

АРХІТЕКТУРА МУЛЬТИАГЕНТНОЇ СИСТЕМИ ДЛЯ ТЕСТУВАННЯ НА ПРОНИКНЕННЯ

Вінницький національний технічний університет

Анотація

Запропоновано архітектуру мультиагентної системи для тестування на проникнення, яка дозволяє автоматизувати процеси виявлення, експлуатації та оцінки вразливостей інформаційних систем. Описано основні компоненти системи та їхню взаємодію, що сприяє підвищенню ефективності тестування та оптимізації ресурсів кібербезпеки.

Ключові слова: штучний інтелект, мультиагентна система, тестування на проникнення, кібербезпека, автоматизація, вразливості.

Abstract

The proposed architecture of a multi-agent system for penetration testing enables automation of vulnerability detection, exploitation, and assessment processes. The key components and their interactions are described, improving the efficiency of security testing and optimizing cybersecurity resources..

Keywords: artificial intelligence, multi-agent system, penetration testing, cybersecurity, automation, vulnerabilities.

Вступ

В умовах сучасної війни, кіберпростір стає ключовим полем бою, що підкреслює необхідність посилення кібербезпеки держави. Захист критичної інформаційної інфраструктури є пріоритетом для забезпечення національної безпеки [1, 2]. Зловмисники використовують розвинені методи атак, включаючи фішинг, експлуатацію вразливостей нульового дня та складні деструктивні кібератаки, спрямовані на підрив функціонування державних та приватних систем. Саме тому необхідно впроваджувати сучасні підходи до моніторингу подібних загроз, оцінки вразливостей та активного тестування на проникнення для виявлення та усунення потенційних загроз.

Результати дослідження

Зростання кількості та складності кібератак вимагає впровадження ефективних методів оцінки вразливостей інформаційних систем [2]. Згідно з сучасними тенденціями в кібербезпеці, тестування на проникнення є одним із ключових підходів для виявлення та усунення потенційних загроз. Тестування на проникнення (penetration testing, pentest) – це метод оцінки безпеки інформаційних систем, який полягає в імітації кібератак для виявлення вразливостей, оцінки ризиків і перевірки ефективності засобів захисту. Даний підхід дозволяє проактивно усувати слабкі місця та покращувати стратегії кіберзахисту.

Автоматизація цього процесу може значно зменшити використання ресурсів та часу, а також, що дуже важливо, підвищити точність виявлення помилок, зменшуючи людський фактор. На разі, використовуються різні методи автоматизації таких завдань, але більшість з них фокусуються на одному напрямку або ж етапі тестування на проникнення. Було б доцільно розробити систему, яка автоматизовує виконання більшості завдань і мінімізує втручання тестувальника у процес тестування на проникнення. Одним із підходів до такої автоматизації може бути використання мультиагентної системи на основі штучного інтелекту, її використання дозволить підвищити ефективність та точність тестування [3].

Моделі машинного навчання надають наступні можливості [4]:

- виявлення аномалії у мережевому трафіку та передбачення потенційних вразливостей;
- генерування динамічних сигнатур атак, які не розпізнаються традиційними антивірусами;
- аналіз лог-файлів для автоматичного виявлення підозрілих дій.

Мультиагентна система – це розподілена система, що складається з автономних програмних агентів, які взаємодіють між собою для досягнення спільної мети. Кожен агент виконує певні завдання, обмінюється інформацією та приймає рішення на основі локальної чи глобальної стратегії, що робить такі системи ефективними для складних обчислювальних завдань, зокрема в кібербезпеці, тестуванні на проникнення та управлінні великими даними [5]. Основні компоненти мультиагентної системи для тестування на проникнення і схему їх взаємодії представлено на рисунку 1.

