процесів та регулювання зв'язку. Модифікуємо ядро MINIX 3 так, щоб передавати примітиви повідомлень усім процесам користувача. Оскільки ядро полегшує всі IPC, то воно є ідеальним місцем для забезпечення дотримання політики IPC. Безпосередньо надаючи примітиви IPC всім процесам користувача, також спрощуємо шляхи зв'язку та потік інформації. Крім того, додамо три системні виклики на сервері керування процесами для покращення операцій, пов'язаних з IPC: перетворює ідентифікатор процесу та повертає відповідну кінцеву точку; отримує кінцеву точку процесу за іменем; дозволяє процесу запитувати всі повідомлення, що очікують на розгляд. Ці системні виклики можуть призначити кожному процесу, серверу, унікальний номер під час періоду завантаження. Вони призначені для заміни оригінальних системних викликів для завантаження серверів процесів та системи із заданими номерами процесів. Ідентифікатори процесів призначаються випадковим чином і можуть змінюватися, тому потрібен цей номер для допомоги у побудові визначень політики IPC. Використовуємо поле, щоб унікально ідентифікувати кожен процес і застосовувати політику контролю.

Використовуючи проксі-сервери, можемо ефективно знизити потенційні ризики, викликані шкідливими внутрішніми процесами. Однак іншим набором векторів атак для пристроїв САБ є поширені мережеві атаки, такі як атаки типу "людина посередині", DDoS-атаки, атаки сніфінгу, спуфінг-атаки тощо. Наприклад, якщо зовнішній шкідливий пристрій отримує доступ до мережевого середовища, на цей пристрій не поширюється жодна політика мікроядра, що становить загрозу. Крім того, САБ часто включають кілька протоколів промислового управління одночасно, які часто залежать від постачальника. Складність і залежність САБ унеможливлюють перехід на захищені протоколи на практиці. Більш практичним рішенням було б забезпечити VPN-подібний захищений мережевий тунель між пристроями на САБ за допомогою механізмів аутентифікації та шифрування на прикладному рівні. Це добре вивчена проблема в комп'ютерних мережах. Такі протоколи широко використовуються в Інтернеті та корпоративних мережах. Хоча використання зв'язку на основі проксі-сервера є ідеальним середовищем для створення безпечних мережевих тунелів на основі програми між застосунками. У запропонованому методі використовуватимемо попередньо розподілену пару публічних/закритих ключів у проксі. Симетричний сесійний ключ узгоджується між проксі-парами за допомогою їх асиметричної пари ключів за допомогою стандартного алгоритму обміну ключами. Це рішення може бути розширене за допомогою спеціальної криптографічної апаратної підтримки, наприклад, за допомогою модуля довірчої платформи для забезпечення міцнішого ланцюга довіри. Наприклад, при підтримці його авторизована пара кореневих ключів із сертифікатом може зберігати обладнання. Драйвер модуля можна використовувати для надання послуг для проксі-процесів для виведення ключів, запечатування даних та інших криптографічних функцій. У цьому випадку асиметричні пари ключів для проксі-серверів можуть бути отримані під час виконання, і проксі-серверам ніколи не потрібно буде зберігати ключі в енергонезалежній пам'яті.

Додавання всіх цих легких проксі-процесів дозволяє системі зіставляти мережевий зв'язок у локальний IPC, а також дозволяє мікроядру регулювати міжпристрої IPC шляхом арбітражу локальних IPC між локальним процесом і проксі-сервером за допомогою політик МКД. Глобальна безпека досягається

завдяки єдиним локальним перевіркам безпеки, які застосовуються на найнижчому рівні в кожному пристрої. Коли процес надсилає повідомлення проксі-серверу, зв'язок відбувається за такою логікою. Кроки методу:

1) управління САБ обробляє видачу повідомлень для ініціювання зв'язку;
2) запит потрапляє в ядро, а ядро перевіряє валідність цього зв'язку з МКД;
3) ядро пересилає повідомлення проксі-серверу, який очікує на повідомлення;
4) проксі-сервер отримує повідомлення через надіслане йому повідомлення-підтвердження;
5) проксі-сервер перевіряє валідність вмісту повідомлень для конкретної програми;
6) проксі-сервер шифрує і пересилає вихідне повідомлення на мережевий сервер;
7) проксі-сервер відповідає на процес керування САБ, вказуючи на те, що повідомлення було надіслано.

