

АВТОМАТИЗАЦІЯ ПРОЦЕСІВ ПОБУДОВИ ТА СУПРОВОДУ КОМПЛЕКСНИХ СИСТЕМ ЗАХИСТУ ІНФОРМАЦІЇ

Вінницький національний технічний університет

Анотація

У тезах розглянуто проблематику створення та впровадження комплексних систем захисту інформації (КСЗІ) в сучасних інформаційно-телекомунікаційних системах (ІТС). Актуальність дослідження зумовлена стрімким зростанням обсягів даних, що підлягають захисту згідно з вимогами НД ТЗІ (нормативних документів системи технічного захисту інформації), та невідповідністю традиційних «ручних» методів побудови систем захисту динаміці сучасних кіберзагроз. Проаналізовано основні недоліки класичного підходу до створення КСЗІ, серед яких: висока трудомісткість розробки проєктної документації, значний вплив людського фактору при налаштуванні засобів захисту, статичність впроваджених політик безпеки та складність підтримки актуального стану захищеності після атестації системи. Особливу увагу приділено проблемі «паперової безпеки», коли система формально відповідає вимогам, але технічно залишається вразливою через помилки конфігурації. Запропоновано методiku автоматизації етапів життєвого циклу КСЗІ шляхом впровадження підходів *Infrastructure as Code (IaC)* та *Security as Code (SaC)*. Розглянуто можливість використання систем управління конфігураціями (наприклад, *Ansible*, *Chef*, *Puppet*) для автоматизованого розгортання політик безпеки, що гарантує ідентичність налаштувань на всіх вузлах мережі та виключає помилки адміністратора. Також досліджено ефективність застосування протоколу автоматизації контенту безпеки (SCAP) для безперервного моніторингу відповідності (compliance) системи встановленим вимогам. Обґрунтовано, що перехід від періодичного ручного контролю до автоматизованого моніторингу дозволяє підтримувати КСЗІ в актуальному стані в режимі реального часу, значно скорочує час на підготовку до державної експертизи та зменшує вартість експлуатації системи захисту.

Ключові слова: комплексна система захисту інформації, автоматизація процесів захисту, нормативна база ТЗІ, людський фактор, управління конфігураціями, *Security as Code*, SCAP, безперервний моніторинг, політики безпеки.

Abstracts

The abstract discusses the issues of creating and implementing comprehensive information protection systems (CIPS) in modern information and telecommunications systems (ITS). The relevance of the study is due to the rapid growth in the volume of data subject to protection in accordance with the requirements of ND TZI (regulatory documents of the technical information protection system) and the inadequacy of traditional 'manual' methods of building protection systems to the dynamics of modern cyber threats. The main shortcomings of the classical approach to creating CIS are analysed, including: the high labour intensity of developing project documentation, the significant influence of the human factor in configuring protection measures, the static nature of the security policies implemented, and the complexity of maintaining the current state of security after system certification. Particular attention is paid to the problem of 'paper security,' when the system formally meets the requirements but remains technically vulnerable due to configuration errors. A methodology for automating the stages of the ISMS life cycle by implementing *Infrastructure as Code (IaC)* and *Security as Code (SaC)* approaches is proposed. The possibility of using configuration management systems (e.g., *Ansible*, *Chef*, *Puppet*) for automated deployment of security policies is considered, which guarantees the identity of settings on all network nodes and eliminates administrator errors. The effectiveness of using the Security Content Automation Protocol (SCAP) for continuous monitoring of system compliance with established requirements was also investigated. It was substantiated that the transition from periodic manual control to automated monitoring allows maintaining the ISMS up to date in real time, significantly reduces the time required for preparation for state examination, and reduces the cost of operating the protection system.

Keywords: comprehensive information security system, automation of security processes, regulatory framework for information security, human factor, configuration management, *Security as Code*, SCAP, continuous monitoring, security policies.

Вступ

В умовах стрімкого розвитку цифровізації в державному та комерційному секторах України захист інформації в інформаційно-телекомунікаційних системах (ІТС) набуває критичного значення. Відповідно до вимог чинного законодавства України, зокрема Закону України «Про захист інформації в інформаційно-телекомунікаційних системах», побудова КСЗІ є обов'язковою умовою для роботи з державними інформаційними ресурсами та персональними даними [1]. Процес розробки КСЗІ базується на нормативних документах системи технічного захисту інформації (НД ТЗІ), які були створені у 90-х та 2000-х роках. Хоча цей підхід формує фундаментальну методологічну базу, на практиці часто виявляється занадто статичним, бюрократичним та ресурсномістким для сучасних динамічних ІТ-середовищ.