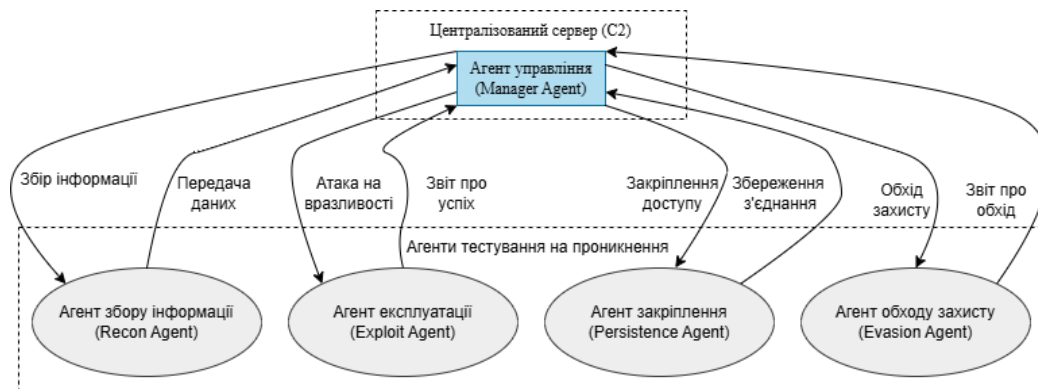


Рис. 1 - Основні компоненти мультиагентної системи для тестування на проникнення

Дана схема демонструє структуру та взаємодію основних компонентів мультиагентної системи для тестування на проникнення. Вона складається з централізованого сервера управління (Manager Agent) та чотирьох основних агентів, кожен із яких виконує певні завдання в процесі тестування безпеки інформаційної системи. Центральний сервер управління (Manager Agent) виконує роль координатора всіх процесів у системі та надсилає команди агентам, отримує від них звіти та керує тестуванням. В задачі агента збору інформації (Recon Agent) входить отримання завдання від Manager Agent, OSINT-збір, сканування портів, аналіз мережевих сервісів та передача знайдених даних про ціль назад на центральний сервер для подальшого аналізу. Агент експлуатації вразливостей (Exploit Agent) отримує від Manager Agent інформацію про знайдені вразливості, використовує експлойти та відомі CVE для перевірки стійкості системи і надсилає звіт про успішність атак назад до Manager Agent. В свою чергу агент закріплення (Persistence Agent) отримує команди від Manager Agent для створення бекдорів, прихованих користувачів або інших механізмів доступу, налаштовує механізми довготривалого збереження доступу до системи і також надсилає зворотний звіт про встановлені точки доступу. І на останок агент обходу захисту (Evasion Agent) отримує завдання з обходу систем виявлення (EDR, IDS, IPS), використовує методи стелс-атаки, шифрування, зміни сигнатур для уникнення виявлення. В кінці відбувається передавання даних про успішність обходу назад до Manager Agent.

Тобто в процесі тестування Recon Agent проводить попередню розвідку, збираючи інформацію про цільову систему, та передає ці дані Manager Agent. Після аналізу отриманих відомостей Manager Agent передає їх Exploit Agent, який здійснює атаки для виявлення вразливостей. Якщо експлуатація успішна, Persistence Agent отримує завдання на закріплення доступу до системи. Одночасно Evasion Agent перевіряє можливість обходу механізмів захисту, таких як антивіруси, IDS або EDR. Усі зібрані результати передаються до Manager Agent, який формує фінальний звіт про тестування.

Запропонована мультиагентна система дозволяє автоматизувати процес тестування на проникнення, ефективно оцінювати стан кібербезпеки та виявляти потенційні загрози, що є критично важливим у сучасних умовах кіберзагроз.

Принцип роботи системи можна розглянути на теоретичному сценарії перевірки фінансової компанії, що вирішила перевірити безпеку своєї IT-інфраструктури перед впровадженням нової онлайн-платформи для клієнтів. Для цього була запущена мультиагентна система тестування на проникнення, яка автоматично аналізувала можливі вразливості в декілька етапів.

На першому етапі Recon Agent здійснив збір інформації про систему, аналізуючи відкриті джерела та виконуючи сканування мережі. Він виявив відкриті порти, активні служби та потенційні вразливості, після чого передав отримані дані Manager Agent для подальшого аналізу.

Наступним кроком Exploit Agent використав знайдені уразливості для перевірки можливості проникнення в систему. Він тестував веб-сервери на наявність SQL-ін'єкцій, проводив атаки на паролі та застосовував інструменти, такі як Metasploit, для імітації реальних загроз. Після успішного виконання атак результати передавалися на сервер управління.

Щоб забезпечити довготривалий доступ, Persistence Agent використовував техніки створення прихованих облікових записів і закріплення бекдорів у системі. Одночасно Evasion Agent працював над тим, щоб уникнути виявлення антивірусами та системами захисту, змінюючи сигнатури атак та аналізуючи журнали подій.

Після завершення тестування Manager Agent зібрав усі результати та сформував звіт із детальним описом знайдених вразливостей і рекомендаціями щодо їх усунення. Це дозволило компанії усунути критичні недоліки ще до запуску платформи та підвищити рівень безпеки, запобігаючи потенційним кібератакам.

Для ефективного функціонування система має бути розгорнута в гібридному середовищі, поєднуючи локальні сервери та хмарні ресурси. Центральний C2-сервер (Command & Control) виконує

координацію агентів, збирає результати тестувань і забезпечує їхню інтеграцію з іншими системами кібербезпеки. Для зберігання результатів тестувань використовується реляційна база даних (наприклад, PostgreSQL або MySQL), що дозволяє ефективно обробляти великі обсяги даних. Інтеграція через API дозволяє зв'язати систему з зовнішніми інструментами, такими як Metasploit, Cobalt Strike, Shodan, що дає змогу використовувати актуальні бази вразливостей та автоматизувати експлуатацію знайдених вразливостей.

Перспективи розвитку подібних систем включають автоматизоване навчання агентів, використання генеративних моделей для передбачення нових загроз, а також інтеграцію з SIEM-системами та різними фреймворками для забезпечення комплексного підходу до кібербезпеки [6].