Для передачі великого обсягу даних може використовуватися загальна пам'ять. MINIX 3 підтримує операції спільної пам'яті. Але реалізація реалізується за допомогою IPC-сервера процесу. Щоб налаштувати спільну пам'ять, процеси обмінюються даними з IPC-сервером за допомогою передавання повідомлень. Внесемо зміни до сервера IPC, щоб він відповідав вимогам політики. Процеси, які ініціалізують спільну пам'ять, явно вказують кінцеву точку процесу, яка може отримати доступ до неї через сегмент пам'яті. Успішне створення сегмента спільної пам'яті повертає випадковий секретний ключ. Процес може передати секретний ключ цільовому вузлу за допомогою стандартного IPC, що може статися лише в тому випадку, якщо їм дозволено обмінюватися даними. Тільки явно вказаний процес з випадковим секретним ключем може приєднати спільну пам'ять. Спосіб, у який спільна пам'ять також опосередковується. На рис. 1 зображено пропоновану стратегію з використанням мікроядра.

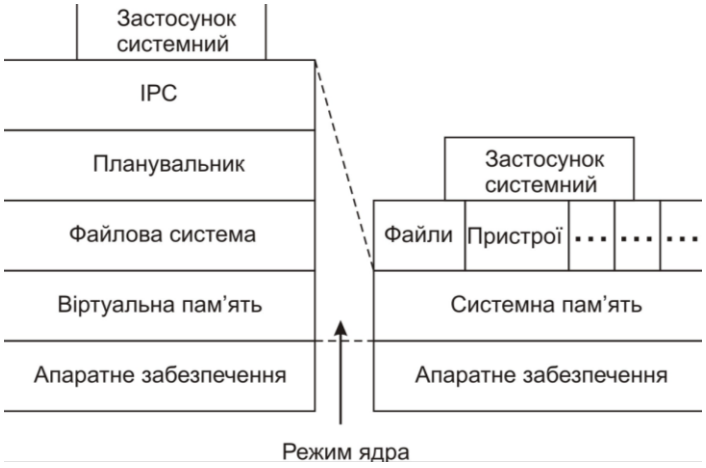

Рис.1 – Схема режимів ядра ОС

Прототип з використанням мікроядра seL4 має аналогічну реалізацію і досягає тієї ж мети. Завдяки моделі контролю доступу, заснованій на можливостях, мікроядро seL4 вже має надійний контроль доступу. Зміни в просторі ядра не потрібні. Політика реалізується у формі розподілу можливостей кінцевих точок, що відбувається в кореневому ініціальному користувацькому процесі. Мережевий зв'язок на основі проксі-сервера працює так само. Це свідчить про те, що обрана архітектура добре працює з архітектурою мікроядра в цілому.

Таким чином, розроблено спосіб обробки повідомлень та конфігурування мікроядра. Він реалізується у формі розподілу можливостей кінцевих точок, що відбувається в кореневому користувацькому процесі кібербізичної системи автоматизації будівель.

Дослідження результатів експериментів

Однією з головних проблем при прийнятті архітектури мікроядра була її нижча продуктивність у порівнянні з монолітними архітектурами. Однак це проблемне питання було мінімазовано сучасним апаратним забезпеченням і вдосконаленням архітектури ОС. Сучасні мікроядра, такі як seL4, досягають високої продуктивності за рахунок безлічі методів оптимізації коду. Отже, ці специфічні недоліки можуть не застосовуватися однаково до систем мікроядра в домені САБ. Зокрема, було продемонстровано, що характерний еталон систем мікроядер, затримка IPC, становить менше 80 наносекунд для деяких реалізацій. Загалом, продуктивність IPC є найважливішим показником продуктивності мікроядра, оскільки всі взаємодії компонентів відбуваються через IPC. Оскільки мікроядра сильно налаштовані на продуктивність IPC, цей аналіз висунув гіпотезу, що додаткові витрати на проксі-процеси будуть в межах допустимих для контролерів автоматизації будівель.

На основі спостережень за трафіком САБ з локальних мереж САБ-пристрої обмінюються даними в мережі за допомогою протоколів. На прикладному рівні блок даних протокол інкапсульований у частині даних. Максимальна довжина може досягати великого значення, однак для зворотної сумісності розмір пакета зазвичай обмежений, що вказується в кожному полі.

Таким чином, ця оцінка оцінює лише продуктивність затримки мережі у кожній ситуації. Крім того,

трафік САБ відносно розріджений з близько 35 пакетами/секунду для магістралі і включає близько 100 пристроїв. Через ці спостереження з низькою пропускною здатністю, оцінка не вимірювала пропускну здатність системно, хоча могла перевищити 20 пакетів за секунду.

