УДК 004.75:004.52
DOI https://doi.org/10.32782/IT/2023-3-3

Антоніна КАШТАЛЬЯН
к.т.н., докторанка, доцент, Хмельницький національний університет, вул. Інститутська, 11, м. Хмельницький, Україна, 29016, yantonina@ukr.net
***ORCID:** 0000-0002-4925-9713*
***Scopus Author ID:** 57218242499*

Бібліографічний опис статті: Каштальян, А. (2023). Концептуальна модель архітектури мультикомп'ютерних систем із приманками та пастками для виявлення та протидії зловмисному програмному забезпеченню та комп'ютерним атакам. *Information Technology: Computer Science, Software Engineering and Cyber Security,* 3, 22–31, doi: https://doi.org/10.32782/IT/2023-3-3

КОНЦЕПТУАЛЬНА МОДЕЛЬ АРХІТЕКТУРИ МУЛЬТИКОМП'ЮТЕРНИХ СИСТЕМ ІЗ ПРИМАНКАМИ ТА ПАСТКАМИ ДЛЯ ВИЯВЛЕННЯ ТА ПРОТИДІЇ ЗЛОВМИСНОМУ ПРОГРАМНОМУ ЗАБЕЗПЕЧЕННЮ ТА КОМП'ЮТЕРНИМ АТАКАМ

В роботі здійснено аналіз такого класу систем виявлення та протидії зловмисному програмному забезпеченню та комп'ютерним атакам як обманні системи. В таких системах закладаються функціонали з приманок і пасток. І такі системи використовують додатково та сумісно з рештою систем іншого спрямування для виявлення та протидії зловмисному програмному забезпеченню та комп'ютерним атакам. При експлуатації корпоративних мереж використовуються різноманітні системи виявлення та протидії зловмисному програмному забезпеченню та комп'ютерним атакам. Основним завданням для адміністраторів корпоративних мереж є те, щоб застосовувані ними засоби та їх особливості не були відомі зловмисникам.

В роботі запропоновано концептуальну модель архітектури мультикомп'ютерних систем з приманками та пастками для виявлення та протидії зловмисному програмному забезпеченню та комп'ютерним атакам. Особливістю запропонованої моделі є те, що в ній синтезовано характерні властивості такого класу систем та особливу характеристичну властивість. Цією характеристичною властивістю є контролер системи за прийнятими в ній рішеннями. Це необхідно для того, щоб системи такого класу були невідомими для зловмисників. Це дасть змогу забезпечити ефективну протидію зловмисникам, які здійснюють спроби проникнення в корпоративні мережі, використовуючи різноманітні способи та засоби.

В роботі запропонована методика розрахунку ефективності мультикомп'ютерних систем такого класу. Також, було здійснено постановку експерименту для розробленої системи згідно запропонованої концептуальної моделі. Результати проведеного експерименту підтверджують перспективність досліджень в напрямі використання контролера в мультикомп'ютерних системах приманок та пасток для виявлення та протидії ЗПЗ та КА.

Напрямом подальших досліджень буде деталізація запропонованої концептуальної моделі архітектури мультикомп'ютерних систем до рівня типових елементів та компонентів і, відповідно, доповнення її відображенням зв'язків між ними.

***Ключові слова:** обманні системи; мультикомп'ютерні системи; контролер; зловмисне програмне забезпечення; комп'ютерні атаки.*

Antonina KASHTALIAN
PhD, Associate Professor of the Department of Physics and Electrical Engineering, Doctoral Staff, Khmelnytskyi National University, 11, Instytutska str., Khmelnytskyi, Ukraine, 29016, yantonina@ukr.net
***ORCID:** 0000-0002-4925-9713*
***Scopus Author ID:** 57218242499*

To cite this article: Kashtalian, A. (2023). Kontseptualna model arkhitektury multykompiuternykh system iz prymankamy ta pastkamy dlia vyiavlennia ta protydii zlovmysnomu prohramnomu zabezpechenniu ta kompiuternym atakam [A conceptual model of the architecture of multi-computer systems with decoys and traps for detecting and countering malware and computer attacks]. *Information Technology: Computer Science, Software Engineering and Cyber Security*, 3, 22–31, doi: https://doi.org/10.32782/IT/2023-3-3

A CONCEPTUAL MODEL OF THE ARCHITECTURE OF MULTI-COMPUTER SYSTEMS WITH DECOYS AND TRAPS FOR DETECTING AND COUNTERING MALWARE AND COMPUTER ATTACKS

The work analyzes such a class of systems for detecting and countering malicious software and computer attacks as deception systems. In such systems, functional systems of baits and traps are laid. And such systems are used in addition and compatible with the rest of the systems of the other direction to detect and counter malicious software and computer attacks. When operating corporate networks, various systems for detecting and countering malicious software and computer attacks are used. The main task for administrators of corporate networks is to ensure that the tools they use and their features are not known to attackers.

The paper proposes a conceptual model of the architecture of multicomputer systems with decoys and traps for detecting and countering malicious software and computer attacks. The peculiarity of the proposed model is that it synthesizes the characteristic properties of this class of systems and a special characteristic property. This characteristic property is the controller of the system according to the decisions made in it. This is necessary so that systems of this class are unknown to attackers. This will make it possible to provide effective countermeasures against attackers who attempt to penetrate corporate networks using various methods and means.

The paper proposes a method for calculating the efficiency of multicomputer systems of this class. Also, an experiment was set up for the developed system according to the proposed conceptual model. The results of the conducted experiment confirm the perspective of research in the direction of using the controller in multi-computer systems of baits and traps for the detection and countermeasures of malware and computer attacks.

The direction of further research will be detailing the proposed conceptual model of the architecture of multicomputer systems to the level of typical elements and components and, accordingly, supplementing it with a display of the connections between them.

Key words: *deceptive systems; multicomputer systems; controller; malicious software; computer attacks.*

Вступ. При експлуатації корпоративних мереж використовуються різноманітні системи виявлення та протидії зловмисному програмному забезпеченню (ЗПЗ) та комп'ютерним атакам (КА). Архітектура таких систем та їх особливості повинні бути невідомими для зловмисників. Це дасть змогу забезпечити ефективну протидію зловмисникам, які здійснюють спроби проникнення в корпоративні мережі, використовуючи різноманітні способи та засоби. Перспективним напрямом дослідження стає розробка нової архітектури систем, які б використовувались на різних етапах виявлення та протидії ЗПЗ та КА. Ці системи могли б стати основою синтезу систем з приманками та пастками для виявлення та протидії ЗПЗ і КА, що демонструють свою ефективність і, відповідно, зацікавленість в користувачів. Це підтверджується, також тим, що спостерігається стрімкий розвиток таких систем на ринку спеціалізованого антивірусного програмного забезпечення для використання в корпоративних мережах. Розміщення таких систем в корпоративних мережах повинно залучати більшість з комп'ютерних станцій, які активно використовуються та функціонують в комп'ютерній мережі. Тому, поєднання комп'ютерних станцій з використанням проміжного програмного забезпечення дасть змогу створити спеціалізовану мультикомп'ютерну систему. Важливою особливістю в архітектурі таких систем є реалізація спроможності, побудованої згідно неї системи, приймати рішення, відповідно до поточного стану в цілому, зокрема, для уникнення вивчення поведінки системи зловмисниками при повторенні їх дій спроможності системи при однакових початкових станах вибирати різні наступні кроки. Така вимога потребує синтезу в архітектурі системи окремого контролера за результатами прийнятих рішень відповідною підсистемою.