Актуальність теми дослідження зумовлена протиріччям між необхідністю швидкого впровадження інформаційних сервісів та довготривалим, переважно ручним процесом забезпечення безпеки. Традиційна модель побудови КСЗІ передбачає значний обсяг роботи з документами: розробку Технічного завдання, Моделі загроз, Плану захисту, а також ручне налаштування засобів захисту (ОС, СУБД, мережевого обладнання). Такий підхід має два суттєві недоліки. Перш за все, людський фактор: системні адміністратори можуть припуститися помилок при конфігурації сотень параметрів безпеки, що залишає систему вразливою навіть за наявності атестата відповідності. По-друге, проблема актуалізації: сучасні інформаційні системи постійно оновлюються, змінюються та масштабуються їхня архітектура (особливо в хмарних середовищах), тоді як КСЗІ фіксує конкретний стан системи на момент перевірки, через що вона втрачає свою актуальність.

Відповіддю на ці виклики є впровадження методів автоматизації процесів захисту інформації. Світовий досвід, зокрема стандарти NIST (National Institute of Standards and Technology), пропонує перехід від періодичного контролю (periodic compliance) до безперервного моніторингу (continuous monitoring) та концепції «Безпека як код» (Security as Code) [2]. Використання інструментів автоматизованого управління конфігураціями (наприклад, Ansible, Puppet, Chef) та протоколів автоматизації вмісту безпеки (SCAP) дозволяє не лише пришвидшити побудову КСЗІ, але й гарантувати, що налаштування безпеки залишаються незмінними та коректними протягом усього життєвого циклу системи.

Таким чином, розробка та дослідження методів автоматизації побудови КСЗІ є важливим науково-практичним завданням, яке дозволить підвищити реальний рівень захищеності державних та корпоративних ІТС, зменшити вартість їх обслуговування та мінімізувати вплив людського фактора на безпеку інформації.

Результати дослідження

В основі запропонованого підходу лежить концепція створення спеціалізованого програмного комплексу, який функціонує як експертна система для автоматизованого синтезу організаційно-розпорядчої та проектної документації на комплексну систему захисту інформації. Ключова ідея підходу полягає у відмові від традиційного ручного формування документів і переході до параметричної моделі об'єкта захисту. У межах цієї моделі вхідні технічні характеристики інформаційно-телекомунікаційної системи трансформуються у фінальні нормативні документи шляхом застосування алгоритмічних правил кореляції з вимогами діючих стандартів, зокрема НД ТЗІ 2.5-004-99. Запропоноване рішення реалізує повний цикл підготовки документації завдяки послідовну функціонуванню чотирьох основних модулів: формалізації даних, логічної обробки, шаблонізації та генерації вихідного пакета документів [3].

На першому етапі здійснюється формалізація вхідних даних за допомогою модуля інтерфейсу, який абстрагує користувача від складної бюрократичної термінології. Замість підготовки довільних текстових описів адміністратор безпеки або розробник системи взаємодіє зі структурованими формами та інтерактивними опитувальниками. Алгоритм збору даних передбачає ідентифікацію об'єкта, інвентаризацію апаратних та програмних активів, а також визначення категорії автоматизованої системи. Наприклад, система формує конкретні запитання щодо наявності підключень до глобальних мереж, оброблення інформації різних категорій доступу та механізмів розмежування прав користувачів. На основі отриманих бінарних відповідей програмний комплекс автоматично визначає клас автоматизованої системи та формує відповідну матрицю доступу. Такий підхід дозволяє мінімізувати ризик помилкового визначення категорії системи, спричиненого людським фактором.