Висновки

Розроблено спосіб обробки повідомлень та конфігурування мікроядра. Його реалізація базується на формі розподілу можливостей кінцевих точок, що відбувається в кореневому ініціальному користувацькому процесі. Подано запропоновану методику створення безпечної кіберфізичної системи для автоматизації приміщень підприємств. Вона базується на архітектурі мікроядра. Проведені дослідження з запропонованої кіберфізичною системою на основі мікроядра показали потребу удосконалення операційних систем і їх важливість в контектсі таких завдань.

Напрямами подальших досліджень є удосконалення функцій операційних систем на основі мікроядра для їх використання в кіберфізичних системах автоматизації будівель в контексті забезпечення безпеки.

Література

1. Elphinstone K., Heiser G. From L3 to seL4 what have we learnt in 20 years of L4 microkernels? Proceedings of the Twenty-Fourth ACM Symposium on Operating Systems Principles - *SOSP 13*, 2013.

2. Biggs S., Lee D., Heiser G. The jury is in: Monolithic os design is flawed. *Proceedings of the 9th Asia-Pacific Workshop on Systems*, 2018.

3. Denysiuk D., Savenko O., Lysenko S., Savenko B., Kashtalian A. Method for Detecting Steganographic Changes in Images Using Machine Learning. In: *2023 13th International Conference on Dependable Systems, Services and Technologies (DESSERT)*, Athens, Greece. 2023. P. 1-6. doi:10.1109/DESSERT61349.2023.10416453.

4. Markowsky G., Savenko O., Lysenko S., Nicheporuk A. The technique for metamorphic viruses' detection based on its obfuscation features analysis. *CEUR-WS* 2104. 2018. P. 680-687.

5. Lysenko S., Savenko Ol., Bobrovnikova K. DDoS Botnet Detection Technique Based on the Use of the Semi-Supervised Fuzzy c-Means Clustering. *CEUR-WS* 2104. 2018. P. 688-695.

6. Feng M., Xiao B., Yu B., Qian J., Zhang X., Chen P., Li, B. A Novel Deception Defense-Based Honeypot System for Power Grid Network. *International Conference on Smart Computing and Communication,* 2021, Vol. 13202, pp. 297-307. Cham: Springer International Publishing. DOI: 10.1007/978-3-030-97774-0_27

7. Savenko B., Lysenko S., Bobrovnikova K., Savenko O., Markowsky G.. Detection DNS Tunneling Botnets // *Proceedings of the 2021 IEEE 11th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), IDAACS'2021*, Cracow, Poland, September 22-25, 2021.

8. Walter E., Ferguson-Walter K., Ridley A. Incorporating deception into cyberbattlesim for autonomous defense. 2021. *arXiv preprint arXiv:2108.13980*. DOI: 10.48550/arXiv.2108.13980

9. Lysenko S., Bobrovnikova K., Shchuka R., Savenko O. A Cyberattacks Detection Technique Based on Evolutionary Algorithms. *11th International Conference on Dependable Systems. Services and Technologies (DESSERT).* 2020. V. 1. P. 127-132. DOI: 10.1109/DESSERT50317.2020.9125016

10. Almeshekah M. H., Spafford E. H. Cyber Security Deception. In: Jajodia. S., Subrahmanian. V., Swarup. V., Wang. C. (eds) *Cyber Deception*, 2016, p. 318, Cham. Springer. DOI: 10.1007/978-3-319-32699-3_2

11. Dahbul R. N., Lim C., Purnama J. Enhancing honeypot deception capability through network service fingerprint. *Journal of Physics: Conference Series,* 2017, vol. 801, article no. 012057. DOI: 10.1088/1742-6596/801/1/012057

12. Savenko B., Kashtalian A. Method for Determining the Efficiency of a Distributed Anomaly Detection System. *CSIT.* 2022. V. 2. P. 14-22. https://doi.org/10.31891/csit-2022-2-2.

13. Wegerer M., Tjoa S. Defeating the Database Adversary Using Deception – A MySQL Database Honeypot. *International Conference on Software Security and Assurance (ICSSA),* Saint Pölten. Austria, 2016. pp. 6-10. DOI: 10.1109/ICSSA.2016.8

14. Lysenko S., Savenko O., Bobrovnikova K., Kryshchuk A. Self-adaptive system for the corporate area network resilience in the presence of botnet cyberattacks. *Communications in Computer and Information Science.* 2018. V. 860. P. 385-401. DOI: 10.1007/978-3-319-92459-5_31.