Для практичної реалізації запропонованої мультиагентної архітектури можуть бути використані сучасні фреймворки машинного навчання з підкріпленням, такі як OpenAI Gym, Stable-Baselines3 або Ray RLLib. Крім того, перспективним напрямком є побудова системи на базі великих мовних моделей з використанням таких мультиагентних фреймворків, як CrewAI, LangGraph, AutoGPT та ін. [7]. Використання мультиагентного підходу на основі великих мовних моделей у системах тестування на проникнення дозволить автоматизувати складні сценарії атак, адаптуючись до нових загроз у режимі реального часу. Такий підхід підвищує ефективність виявлення вразливостей, зменшуючи потребу в ручному аналізі та прискорюючи процеси кіберзахисту.

Висновки

Мультиагентні системи для тестування на проникнення значно підвищують ефективність виявлення вразливостей та покращують рівень кіберзахисту. Вони дозволяють автоматизувати процеси аналізу та експлуатації слабких місць у системах безпеки, що зменшує ризики атак та скорочує час реагування на загрози.

Застосування штучного інтелекту та автоматизованих підходів робить такі системи незамінними для організацій, які прагнуть забезпечити високий рівень захисту від кібератак. Впровадження мультиагентної архітектури дозволяє підвищити гнучкість та масштабованість тестування, що особливо важливо в умовах зростаючих загроз та гібридної війни [8].

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Міненко С., Кочнева В., Бабич Є. Оцінка рівня кібербезпеки України в умовах війни // Європейський науковий журнал економічних та фінансових інновацій. – 2024. – Т. 2, № 14. – С. 487–500. DOI: 10.32750/2024-0243. URL: https://www.researchgate.net/publication/388310208_OCINKA_RIVNA_KIBERBEZPEKI_UKRAINI_V_UMOVAN_VIJNI (дата звернення: 18.03.2025).
2. Стратегія кібербезпеки України/ Рада національної безпеки і оборони України, 2021. URL: https://www.rnbo.gov.ua/files/2021/STRATEGIYA%20KYBERBEZPEKI/proekt%20strategii_kyberbezpek_i_Ukr.pdf (дата звернення: 18.03.2025).
3. Толкачова А., Посувайло М.-М. Тестування на проникнення з використанням глибокого навчання з підкріпленням. «Кібербезпека: освіта, наука, техніка» 2024. С. 17–30. URL: <https://doi.org/10.28925/2663-4023.2024.23.1730> (дата звернення: 18.03.2025).
4. Притула А. В., Куперштейн Л. М. Застосування штучного інтелекту для тестування на проникнення. LIII Всеукраїнська науково-технічна конференція факультету інформаційних технологій та комп'ютерної інженерії (2024). URL: <https://conferences.vntu.edu.ua/index.php/itpf/all-fitki-2024/paper/view/20606/17127> (дата звернення: 22.03.2025).
5. Купін. А. І., Косей М. П. Огляд архітектури мультиагентних систем та алгоритмів ройового інтелекту. Вчені записки ТНУ імені В.І. Вернадського. Серія: Технічні науки. 2024. № 2. С. 98–104. URL: <https://doi.org/10.32782/2663-5941/2024.2/14> (дата звернення: 22.03.2025).
6. Притула А.В., Куперштейн Л. М., Маліновський В. І., аналіз технологій тестування на проникнення web-додатки. Наукові праці Вінницького національного університету. 2024. № 2. URL: <https://doi.org/10.31649/2307-5376-2024-2-45-53> (дата звернення: 22.03.2025).
7. Bhavsar P., Mastering agents: langgraph vs autogen vs crew AI - galileo AI. *Galileo AI*. URL: <https://www.galileo.ai/blog/mastering-agents-langgraph-vs-autogen-vs-crew> (дата звернення: 22.03.2025).
8. Топчій В.В., Бодунова О.М. Проблемні питання забезпечення кібербезпеки в Україні. Електронне наукове видання «Аналітично-порівняльне правознавство», 2025. № 01. DOI: <https://doi.org/10.24144/2788-6018.2025.01.110> (дата звернення: 18.03.2025).

Притула Андрій Вікторович – студент групи 125-23а, факультет інформаційних технологій та комп’ютерної інженерії, Вінницький національний технічний університет, Вінниця, e-mail: andrik.pritula@gmail.com.

Куперштейн Леонід Михайлович – к.т.н., доцент кафедри захисту інформації, Вінницький національний технічний університет, м. Вінниця email: kupershtein.lm@gmail.com

Prytula Andrii V. – Student of Faculty of Information Technologies and Computer Engineering, Vinnytsia National Technical University, e-mail: andrik.pritula@gmail.com.

Kupershtein Leonid M. – PhD, Associated Professor of Information Protection Chair, Vinnytsia National Technical University, Vinnytsia, email: kupershtein.lm@gmail.com