Ключовим елементом запропонованого рішення є модуль логічної обробки та кореляції вимог, який містить оцифровану базу знань нормативних документів ТЗІ. Цей модуль виконує динамічний мапінг технічних характеристик системи на вимоги законодавства. Алгоритм функціонує на основі системи умовних правил: якщо у вхідних даних зафіксовано наявність певного технічного компонента, система автоматично активує відповідний набір вимог до захисту. Під час синтезу Моделі загроз використовується шаблонний підхід, за якого наявність конкретної технології, наприклад бездротових мереж Wi-Fi, автоматично додає до переліку актуальних загроз пункти про перехоплення радіоефіру чи нав'язування хибної точки доступу, а наявність веб-сервера ініціює включення загроз прикладного рівня. Такий підхід забезпечує повноту та релевантність моделі загроз реальній архітектурі інформаційної системи [4].

Третій етап реалізується модулем шаблонізації, який забезпечує трансформацію логічних структур у людинозрозумілий формат офіційних документів. Функціональність модуля базується на механізмах підстановки змінних і використанні умовних текстових блоків. У результаті окремі розділи документа формуються автоматично лише за наявності відповідних технічних компонентів у складі інформаційної системи. Наприклад, вимоги до захисту середовища віртуалізації у технічному завданні формуються лише у випадку використання гіпервізора. У разі відсутності компонента алгоритм автоматично виключає цей блок тексту та коригує нумерацію розділів, забезпечуючи логічну цілісність структури документа. Важливою перевагою такого підходу є забезпечення перехресної узгодженості (cross-consistency) між різними документами пакета. Кожній загрозі у Моделі загроз гарантовано відповідає вимога у Технічному завданні та пункт перевірки у Програмі та методиці випробувань, що є критично важливим для успішного проходження державної експертизи [4].

Результатом роботи алгоритму є автоматично сформований та узгоджений пакет документів, що включає Акт категоріювання, Технічне завдання, Модель загроз, План захисту, Паспорт формуляра та Програму методики випробувань. Практична цінність запропонованого методу полягає у мінімізації семантичних помилок, оскільки виключаються розбіжності між реальною конфігурацією системи та її документальним описом. Крім того, забезпечується висока швидкість адаптації документації: у разі зміни апаратної конфігурації або модернізації системи достатньо оновити вхідні параметри, після чого програмний комплекс за лічені секунди регенерує весь пакет документів обсягом у сотні сторінок.

Другим важливим напрямом автоматизації побудови КСЗІ є впровадження технічних засобів автоматизованого управління конфігураціями, що дозволяє реалізувати концепцію «Інфраструктура як код» (Infrastructure as Code – IaC) у контексті виконання вимог нормативних документів технічного захисту інформації (НД ТЗІ). Традиційний підхід до налаштування операційних систем та прикладного програмного забезпечення базується на ручній праці системних адміністраторів. Така модель характеризується високою ймовірністю виникнення помилок, складністю масштабування та проблемою «дрейфу конфігурацій» (configuration drift), коли фактичні налаштування системи з часом відхиляються від еталонного стану, зафіксованого в проєктній документації. Для розв'язання цієї проблеми пропонується використання спеціалізованих скриптових мов (Python, Bash, PowerShell) та систем управління конфігураціями (Ansible, Chef, Puppet, SaltStack) як інструментів детермінованого приведення інформаційно-телекомунікаційної системи у відповідність до вимог безпеки [5].

Основна ідея підходу полягає у заміні текстових інструкцій для адміністратора (наприклад, «відкрийте файл конфігурації та змініть параметр X на Y») на виконуваний програмний код, який автоматично реалізує необхідні налаштування без участі людини. На базовому рівні автоматизації це здійснюється шляхом написання імперативних скриптів на мовах Shell або Python, які виконують команди операційної системи для налаштування прав доступу, політик паролів, відключення невикористовуваних сервісів та конфігурації міжмережевих екранів. Проте скриптовий підхід має суттєве обмеження – відсутність ідемпотентності, тобто здатності коректно виконуватися під час повторного запуску без порушення роботи системи. Тому більш ефективним рішенням є використання декларативних систем управління конфігураціями, зокрема Ansible, які дозволяють описувати цільовий стан системи (Desired State), а не алгоритм його досягнення [5].