15. Kedrowitsch A., Danfeng Y., Gang W., Cameron K. A First Look: Using Linux Containers for Deceptive Honeypots. *Proceedings of the 2017 Workshop on Automated Decision Making for Active Cyber Defense (SafeConfig '17)*. Association for Computing Machinery, New York, NY, USA, 2017, pp. 15–22. DOI: 10.1145/3140368.3140371

16. Razali M. F., Razali M. N., Mansor F. Z., Muruti G., Jamil N. IoT Honeypot: A Review from Researcher's Perspective. *IEEE Conference on Application. Information and Network Security (AINS).* Langkawi. Malaysia, 2018. pp. 93-98. DOI: 10.1109/AINS.2018.8631494

17. Pomorova O., Savenko O., Lysenko S., Kryshchuk A., Bobrovnikova K. A Technique for the Botnet Detection Based on DNS-Traffic Analysis. In: Gaj, P., Kwiecień, A., Stera, P. (eds) *Computer Networks. CN 2015. Communications in Computer and Information Science.* 2015. V. 522. P. 127-138. DOI: 10.1007/978-3-319-19419-6_12.

18. Bobrovnikova K., Lysenko S., Savenko B., Gaj P., Savenko O. Technique for IoT malware detection based on control flow graph analysis. *Radioelectronic and Computer Systems.* 2022. V. 1. P. 141–153. DOI:

10.32620/reks.2022.1.11.

19. Lysenko S., Savenko O., Bobrovnikova K., Kryshchuk A., Savenko B. Information technology for botnets detection based on their behaviour in the corporate area network. *Communications in Computer and Information Science.* 2017. V. 718. P. 166–181. DOI: 10.1007/978-3-319-59767-6_14.

20. Rowe N. C. Honeypot Deception Tactics. In: Al-Shaer, E., Wei, J., Hamlen, K., Wang, C. (eds) *Autonomous Cyber Deception*. Springer. Cham, 2019. DOI: 10.1007/978-3-030-02110-8_3

21. Lysenko S., Savenko O., Bobrovnikova, K. DDoS Botnet Detection Technique Based on the Use of the Semi-Supervised Fuzzy c-Means Clustering. *CEUR-WS.* 2018. V. 2104. P. 688-695.

22. Мельник В.М., Сорочинський О.Ю., Глухенький О.А., Семенюк Б.В. Метод створення безпечної кіберфізичної системи для автоматизації приміщень підприємств з використанням операційних систем на основі мікроядра / Збірник наукових праць за матеріалами XV Всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук АПКН-2023». Хмельницький, 2023, С.190-192. https://kn.khmnu.edu.ua/wp-content/uploads/sites/18/apkn-2023-corpuspaper.pdf

References

1. Elphinstone K., Heiser G. From L3 to seL4 what have we learnt in 20 years of L4 microkernels? Proceedings of the Twenty-Fourth ACM Symposium on Operating Systems Principles - *SOSP 13*, 2013.

2. Biggs S., Lee D., Heiser G. The jury is in: Monolithic os design is flawed. *Proceedings of the 9th Asia-Pacific Workshop on Systems*, 2018.

3. Denysiuk D., Savenko O., Lysenko S., Savenko B., Kashtalian A. Method for Detecting Steganographic Changes in Images Using Machine Learning. In: *2023 13th International Conference on Dependable Systems, Services and Technologies (DESSERT)*, Athens, Greece. 2023. P. 1-6.

doi:10.1109/DESSERT61349.2023.10416453.

4. Markowsky G., Savenko O., Lysenko S., Nicheporuk A. The technique for metamorphic viruses' detection based on its obfuscation features analysis. *CEUR-WS* 2104. 2018. P. 680-687.

5. Lysenko S., Savenko Ol., Bobrovnikova K. DDoS Botnet Detection Technique Based on the Use of the Semi-Supervised Fuzzy c-Means Clustering. *CEUR-WS* 2104. 2018. P. 688-695.

6. Feng M., Xiao B., Yu B., Qian J., Zhang X., Chen P., Li, B. A Novel Deception Defense-Based Honeypot System for Power Grid Network. *International Conference on Smart Computing and Communication,* 2021, Vol. 13202, pp. 297-307. Cham: Springer International Publishing. DOI: 10.1007/978-3-030-97774-0_27

7. Savenko B., Lysenko S., Bobrovnikova K., Savenko O., Markowsky G.. Detection DNS Tunneling Botnets // *Proceedings of the 2021 IEEE 11th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), IDAACS'2021*, Cracow, Poland, September 22-25, 2021.