Для вирішення проблеми розробки архітектури таких систем здійснимо розроблення концептуальної моделі архітектури мультикомп'ютерних систем з приманками та пастками для виявлення та протидії ЗПЗ і КА. Такі системи відносяться до класу систем обману, в яких наявні приманки та пастки.

Аналіз останніх досліджень і публікацій. Системи, побудовані на основі обманних технологій, містять різного типу приманки та пастки, які імітують роботу реальних систем (Zobal L.D., 2019, p. 1-9). Обманні системи розглядаються як підробні системи, інтегровані в комп'ютерні системи (Almeshekah M.H., 2016), які використовують для спотворення стану мережі для введення в оману зловмисників (Fraunhol D., 2018), відслідковування та взаємодії з ними, збору даних про атаки, дії та характеристики зловмисників (Zielinski D., 2022). При розробці обманних систем до них висуваються вимоги одночасно бути привабливими для зловмисників та мінімізувати супутні витрати на їх функціонування. В зв'язку з цим суттєва увага приділяється розробці конфігурації такої системи та розташуванню приманок. В роботі (Acosta J.C., 2021, p. 1-18) запропонований підхід для вибір-

кового встановлення приманки, який дозволяє динамічно використовувати ресурси відповідно до дій зловмисника. Приманки такої системи складаються з простору імен ядра та віртуальних машин, що активуються. Ряд робіт присвячено локалізації приманок в мережі. В роботі (Anwar A.H., 2022, p. 3438-3452) розглянуто двофазний метод обману, в першій фазі якого розробляється проактивна політика локалізації приманки, в другій фазі реалізується реактивний підхід, що динамічно визначає розташування приманок. Наявність приманок та самої deception системи має бути прихована від зловмисників, з метою чого використовують методи запобігання виявленню, що ґрунтуються на досліджених стратегіях виявлення приманок (Tsikerdekis M., 2018, p. 1-6) та динамічних приманках (Mphago B., 2017, p. 179-185).

Методи згідно обманних технологій комбінують застосування штучного інтелекту, теорії ігор та навчання з підкріпленням. Стратегія розташування приманок, також, враховує вподобання зловмисників (Sayed M.A., 2023), модель динамічної гри для двох гравців явно враховує розвиток станів в результаті змін у підключенні до мережі. Теоретико-ігровий підхід, також, пропонується використовувати для захисту найбільш цінних ресурсів мережі (Anwar A.H., 2022, p. 543-549), оцінки впливу розміру мережі на рішення зловмисників щодо атак (Katakwar H., 2020) та захисту ресурсів мережі від DoS атак (Çeker H., 2016), що дозволяє моделювати взаємодію між засобами захисту та зловмисниками згідно моделі гри та визначати оптимальну конфігурацію для запобігання атакам. В роботі (Huang L., 2021, p. 4843-4856) запропоновано теоретико-ігровий метод для проєктування обманних механізмів, що складаються з генератора, стимулюючого модулятора та модулятора довіри, які дозволяють спонукати зловмисників до бажаних дій. На основі гіпергри запропоновано метод розгортання системи приманок, що збалансовано використовує приманки високого на низького рівня взаємодії (Anwar A.H., 2023, p. 3393-3398).

Обманні системи широко використовуються для захисту різного типу систем, в тому числі IoT (Razali M.F., 2018), промислової автоматизації, цифрових двійників (Priya V.S.D., 2023), вбудованих пристроїв та мереж, під'єднаних до мережі Інтернет (Sikos L.F., 2023, Feng M., 2022). Мережі автоматизованих систем управління потребують використання спеціальних протоколів та специфічних методів захисту від кіберзагроз. Для цього адаптується обманна технологія приманок (Abe S., 2018, p. 372-379). Використання обманних систем приманок є важливим у сфері баз даних для захисту критичних даних організацій (Wegerer, 2016, p. 6-10).

Сучасні системи, побудовані згідно обманних технологій, мають ряд переваг порівняно з іншими. Зокрема, це масштабованість, гнучкість, керованість та інтегрованість систем. Такі системи здатні автоматизувати, конфігурувати та розгортати приманки та пастки, використовувати штучний інтелект для оптимізації обманного середовища (Acalvio ShadowPlex), підтримувати значну кількість обманних об'єктів, є гнучкими до налаштувань та інтеграції з існуючими системами захисту та менеджменту (CounterCraft Cyber Deception Platform). Системи можуть бути розгорнуті локально, в хмарному середовищі, в центрах даних, гібридних середовищах, тощо (Attivo ThreatDefend Deception and Response Platform), визначають будь-які зміни в середовищі та активують обманні можливості для захисту від атак поштових сервісів, мобільного зв'язку, соціальних мереж та стаціонарних робочих станцій (Proofpoint Identity Threat Defense). Такі системи мають високу точність виявлення зловмисних дій із відсутністю хибно позитивних спрацювань (Fidelis Deception platform).

Таким чином, проведений аналіз останніх досліджень та реальних розробок обманних систем підтверджує перспективність такого напряму для вирішення проблеми виявлення та протидії ЗПЗ та КА. Зростання кількості досліджень в цьому суттєво ускладнюватиме зловмисникам їх спроби у здійснення атак на комп'ютерні системи користувачів через різноманітність таких засобів.

Методологія дослідження. Користувачам комп'ютерних мереж необхідні системи для виявлення ЗПЗ та КА, які дадуть змогу, крім забезпечення безпеки на різних етапах можливого проникнення в комп'ютерні системи чи станції, що об'єднані в мережу, для етапу, коли на всіх попередніх етапах такі виявлення не були здійснені, але могли мати місце проникнення в систему. Тобто, розглядатимемо ті системи, які використовуватимуться для виявлення ЗПЗ та КА, які змогли пройти всі етапи захисту комп'ютерних станцій та мережі, і їх відсоток може бути невеликим. Крім того, такі системи повинні залучатись і до попередження та виявлення ЗПЗ та КА на всіх етапах їх спроб проникнення та інфікування об'єктів КС, як додаткові системи. Розробники різних засобів попередження, виявлення та протидії ЗПЗ та КА заявляють про великий відсоток правильного достовірного виявлення, але неве-

ликий відсоток для невиявлених ЗПЗ та КА при стрімкому щорічному зростанні кількості таких засобів може бути великим кількісно. Це може призвести до їх проникнення в КС, що створить проблеми користувачам і, дійсно, може бути представлений певною кількістю таких засобів, для виявлення яких необхідні системи виявлення та протидії, що матимуть нестандартні конфігурування і спроможності до автоматичної зміни своєї архітектури та прийнятих рішень без втручання користувача. Потреба в таких системах зростає і через необхідність та бажання здійснення зі сторони власників комп'ютерних мереж, які, наприклад, експлуатуються на підприємствах, прихованого спостереження за процесами в мережі та виявлення тих з них, які можуть створити загрози даним, що зберігаються у відповідних ресурсах в мережі.

Серед різних за призначенням систем виявлення ЗПЗ та КА є системи, які крім виявлення загроз створюють в комп'ютерних мережах хибні об'єкти для атак, що надає змогу адміністраторам таких мереж можливість відслідковувати процеси в мережах, які є зловмисними чи аномальними і потребують зупинки. Тому, перспективними для розробки є системи, які орієнтовані на виявлення ЗПЗ та КА, що пройшли певні етапи захисту, на яких використовувались традиційні засоби і системи попередження, виявлення та протидії, призначення яких та можливі варіанти конфігурування при використанні відомі зловмисникам. Серед таких систем особливе місце в класифікації займають системи попередження, виявлення та протидії із певною множиною приманок та пасток для ЗПЗ та КА. Їх використання створює хибні об'єкти атаки для зловмисника та дозволяє зберегти відомості про такі атаки та розповсюдження ЗПЗ в комп'ютерних станціях в мережі.