8. Walter E., Ferguson-Walter K., Ridley A. Incorporating deception into cyberbattlesim for autonomous defense. 2021. *arXiv preprint arXiv:2108.13980*. DOI: 10.48550/arXiv.2108.13980

9. Lysenko S., Bobrovnikova K., Shchuka R., Savenko O. A Cyberattacks Detection Technique Based on Evolutionary Algorithms. *11th International Conference on Dependable Systems. Services and Technologies (DESSERT).* 2020. V. 1. P. 127-132. DOI: 10.1109/DESSERT50317.2020.9125016

10. Almeshekah M. H., Spafford E. H. Cyber Security Deception. In: Jajodia. S., Subrahmanian. V., Swarup. V., Wang. C. (eds) *Cyber Deception*, 2016, p. 318, Cham. Springer. DOI: 10.1007/978-3-319-32699-3_2

11. Dahbul R. N., Lim C., Purnama J. Enhancing honeypot deception capability through network service fingerprint. *Journal of Physics: Conference Series,* 2017, vol. 801, article no. 012057. DOI: 10.1088/1742-6596/801/1/012057

12. Savenko B., Kashtalian A. Method for Determining the Efficiency of a Distributed Anomaly Detection System. *CSIT.* 2022. V. 2. P. 14-22. https://doi.org/10.31891/csit-2022-2-2.

13. Wegerer M., Tjoa S. Defeating the Database Adversary Using Deception – A MySQL Database Honeypot. *International Conference on Software Security and Assurance (ICSSA),* Saint Pölten. Austria, 2016. pp. 6-10. DOI: 10.1109/ICSSA.2016.8

14. Lysenko S., Savenko O., Bobrovnikova K., Kryshchuk A. Self-adaptive system for the corporate area network resilience in the presence of botnet cyberattacks. *Communications in Computer and Information Science.* 2018. V. 860. P. 385-401. DOI: 10.1007/978-3-319-92459-5_31.

15. Kedrowitsch A., Danfeng Y., Gang W., Cameron K. A First Look: Using Linux Containers for Deceptive Honeypots. *Proceedings of the 2017 Workshop on Automated Decision Making for Active Cyber Defense (SafeConfig '17)*. Association for Computing Machinery, New York, NY, USA, 2017, pp. 15–22. DOI: 10.1145/3140368.3140371

16. Razali M. F., Razali M. N., Mansor F. Z., Muruti G., Jamil N. IoT Honeypot: A Review from Researcher's Perspective. *IEEE Conference on Application. Information and Network Security (AINS).* Langkawi. Malaysia, 2018. pp. 93-98. DOI: 10.1109/AINS.2018.8631494

17. Pomorova O., Savenko O., Lysenko S., Kryshchuk A., Bobrovnikova K. A Technique for the Botnet Detection Based on DNS-Traffic Analysis. In: Gaj, P., Kwiecień, A., Stera, P. (eds) *Computer Networks. CN 2015. Communications in Computer and Information Science.* 2015. V. 522. P. 127-138. DOI: 10.1007/978-3-319-19419-6_12.

18. Bobrovnikova K., Lysenko S., Savenko B., Gaj P., Savenko O. Technique for IoT malware detection based on control flow graph analysis. *Radioelectronic and Computer Systems.* 2022. V. 1. P. 141–153. DOI: 10.32620/reks.2022.1.11.

19. Lysenko S., Savenko O., Bobrovnikova K., Kryshchuk A., Savenko B. Information technology for botnets detection based on their behaviour in the corporate area network. *Communications in Computer and Information Science.* 2017. V. 718. P. 166–181. DOI: 10.1007/978-3-319-59767-6_14.

20. Rowe N. C. Honeypot Deception Tactics. In: Al-Shaer, E., Wei, J., Hamlen, K., Wang, C. (eds) *Autonomous Cyber Deception*. Springer. Cham, 2019. DOI: 10.1007/978-3-030-02110-8_3

21. Lysenko S., Savenko O., Bobrovnikova, K. DDoS Botnet Detection Technique Based on the Use of the Semi-Supervised Fuzzy c-Means Clustering. *CEUR-WS.* 2018. V. 2104. P. 688-695.

22. Melnyk V.M., Sorochynskyi O.Iu., Hlukhenkyi O.A., Semeniuk B.V. Metod stvorennia bezpechnoi kiberfizychnoi systemy dlia avtomatyzatsii prymishchen pidpryiemstv z vykorystanniam operatsiinykh system na osnovi mikroiadra / Zbirnyk naukovykh prats za materialamy XV Vseukrainskoi naukovo-praktychnoi konferentsii «Aktualni problemy kompiuternykh nauk APKN-2023». Khmelnytskyi, 2023, S.190-192. https://kn.khmnu.edu.ua/wp-content/uploads/sites/18/apkn-2023-corpuspaper.pdf