Для покращення ефективності систем виявлення та протидії ЗПЗ та КА за рахунок використання приманок та пасток, необхідним є інтегрування цих засобів в складні системи із залученням всіх комп'ютерних станцій в мережі та організації функціонування їх таким чином, щоб вони могли реагувати на зловмисні та аномальні процеси сумісно та без втручання користувача. Таким чином, необхідним є побудова не однієї приманки та пастки в певній комп'ютерній станції, а мережі приманок та пасток для здійснення комплексного захисту комп'ютерної мережі на етапі, коли КА змогли пройти через міжмережне екранування, а ЗПЗ змогло подолати перевірку антивірусними засобами і системами. Така система з приманками та пастками включатиме приманки, які здійснюють моніторинг зловмисного трафіку, тому вона може забезпечити максимально швидке його виявлення, а також виявлення патернів нових атак. Пастки при поєднанні їх в мережі можуть імітувати тіньову комп'ютерну мережу. Така система з приманок і пасток може бути комбінованою з них системою і для досягнення ефективного результату повинна включати тіньові приманки та пастки, які дозволять встановити та відслідкувати поведінку зловмисника при атаці, а також виявити ЗПЗ та КА з більшою вірогідністю. Важливим завданням, яке має бути вирішене при використанні таких систем полягає не тільки у застосуванні приманок та пасток, але й в управлінні їх використанням. Ефективність таких засобів суттєво залежить від організаційної складової частини системи. Використовуючи такі засоби в реальних системах, покращення ефективності може бути досягнуто за рахунок заміни оператора чи користувача на відповідну підсистему, яка зможе забезпечити ефективну організацію.

Антивірусні приманки та пастки як окремі частини системи можуть окремо приманками чи пастками, але можуть бути скомбіновані разом як окремі частини системи. Взаємодія їх функціоналів між собою може бути здійснена за потреби. Також, їх інтелектуалізація може стосуватись окремо приманок і окремо пасток, коли вони поєднані, але може і бути віднесеною до обох з них одночасно.

Використання лише програмної системи одночасно в якості і приманок та системи, в якій будуть прийматись рішення щодо наступного опрацювання отриманих подій в мережі приманок та пасток і окремих приманках та пастках, є недостатнім. Це пов'язано з особливостями проведення КА та поведінкою ЗПЗ при поширенні і виконанні деструктивних дій. Тобто, вплив ЗПЗ та КА відбувається програмними засобами і, тому, забезпечення протидії винятково програмними засобами не завжди забезпечує бажаний результат, що підтверджується і розробниками систем попередження і виявлення вторгнень та антивірусних засобів. Крім того, організація ефективної взаємодії між комп'ютерними станціями в корпоративних мережах для підтримки мережних застосунків суттєво залежить від часу передачі повідомлень і їх обробки. Враховуючи такі особливості при побудові приманок, пасток та мереж приманок і пасток необхідно синтезувати систему, в якій до процесу виявлення ЗПЗ та КА були б залучені, також, комп'ютерні станції в мережі. І така система могла б в процесі обробки отриманих даних з приманок та пасток приймати рішення

про свої наступні кроки, зокрема і в частині зміни конфігурування та використання комп'ютерних станцій в мережі.

Розглянемо досліджувані системи $\mathfrak{S}$ з виділенням в них множин синтезованих характеристик та властивостей:

$$\mathfrak{S} = \left\{ \left(\mathfrak{v}_1, \mathfrak{v}_2, \ldots, \mathfrak{v}_{10,1}, \mathfrak{v}_{11} \right) \middle| \left(\mathfrak{v}_1, \mathfrak{v}_2, \ldots, \mathfrak{v}_{10,1}, \mathfrak{v}_{11} \right) \in \right.$$
$$\left. \in \mathfrak{V}_1 \times \mathfrak{V}_2 \times \ldots \times \mathfrak{V}_{10,1} \times \mathfrak{V}_{11} \right\} \quad (1)$$

де $\mathfrak{V}_i$ ($i = n_{\mathfrak{V}}$, $n_{\mathfrak{V}}$ – кількість характеристик) – підмножини з елементами, що характеризують особливості архітектури систем; $\mathfrak{v}_{10,1}$ – елемент, що визначає наявність контролера в системі; $\mathfrak{v}_{10,1} \in \mathfrak{V}_{10,1}$; множина $\mathfrak{V}_{10,1}$ – одноелементна множина; $\mathfrak{v}_1, \mathfrak{v}_2, \ldots, \mathfrak{v}_9, \mathfrak{v}_{11}$ – позначення елементів в множинах $\mathfrak{V}_1, \mathfrak{V}_2, \ldots, \mathfrak{V}_9, \mathfrak{V}_{11}$ відповідно.

Таким чином, кількість систем типу $\mathfrak{S}$ є різною, але згідно формули (1) всіх їх поєднує наявність в їх архітектурі контролера. Кількість підмножин $\mathfrak{V}_i$ ($i = n_{\mathfrak{V}}$, $n_{\mathfrak{V}}$ – *кількість характеристик*) може бути різною, зокрема і менше, ніж $n_{\mathfrak{V}}$, але наявність одноелементної множини $\mathfrak{V}_{10,1}$ та множини $\mathfrak{V}_{11}$ в прямому добутку множин є обов'язковим.

Такий поділ архітектури систем за внутрішньою будовою дає змогу визначити необхідні елементи та компоненти в архітектурі системи, яка міститиме контролер та спеціалізований функціонал. Для створення мультикомп'ютерних систем з комбінованими приманками і пастками та контролером прийняття рішень для виявлення та протидії ЗПЗ і КА розробимо концептуальну модель архітектури такого класу систем $\mathfrak{S}$. Наявність такої моделі архітектури мультикомп'ютерних систем дасть змогу здійснити розроблення методологічних основ та методів створення таких систем. Введемо підмножини для таких елементів та компонентів в архітектурі системи і задамо загальну множину $\mathfrak{M}_{\mathfrak{S}}$ для них так:

$$\mathfrak{M}_{\mathfrak{S}} = \bigcup_{i=1}^{\mathfrak{n}_{\mathfrak{M}_{\mathfrak{S}}}} \mathfrak{M}_i, \quad (2)$$

де $\mathfrak{M}_i$ – i – та підмножина для певних елементів та компонентів в архітектурі системи; $i = 1,2,\ldots,\mathfrak{n}_{\mathfrak{M}_{\mathfrak{S}}}$; $\mathfrak{n}_{\mathfrak{M}_{\mathfrak{S}}}$ – кількість підмножин.

Згідно формули (2) задамо кожну з введених підмножин $\mathfrak{M}_i$ ($i = 1,2,\ldots,\mathfrak{n}_{\mathfrak{M}_{\mathfrak{S}}}$; $\mathfrak{n}_{\mathfrak{M}_{\mathfrak{S}}}$ – кількість підмножин) її елементами та встановимо для кожного з них в межах заданих підмножин їх вплив на безпеку системи. Визначення рівня безпеки конкретних елементів та компонентів в архітектурі системи задамо множиною функцій, кожна з функцій якої буде застосовна до елементів конкретної заданої підмножини $\mathfrak{M}_i$ ($i = 1,2,\ldots,\mathfrak{n}_{\mathfrak{M}_{\mathfrak{S}}}$; $\mathfrak{n}_{\mathfrak{M}_{\mathfrak{S}}}$ – кількість підмножин).

Задамо кожну з підмножин елементів та компонентів в архітектурі мультикомп'ютерних систем класу $\mathfrak{S}$ через їх елементи так:

$$\mathfrak{M}_j = \left\{ \mathfrak{m}_{j,1}, \mathfrak{m}_{j,2}, \ldots, \mathfrak{m}_{j,\mathfrak{n}_{\mathfrak{M}_j}} \right\}; \quad (3)$$
$$\mathfrak{M}_{10} = \left\{ \mathfrak{m}_{10,1} \right\},$$

де $\mathfrak{m}_{j,l}$ – l-елемент $\mathfrak{M}_j$ підмножини; $l = 1,2,\ldots,\mathfrak{n}_{\mathfrak{M}_j}$; $j = 1,2,\ldots,9,11,\ldots,\mathfrak{n}_{\mathfrak{M}_{\mathfrak{S}}}$; $\mathfrak{n}_{\mathfrak{M}_j}$ – кількість елементів підмножини $\mathfrak{M}_j$; $\mathfrak{n}_{\mathfrak{M}_{\mathfrak{S}}}$ – кількість підмножин.

Елемент підмножини $\mathfrak{M}_{10}$ є твірним і визначальним для формування архітектури систем класу $\mathfrak{S}$. Тому, підмножина $\mathfrak{M}_{10}$ в формулі (3) визначена одним елементом. Множина $\mathfrak{M}_{\mathfrak{S}}$ буде містити всі елементи підмножин $\mathfrak{M}_j$ ($j = 1,2,\ldots,\mathfrak{n}_{\mathfrak{M}_{\mathfrak{S}}}$; $\mathfrak{n}_{\mathfrak{M}_j}$ – кількість елементів підмножини $\mathfrak{M}_j$). В архітектурі системи класу $\mathfrak{S}$ елементи множини $\mathfrak{M}_{\mathfrak{S}}$ можуть бути всі або може бути частина з них. Також, може бути варіант формування системи згідно входження по одному елементу з кожної із підмножин $\mathfrak{M}_j$ ($j = 1,2,\ldots,\mathfrak{n}_{\mathfrak{M}_{\mathfrak{S}}}$; $\mathfrak{n}_{\mathfrak{M}_j}$ – кількість елементів підмножини $\mathfrak{M}_j$). Тобто варіантів архітектури системи класу $\mathfrak{S}$ з комбінуванням елементів множини $\mathfrak{M}_{\mathfrak{S}}$ може бути багато, але в загальному випадку їх кількість є скінченою.

Введемо множину функцій $\mathfrak{F} = \left\{ \mathfrak{f}_1, \mathfrak{f}_2, \ldots, \mathfrak{f}_{\mathfrak{n}_{\mathfrak{F}}} \right\}$ ($\mathfrak{n}_{\mathfrak{F}}$ – кількість функцій в множині $\mathfrak{F}$), в якій функції будуть виконувати операції над елементами множини $\mathfrak{M}_{\mathfrak{S}}$ як окремими одноелементними множинами. Ці функції будуть діями над елементами множини $\mathfrak{M}_{\mathfrak{S}}$ і результатом цих дій буде формування різних підмножин. Такі підмножини відображатимуть архітектуру системи класу $\mathfrak{S}$. Серед цих функцій будуть такі: об'єднання елементів; вилучення елементу з підмножини; перетворення підмножини з декількох елементів на декілька одноелементних підмножин; об'єднання декількох підмножин, зміна елементів в підмножинах та інших. Таким чином, елементи множини $\mathfrak{M}_{\mathfrak{S}}$ будемо розглядати як одноелементні множини, тоді результатом виконання функцій з множини $\mathfrak{F}$ будуть підмножини з елементів множини $\mathfrak{M}_{\mathfrak{S}}$, тобто множини підмножин $\mathfrak{P}(\mathfrak{M}_{\mathfrak{S}})$. Функції будуть операторами в множині $\mathfrak{P}(\mathfrak{M}_{\mathfrak{S}})$. Функції компонують систему класу $\mathfrak{S}$ з елементів підмножин. При цьому можуть бути одиничні елементи, а також можуть бути комбіновані варіанти з одиничних варіантів. Це досягається за рахунок розширення кількості елементів та, відповідно, їх комбінувань при переході до мно-

жини $\mathfrak{P}(\mathfrak{M}_{\mathfrak{S}})$. Підмножина $\mathfrak{M}_{10}$ буде визначена одним елементом і вона буде присутня в усіх варіантах поєднань підмножин, тобто елемент цієї множини буде в кожному об'єднанні підмножин. Інакше, не всі елементи множини підмножин $\mathfrak{P}(\mathfrak{M}_{\mathfrak{S}})$ формуватимуть архітектури систем класу $\mathfrak{S}$. Таке обмеження до підмножин множини $\mathfrak{P}(\mathfrak{M}_{\mathfrak{S}})$ є обов'язковим. В результати підмножини міститимуть об'єднання мінімум двох різних множин, одна з яких буде містити твірний елемент класу. Тому, одноелементних підмножин в множині $\mathfrak{P}(\mathfrak{M}_{\mathfrak{S}})$ не буде. Таким чином, буде справедливе функційне відображення в множині підмножин $\mathfrak{P}(\mathfrak{M}_{\mathfrak{S}})$:

$$\mathfrak{P}(\mathfrak{M}_{\mathfrak{S}}) \xrightarrow{\mathfrak{F}} \mathfrak{P}(\mathfrak{M}_{\mathfrak{S}}). \quad (4)$$

Співвідношення, яке задане формулою (4), означає, що при виконанні функцій з множини $\mathfrak{F}$ здійснюється перехід до іншого елемента з множини $\mathfrak{P}(\mathfrak{M}_{\mathfrak{S}})$. Для систем класу $\mathfrak{S}$ це відображатиме зміну їх архітектури.

Введемо на множині $\mathfrak{P}(\mathfrak{M}_{\mathfrak{S}})$ предикати, які будуть відображати результати входження та не входження елементів в підмножинах. І, відповідно, вони формуватимуть з елементів та компонентів відомості про певну архітектуру системи класу $\mathfrak{S}$. Задамо множину предикатів $\mathfrak{P}_{\mathfrak{M}} = \left\{\mathfrak{p}_1, \mathfrak{p}_2, \ldots, \mathfrak{p}_{\mathfrak{n}_{\mathfrak{P}_{\mathfrak{M}}}}\right\}$ ($\mathfrak{n}_{\mathfrak{P}_{\mathfrak{M}}}$ – кількість предикатів в множині $\mathfrak{P}_{\mathfrak{M}}$) таким чином, щоб кожен з предикатів буде задавати наявність чи відсутність елементів множини $\mathfrak{M}_{\mathfrak{S}}$ в елементах-підмножинах множини $\mathfrak{P}(\mathfrak{M}_{\mathfrak{S}})$. Якщо підмножини множини $\mathfrak{P}(\mathfrak{M}_{\mathfrak{S}})$ будуть об'єднані і, при цьому, будуть мати різну кількість елементів, тобто буде об'єднано декілька підмножин з різними елементами і різною їх кількістю, тоді до кожної з них будуть застосовуватись різні предикати, а результат їх виконання буде сформовано множиною векторів. Підмножина буде містити елементи множини $\mathfrak{M}_{\mathfrak{S}}$ і їх кількість може бути меншою за загальну кількість елементів множини $\mathfrak{M}_{\mathfrak{S}}$, а вектор з компонентами буде мати кількість компонент, що дорівнює кількості елементів множини $\mathfrak{M}_{\mathfrak{S}}$. Якщо буде дві і більше підмножини з множини $\mathfrak{P}(\mathfrak{M}_{\mathfrak{S}})$, тоді результатом виконання предикатів на їх елементах буде множина векторів такої ж кількості. Таким чином, задамо відображення для елементів множини підмножини $\mathfrak{P}(\mathfrak{M}_{\mathfrak{S}})$, тобто підмножини, у вектор так:

$$\mathfrak{P}(\mathfrak{M}_{\mathfrak{S}}) \xrightarrow{\mathfrak{P}_{\mathfrak{M}}} \mathfrak{W}(\mathfrak{M}_{\mathfrak{S}}), \quad (5)$$

де $\mathfrak{W}(\mathfrak{M}_{\mathfrak{S}})$ – множина векторів.

Елементи множини векторів $\mathfrak{W}(\mathfrak{M}_{\mathfrak{S}})$ позначимо $\mathfrak{w}_{\mathfrak{j}} = \left(\mathfrak{w}_{\mathfrak{j},1}, \mathfrak{w}_{\mathfrak{j},2}, \ldots, \mathfrak{w}_{\mathfrak{j},\mathfrak{n}_{\mathfrak{M}_{\mathfrak{v}}}}\right)$, де $\mathfrak{w}_{\mathfrak{j}} \in \mathfrak{W}(\mathfrak{M}_{\mathfrak{S}})$, $\mathfrak{i} = 1,2,\ldots,\mathfrak{n}_{\mathfrak{M}_{\mathfrak{v}}}$, $\mathfrak{n}_{\mathfrak{M}_{\mathfrak{v}}}$ – загальна кількість елементів і компонентів в архітектурі системи класу $\mathfrak{S}$, $\mathfrak{j} = 1,2,\ldots,\mathfrak{n}_{\mathfrak{W}(\mathfrak{M}_{\mathfrak{S}})}$, $\mathfrak{n}_{\mathfrak{W}(\mathfrak{M}_{\mathfrak{S}})}$ – кількість елементів в множині $\mathfrak{W}(\mathfrak{M}_{\mathfrak{S}})$. Після виконання певної функції з множини функцій $\mathfrak{F}$ згідно з формулою (4) можливим варіантом може бути поєднання двох або більше підмножин чи зменшення кількості поєднаних підмножин. Тоді, в такому випадку кожну з підмножин множини $\mathfrak{P}(\mathfrak{M}_{\mathfrak{S}})$ після виконання предикатів задамо вектором і об'єднаємо їх в множину, яка буде множиною підмножин з елементів векторів $\mathfrak{P}(\mathfrak{W}(\mathfrak{M}_{\mathfrak{S}}))$.

Базовою множиною, в яку буде відображатись результат виконання предикату до одного елементу одноелементної підмножини, буде двохелементна множина $\{0,1\}$. Тоді, для певної підмножини визначення значень координат векторів задамо так:

$$\mathfrak{w}_{\mathfrak{j},\mathfrak{i}} = \begin{cases} 0, \textit{якщо елемент відсутній в підмножині}; \\ 1, \textit{якщо елемент наявний в підмножині}; \end{cases} \quad (6)$$

$$\mathfrak{w}_{10,1} = 1,$$

$\mathfrak{i} = 1,2,\ldots,\mathfrak{n}_{\mathfrak{M}_{\mathfrak{v}}}$, $\mathfrak{n}_{\mathfrak{M}_{\mathfrak{v}}}$ – загальна кількість елементів і компонентів в архітектурі системи класу $\mathfrak{S}$, $\mathfrak{j} = 1,2,\ldots,\mathfrak{n}_{\mathfrak{W}(\mathfrak{M}_{\mathfrak{S}})}$, $\mathfrak{n}_{\mathfrak{W}(\mathfrak{M}_{\mathfrak{S}})}$ – кількість елементів в множині $\mathfrak{W}(\mathfrak{M}_{\mathfrak{S}})$.

Таким чином, архітектура мультикомп'ютерних систем з комбінованими приманками і пастками та контролером прийняття рішень для виявлення та протидії ЗПЗ і КА в корпоративних мережах в поточний момент часу визначатиметься елементом множини $\mathfrak{P}(\mathfrak{M}_{\mathfrak{S}})$ так:

$$\mathfrak{P}_{\mathfrak{k}}(\mathfrak{M}_{\mathfrak{S}}) \in \mathfrak{P}(\mathfrak{M}_{\mathfrak{S}}), \quad (7)$$

де $\mathfrak{P}_{\mathfrak{k}}(\mathfrak{M}_{\mathfrak{S}})$ – $\mathfrak{k}$ – тий елемент множини $\mathfrak{P}(\mathfrak{M}_{\mathfrak{S}})$; $\mathfrak{k} = 1,2,\ldots,\mathfrak{n}_{\mathfrak{V}(\mathfrak{M}_{\mathfrak{S}})}$; $\mathfrak{n}_{\mathfrak{V}(\mathfrak{M}_{\mathfrak{S}})}$ – кількість елементів в множині $\mathfrak{V}(\mathfrak{M}_{\mathfrak{S}})$.

Згідно формул (2)-(7) задамо архітектуру мультикомп'ютерних систем з комбінованими приманками і пастками та контролером прийняття рішень алгебраїчною системою типу $\tau = (\alpha, \beta)$ так:

$$\mathfrak{A}_{\mathfrak{S}} = \mathfrak{P}(\mathfrak{M}_{\mathfrak{S}}), \mathfrak{F}, \mathfrak{P}_{\mathfrak{M}}, \quad (8)$$

де $\mathfrak{F}$ – множина функцій заданих на множині $\mathfrak{P}(\mathfrak{M}_{\mathfrak{S}})$; $\mathfrak{P}_{\mathfrak{M}}$ – множина предикатів заданих на множині $\mathfrak{P}(\mathfrak{M}_{\mathfrak{S}})$; $\alpha = \mathfrak{n}_{\mathfrak{V}(\mathfrak{M}_{\mathfrak{S}})}$, $\beta = 1$ – арності операцій, тому тип системи $\tau = \left(\mathfrak{n}_{\mathfrak{V}(\mathfrak{M}_{\mathfrak{S}})}, 1\right)$; $\mathfrak{n}_{\mathfrak{V}(\mathfrak{M}_{\mathfrak{S}})}$ – кількість елементів в множині $\mathfrak{V}(\mathfrak{M}_{\mathfrak{S}})$.

В системі класу $\mathfrak{S}$, яку задано формулою (8), виділено елементи та компоненти множиною їх властивостей і зв'язки між ними. Елементи відображають в архітектурі системи її цілісні частини, а компоненти відображають розподі-

лені частини. Кожен з таких елементів та компоненті при деталізації міститиме елементи, які будуть ієрархічно структурованими. Система, крім визначених функційних завдань, які формуватимуть процеси в ній, має свої внутрішні особливості сформовані елементами множини $\mathfrak{V}(\mathfrak{M}_{\mathfrak{S}})$ та функціями і предикатами, які на них впливатимуть. Це формуватиме внутрішні процеси в ній і буде надавати їй власні властивості, які формуватимуться згідно поєднання властивостей, визначених для неї в поточний момент часу елементами множини $\mathfrak{V}(\mathfrak{M}_{\mathfrak{S}})$.

Якщо враховувати в архітектурі наявність лише елементів та компонентів з яких вона буде сформована в певні поточні моменти часу і, при цьому, не враховувати події, які призвели до зміни її архітектури, тобто які саме функції були виконані для зміни її архітектури, тоді модель архітектури такого класу систем $\mathfrak{S}$ задамо за формулою (9):

$$\mathfrak{A}_{\mathfrak{M},\mathfrak{S}} = \left\langle \mathfrak{P}(\mathfrak{M}_{\mathfrak{S}}), \mathfrak{P}_{\mathfrak{M}} \right\rangle; \qquad (9)$$

$$\forall \mathfrak{k} : \mathfrak{V}_{10,1} \subset \mathfrak{P}_{\mathfrak{k}}(\mathfrak{M}_{\mathfrak{S}}),$$

де $\mathfrak{P}_{\mathfrak{M}}$ – множина предикатів заданих на множині $\mathfrak{P}(\mathfrak{M}_{\mathfrak{S}})$; $\alpha = \mathfrak{n}_{\mathfrak{V}(\mathfrak{M}_{\mathfrak{S}})}$, $\beta = 1$ – арності операцій, тому тип системи $\tau = \left(\mathfrak{n}_{\mathfrak{V}(\mathfrak{M}_{\mathfrak{S}})}, 1\right)$; $\mathfrak{n}_{\mathfrak{V}(\mathfrak{M}_{\mathfrak{S}})}$ – кількість елементів в множині $\mathfrak{V}(\mathfrak{M}_{\mathfrak{S}})$, $\mathfrak{P}_{\mathfrak{k}}(\mathfrak{M}_{\mathfrak{S}})$ – $\mathfrak{k}$ – тий елемент множини $\mathfrak{P}(\mathfrak{M}_{\mathfrak{S}})$; $\mathfrak{k} = 1,2,\ldots,\mathfrak{n}_{\mathfrak{V}(\mathfrak{M}_{\mathfrak{S}})}$; $\mathfrak{n}_{\mathfrak{V}(\mathfrak{M}_{\mathfrak{S}})}$ – кількість елементів в множині $\mathfrak{V}(\mathfrak{M}_{\mathfrak{S}})$; $\mathfrak{V}_{10,1} = \{\mathfrak{v}_{10,1}\}$; множина $\mathfrak{V}_{10,1}$ – одноелементна множина.

Концептуальну модель архітектури системи задано не тільки виразом, в якому визначено величину $\mathfrak{A}_{\mathfrak{M},\mathfrak{S}}$, але й умовами та обмеженнями щодо її елементів в контексті поставленої проблеми. Модель задана формулою (9) є концептуальною моделлю архітектури такого класу систем $\mathfrak{S}$, оскільки в ній враховані властивості, що синтезовані в її архітектурі, та їх активізація в поточні моменти часу. Ці синтезовані властивості системи реалізовані елементами та компонентами системи. Так задана концептуальна модель є абстрактною моделлю, бо встановлює синтезовані властивості і зв'язки між ними в архітектурі мультикомп'ютерних систем класу $\mathfrak{S}$ їх формальним описом. Для повного задання всіх елементів концептуальної моделі в частині її внутрішнього наповнення потрібно розробити механізми функціонування елементів та компонентів в архітектурі системи, їх взаємодії, логіки формування та виконання процесів в них і між ними. Також, доповненням до запропонованої концептуальної моделі мають бути результати дослідження щодо обмеження до її застосування чи її повноти поширення на розглядуваний клас систем $\mathfrak{S}$. Тобто, запропонована концептуальна модель повинна охоплювати всі архітектури систем класу $\mathfrak{S}$. Крім цього, в так синтезованій архітектурі згідно моделі $\mathfrak{A}_{\mathfrak{M},\mathfrak{S}}$ не повинна формуватись нова якість системи, яка б відносила систему до іншого класу систем, відмінного від класу систем $\mathfrak{S}$. Така якість може формуватись на рівні розв'язання поставлених завдань для системи і реалізованих в ній методів, але не формування іншого класу систем.

Запропонована концептуальна модель архітектури мультикомп'ютерних систем потребуватиме її деталізації до рівня типових елементів та компонентів і, відповідно, доповнення її відображенням зв'язків між ними. Для такого подання систем класу $\mathfrak{S}$ через їх концептуальну модель було доведено повноту такого подання, а також неможливість формування нової якості в таких системах і віднесення їх до іншого класу систем.

Постановка експерименту, показники ефективності та аналіз результатів експериментальних досліджень. Важливими характеристиками в контексті синтезованих систем певного класу та призначення є їх стійкість та рівновага. Оскільки, розглядуваний клас систем охоплює комп'ютерні станції корпоративної мережі, а зв'язок між ними формується за рахунок проміжного програмного забезпечення, що створює мультикомп'ютерну систему, тобто розподілену систему, то стійкість таких систем та їх рівновага є важливими характеристиками і потребують дослідження. Крім того, для аналізу цих двох характеристик потрібно досліджувати час як показник впливу на них і на результат виконання завдання спеціалізованим функціоналом системи.

Введемо три цільові функції $\mathfrak{F}_{\mathfrak{i}}^{\mathfrak{E}}$ ($\mathfrak{i} = 1,2,3$), які будуть характеризувати відповідно стійкість, рівновагу та результат виконання завдання спеціалізованим функціоналом системи. Результат виконання кожної з цих функцій буде відображено в проміжок [0;1]. Середньоарифметичні значення цих трьох функцій будуть встановлювати три коефіцієнти для характеристики системи:

$$\mathfrak{k}_{\mathfrak{F}_{\mathfrak{i}}^{\mathfrak{E}}}^{\mathfrak{E}} = \frac{\sum_{\mathfrak{j}=1}^{\mathfrak{n}_{\mathfrak{F}_{\mathfrak{i}}^{\mathfrak{E}}}^{\mathfrak{E}}} \mathfrak{F}_{\mathfrak{i}}^{\mathfrak{E}}\left(\mathfrak{t}_{\mathfrak{j}}\right)}{\mathfrak{n}_{\mathfrak{F}_{\mathfrak{i}}^{\mathfrak{E}}}^{\mathfrak{E}}}, \qquad (10)$$

де $\mathfrak{t}_{\mathfrak{j}}$ – час від початку функціонування системи; $\mathfrak{j}$ – індекс для часу, що відображає $\mathfrak{j}$ – тий момент фіксування часу в системі; $\mathfrak{j} = 1,2,\ldots,\mathfrak{n}_{\mathfrak{F}_{\mathfrak{i}}^{\mathfrak{E}}}^{\mathfrak{E}}$; $\mathfrak{n}_{\mathfrak{F}_{\mathfrak{i}}^{\mathfrak{E}}}^{\mathfrak{E}}$ – кількість фіксувань часу, які було здійснено системою; $\mathfrak{F}_{\mathfrak{i}}^{\mathfrak{E}}\left(\mathfrak{t}_{\mathfrak{j}}\right)$ ($\mathfrak{i} = 1,2,3$) – функції, які

характеризують стійкість, рівновагу та результат виконання завдання спеціалізованим функціоналом системи; $\mathfrak{k}^{\mathfrak{E}}_{\mathfrak{F}^{\mathfrak{E}}_{i}}$ $(i=1,2,3)$ – коефіцієнти, які характеризують рівні стійкості, рівноваги та результату виконання завдання спеціалізованим функціоналом системи.

Введемо інтегрований показник характеристики системи, в якому будуть відображені коефіцієнти рівнів стійкості, рівноваги та результату виконання завдання спеціалізованим функціоналом системи, а також кількість компонентів системи.

$$\mathfrak{k}^{\mathfrak{E}}_{\mathfrak{I}}=\frac{\sum_{j=1}^{n^{\mathfrak{E}}_{\mathfrak{I}}}\mathfrak{k}^{\mathfrak{E}}_{\mathfrak{F}^{\mathfrak{E}}_{i}}}{n^{\mathfrak{E}}_{\mathfrak{I}}}\cdot\frac{n_1}{n_2}, \qquad (11)$$

де $n^{\mathfrak{E}}_{\mathfrak{I}}$ – кількість коефіцієнтів; $\mathfrak{F}^{\mathfrak{E}}_{i}(t_j)$ $(i=1,2,\ldots,n^{\mathfrak{E}}_{\mathfrak{I}})$ – функції, які характеризують стійкість, рівновагу та результат виконання завдання спеціалізованим функціоналом системи; $\mathfrak{k}^{\mathfrak{E}}_{\mathfrak{F}^{\mathfrak{E}}_{i}}$ $(i=1,2,\ldots,n^{\mathfrak{E}}_{\mathfrak{I}})$ – значення коефіцієнтів; n_1 – кількість компонентів в системі; n_2 – кількість комп'ютерних станцій в корпоративній мережі.

Для проведення експерименту з системою, яку побудовано згідно запропонованої моделі архітектури за формулою (9), було здійснено запуск системи та забезпечено її функціонування протягом 180 годин. Також, для залучення спеціалізованого функціоналу було активовано п'ять штучних комп'ютерних атак. Для обробки подій пов'язаних з ними були використані приманки. Кількість фіксованих часових значень за у весь час функціонування становив 50, тобто за увесь час функціонування системи було отримано 50 значень усіх характеристичних показників.

Результати значень інтегрального коефіцієнту, трьох коефіцієнтів та проміжних величин такі: $\mathfrak{k}^{\mathfrak{E}}_{\mathfrak{F}^{\mathfrak{E}}_{1}}=0{,}78452$; $\mathfrak{k}^{\mathfrak{E}}_{\mathfrak{F}^{\mathfrak{E}}_{2}}=0{,}69433$; $\mathfrak{k}^{\mathfrak{E}}_{\mathfrak{F}^{\mathfrak{E}}_{3}}=0{,}91521$; $n_1=100$; $n_2=120$; $n^{\mathfrak{E}}_{\mathfrak{F}^{\mathfrak{E}}_{i}}=50$. Таким чином, значення інтегрального коефіцієнту $\mathfrak{k}^{\mathfrak{E}}_{\mathfrak{I}}=0{,}66502$. Такі результати отримано саме для систем з наявним контролером. При використанні систем без контролера при проведенні такого ж есперименту було отримано таке значення інтегрального коефіцієнту $\mathfrak{k}^{\mathfrak{E}}_{\mathfrak{I}}=0{,}64892$. Тобто, певна ефективність в таких системах з контролером досягнута. Вона зростатиме при інтенсивнішому і тривалішому їх використанні. Тоді, кількість оброблюваних подій зростатиме, що вимагатиме постійного використання контролеру, який надає переваги системі порівняно з системами без контролера. Крім того, на результат впливає також і кількість розміщених в корпоративній мережі компонент системи. Якщо їх кількість менша за кількість комп'ютерних станцій, то тоді ефективність такої системи буде меншою.

Висновки. Розроблено концептуальну модель архітектури мультикомп'ютерних систем приманок та пасток для виявлення та протидії ЗПЗ та КА, в якій синтезовано характерні властивості. Обов'язковою характерною властивістю в цій моделі є наявність контролера.

Запропонована методика розрахунку ефективності мультикомп'ютерних систем. Результати проведеного експерименту підтверджують перспективність досліджень в напрямі використання контролера в мультикомп'ютерних системах приманок та пасток для виявлення та протидії ЗПЗ та КА.

Запропонована концептуальна модель архітектури мультикомп'ютерних систем потребуватиме її деталізації до рівня типових елементів та компонентів і, відповідно, доповнення її відображенням зв'язків між ними. Тому, напрямом подальших досліджень буде розробка компонентів системи та встановлення зв'язків між ними.

ЛІТЕРАТУРА:

1. Zobal L. D. Kolář, R. Fujdiak. Current State of Honeypots and Deception Strategies in Cybersecurity. *11th International Congress on Ultra-Modern Telecommunications and Control Systems and Workshops (ICUMT). Dublin, Ireland*, 2019. P. 1-9.

2. Almeshekah M.H., Spafford E.H. Cyber Security Deception. Cyber Deception. *Springer, Cham*, 2016.

3. Fraunhol D., Anton S.D., Lipps C., Reti D., Krohmer D., Pohl F., Tammen M., Schotten D. Demystifying Deception Technology: A Survey. arXiv:1804.06196v1 [cs.CR] 17 Apr 2018 1, 2.

4. Zielinski D., Kholidy H.A. An Analysis of Honeypots and their Impact as a Cyber Deception Tactic. arXiv:2301.00045v1.

5. Acosta, J.C., Basak, A., Kiekintveld, C., Kamhoua C. Lightweight On-Demand Honeypot Deployment for Cyber Deception. *In: Gladyshev, P., Goel, S., James, J., Markowsky, G., Johnson, D. (eds) Digital Forensics and Cyber Crime. ICDF2C 2021. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*. *Springer, Cham.* 2019. Vol. 441. P. 1-18.

6. Anwar A.H., Kamhoua C.A., Leslie N.O., Kiekintveld C. Honeypot Allocation for Cyber Deception Under Uncertainty. *IEEE Transactions on Network and Service Management*. Sept. 2022. Vol. 19, no. 3, pp. 3438-3452.

7. Tsikerdekis M., Zeadally S., Schlesener A. Sklavos N., Approaches for Preventing Honeypot Detection and Compromise. *2018 Global Information Infrastructure and Networking Symposium (GIIS), Thessaloniki, Greece*. 2018. P. 1-6.

8. Mphago B., Shedden M.D.M. Deception in Web Application Honeypots: Case of Glastopf. *International Journal of Cyber-Security and Digital Forensics.* 2017. Vol. 6: P. 179-185.

9. Sayed M.A., Anwar A.H., Kiekintveld C. Kamhoua C. Honeypot Allocation for Cyber Deception in Dynamic Tactical Networks: A Game Theoretic Approach. arXiv:2308.11817v2 [cs.GT] 5 Sep 2023.

10. Anwar A.H., Kamhoua C.A. Cyber Deception using Honeypot Allocation and Diversity: A Game Theoretic Approach. *2022 IEEE 19th Annual Consumer Communications & Networking Conference (CCNC)*, *Las Vegas, NV, USA*. 2022. P. 543-549.

11. Katakwar H., Aggarwal P., Maqbool Z. Front V.D. Influence of Network Size on Adversarial Decisions in a Deception Game Involving Honeypots. *Front. Psychol.*, 25 September 2020 Sec. Cognition Volume 11, P. 1-13.

12. Çeker H., Zhuang J., Upadhyaya S., La Q.D. Soong, BH. Deception-Based Game Theoretical Approach to Mitigate DoS Attacks*. Decision and Game Theory for Security. GameSec 2016. Lecture Notes in Computer Science*. *Springer, Cham.* Vol 9996.

13. Huang L. Zhu Q. Duplicity Games for Deception Design With an Application to Insider Threat Mitigation. *IEEE Transactions on Information Forensics and Security.* 2021. Vol. 16, P. 4843-4856.

14. Anwar A.H., Zhu M., Z. Z., Cho J. -H., Kamhoua C. A., Singh M. P. Honeypot-Based Cyber Deception Against Malicious Reconnaissance via Hypergame Theory. *GLOBECOM 2022 – 2022 IEEE Global Communications Conference, Rio de Janeiro, Brazil.* 2022, P. 3393-3398.

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

16. Priya V.S.D., Chakkaravarthy S.S. Containerized cloud-based honeypot deception for tracking attackers. *Sci Rep 13.* 2023. Vol. 1437

17. Sikos, L.F., Valli, C., Grojek, A.E. et al. CamDec: Advancing Axis P1435-LE video camera security using honeypot-based deception. *J Comput Virol Hack Tech (2023)*. 2023.

18. Feng, M. et al. A Novel Deception Defense-Based Honeypot System for Power Grid Network. *In: Qiu, M., Gai, K., Qiu, H. (eds) Smart Computing and Communication. SmartCom 2021. Lecture Notes in Computer Science. Springer, Cham.* 2022. Vol. 13202*.*

19. Abe S., Tanaka Y., Uchida Y., Horata S. Developing Deception Network System with Traceback Honeypot in ICS Network. *SICE Journal of Control, Measurement, and System Integration*, *11:4*. 2018. P. 372-379.

20. 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, P. 6-10.

21. URL: https://www.acalvio.com/product/ 04.09.2023

22. URL: https://www.countercraftsec.com/ 13.09.2023

23. URL: https://www.sentinelone.com/surfaces/identity/ 12.09.2023

24. URL:https://www.proofpoint.com/us/illusive-is-now-proofpoint 12.09.2023

25. URL: https://fidelissecurity.com/platforms/fidelis-deception/ 13.09.2023

REFERENCES:

1. Zobal L. D. Kolář, R. Fujdiak. (2019). Current State of Honeypots and Deception Strategies in Cybersecurity. *11th International Congress on Ultra-Modern Telecommunications and Control Systems and Workshops (ICUMT)*. *Dublin, Ireland*. P. 1-9.

2. Almeshekah M.H., Spafford E.H. (2016). Cyber Security Deception. Cyber Deception. *Springer, Cham*.

3. Fraunhol D., Anton S.D., Lipps C., Reti D., Krohmer D., Pohl F., Tammen M., Schotten D. Demystifying Deception Technology: A Survey. arXiv:1804.06196v1 [cs.CR] 17 Apr. 2018 1, 2.

4. Zielinski D., Kholidy H.A. An Analysis of Honeypots and their Impact as a Cyber Deception Tactic. arXiv:2301.00045v1.

5. Acosta, J.C., Basak, A., Kiekintveld, C., Kamhoua C. (2019). Lightweight On-Demand Honeypot Deployment for Cyber Deception. *In: Gladyshev, P., Goel, S., James, J., Markowsky, G., Johnson, D. (eds) Digital Forensics and Cyber Crime. ICDF2C 2021. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*. *Springer, Cham.* Vol. 441. P. 1-18.

6. Anwar A.H., Kamhoua C.A., Leslie N.O., Kiekintveld C. (2022). Honeypot Allocation for Cyber Deception Under Uncertainty. *IEEE Transactions on Network and Service Management*. Sept. Vol. 19, no. 3, pp. 3438-3452.

7. Tsikerdekis M., Zeadally S., Schlesener A. Sklavos N. (2018). Approaches for Preventing Honeypot Detection and Compromise. *2018 Global Information Infrastructure and Networking Symposium (GIIS), Thessaloniki, Greece*. P. 1-6.

8. Mphago B., Shedden M.D.M. (2017). Deception in Web Application Honeypots: Case of Glastopf. *International Journal of Cyber-Security and Digital Forensics.* Vol. 6: P. 179-185.

9. Sayed M.A., Anwar A.H., Kiekintveld C. Kamhoua C. Honeypot Allocation for Cyber Deception in Dynamic Tactical Networks: A Game Theoretic Approach. arXiv:2308.11817v2 [cs.GT] 5 Sep 2023.

10. Anwar A.H., Kamhoua C.A. (2022). Cyber Deception using Honeypot Allocation and Diversity: A Game Theoretic Approach. *2022 IEEE 19th Annual Consumer Communications & Networking Conference (CCNC)*, *Las Vegas, NV, USA*. P. 543-549.

11. Katakwar H., Aggarwal P., Maqbool Z. Front V.D. Influence of Network Size on Adversarial Decisions in a Deception Game Involving Honeypots. *Front. Psychol.*, 25 September 2020 Sec. Cognition Volume 11, P. 1-13.

12. Çeker H., Zhuang J., Upadhyaya S., La Q.D. Soong, BH. Deception-Based Game Theoretical Approach to Mitigate DoS Attacks*. Decision and Game Theory for Security. GameSec 2016. Lecture Notes in Computer Science*. *Springer, Cham.* Vol 9996.

13. Huang L. Zhu Q. (2021). Duplicity Games for Deception Design With an Application to Insider Threat Mitigation. *IEEE Transactions on Information Forensics and Security.* Vol. 16, P. 4843-4856.

14. Anwar A.H., Zhu M., Z. Z., Cho J. -H., Kamhoua C. A., Singh M. P. (2022). Honeypot-Based Cyber Deception Against Malicious Reconnaissance via Hypergame Theory. *GLOBECOM 2022 – 2022 IEEE Global Communications Conference, Rio de Janeiro, Brazil.* P. 3393-3398.

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

16. Priya V.S.D., Chakkaravarthy S.S. (2023). Containerized cloud-based honeypot deception for tracking attackers. *Sci Rep 13.* Vol. 1437.

17. Sikos, L.F., Valli, C., Grojek, A.E. et al. (2023). CamDec: Advancing Axis P1435-LE video camera security using honeypot-based deception. *J Comput Virol Hack Tech (2023)*.

18. Feng, M. et al. (2022). A Novel Deception Defense-Based Honeypot System for Power Grid Network. *In: Qiu, M., Gai, K., Qiu, H. (eds) Smart Computing and Communication. SmartCom 2021. Lecture Notes in Computer Science. Springer, Cham.* Vol. 13202.

19. Abe S., Tanaka Y., Uchida Y., Horata S. (2018). Developing Deception Network System with Traceback Honeypot in ICS Network. *SICE Journal of Control, Measurement, and System Integration*, *11:4*. P. 372-379.

20. Wegerer M., Tjoa S. (2016). Defeating the Database Adversary Using Deception – A MySQL Database Honeypot. *International Conference on Software Security and Assurance (ICSSA)*, *Saint Pölten, Austria.* P. 6-10.

21. Retrieved from https://www.acalvio.com/product/ 04.09.2023

22. Retrieved from https://www.countercraftsec.com/ 13.09.2023

23. Retrieved from https://www.sentinelone.com/surfaces/identity/ 12.09.2023

24. Retrieved from https://www.proofpoint.com/us/illusive-is-now-proofpoint 12.09.2023

25. Retrieved from https://fidelissecurity.com/platforms/fidelis-deception/ 13.09.2023