Застосування Ansible для реалізації вимог НД ТЗІ дозволяє створити бібліотеку ролей безпеки (Security Roles), кожна з яких відповідає за виконання конкретної вимоги стандарту. Наприклад, вимога щодо обмеження кількості невдалих спроб входу в систему трансформується у програмний код (Playbook), який перевіряє налаштування модуля ПАМ (Pluggable Authentication Modules) в операційній системі Linux і, у разі невідповідності, автоматично вносить корективи. Важливою перевагою такого підходу є можливість версіонування конфігурації системи: код налаштувань зберігається у системі

контролю версій (Git), що дозволяє відстежувати всі зміни, проводити їх аудит (Code Review) та, за необхідності, миттєво повертати систему до попереднього безпечного стану. Це забезпечує повну прозорість процесу налаштування засобів захисту, що є критично важливим при проведенні державної експертизи КСЗІ, оскільки експерт може перевіряти не кожен окремий сервер, а вихідний код, який гарантовано застосовано до всієї інфраструктури [5].

Окрім того, використання систем управління конфігураціями дозволяє вирішити проблему гетерогенності середовища, забезпечуючи уніфіковане налаштування безпеки для різних операційних систем (Windows, Linux) та мережевого обладнання в рамках однієї КСЗІ. Розроблені сценарії автоматизації (Hardening Scripts) можуть включати сотні перевірок та налаштувань, таких як відключення підтримки застарілих протоколів шифрування, налаштування аудиту подій безпеки, контроль цілісності системних файлів та обмеження прав користувачів. Дослідження показують, що автоматизоване розгортання захищеного сервера займає лічені хвилини, тоді як ручне налаштування аналогічного рівня захищеності вимагає кількох годин роботи кваліфікованого фахівця [5]. Це не лише знижує вартість впровадження КСЗІ, але й мінімізує ризики, пов'язані з людським фактором, гарантуючи, що жодна вимога безпеки не буде пропущена або некоректно реалізована. Таким чином, перехід до автоматизованого налаштування засобів захисту є необхідним еволюційним кроком у розвитку методології технічного захисту інформації, що дозволяє забезпечити високий рівень кіберстійкості сучасних інформаційних систем.

Третій напрям автоматизації процесів побудови та супроводу КСЗІ пов'язаний зі зміною парадигми контролю захищеності: переходу від дискретних, періодичних перевірок до безперервного автоматизованого моніторингу стану інформаційної системи. Проведене дослідження проблематики експлуатації атестованих систем виявило критичний недолік традиційного підходу, який полягає у так званому «часовому розриві відповідності». Атестат відповідності КСЗІ видається строком до п'яти років, проте реальний стан налаштувань операційних систем, мережевого обладнання та прикладного програмного забезпечення змінюється щоденно внаслідок адміністративних дій, оновлень або інцидентів безпеки. Вже через короткий проміжок часу після отримання атестата система де-факто перестає відповідати вимогам нормативних документів (НД ТЗІ), хоча де-юре залишається захищеною. Для вирішення цієї проблеми було досліджено ефективність застосування спеціалізованих інструментальних засобів контролю, зокрема сканерів безпеки на базі протоколу SCAP, таких як OpenSCAP, та платформ розширеного виявлення загроз і реагування, таких як Wazuh [6].

В основі запропонованого технічного рішення лежить використання стандартизованих мов опису вразливостей та конфігурацій, зокрема OVAL (Open Vulnerability and Assessment Language) та XCCDF (The Extensible Configuration Checklist Description Format). Дослідження показало, що використання інструменту OpenSCAP дозволяє автоматизувати процес порівняння поточних налаштувань операційної системи з еталонним профілем безпеки (Security Profile), розробленим на основі вимог НД ТЗІ або міжнародних стандартів (CIS, DISA STIG). На відміну від ручного аудиту, який часто носить вибірковий характер через обмеженість людських ресурсів, автоматизований сканер здатний за лічені хвилини перевірити сотні параметрів на кожному вузлі мережі: від довжини пароля та прав доступу до критичних файлів до налаштувань ядра та служб аудиту. Результати експериментального впровадження OpenSCAP у тестовому середовищі продемонстрували можливість виявлення прихованих відхилень від політики безпеки (compliance drift), які неможливо помітити візуально, наприклад, зміну прав доступу до конфігураційних файлів веб-сервера або несанкціоноване включення служби віддаленого доступу [6].

Окрему увагу в роботі приділено аналізу можливостей платформи Wazuh, яка поєднує функції SIEM (Security Information and Event Management) та HIDS (Host-based Intrusion Detection System). Дослідження підтвердило, що агентна архітектура Wazuh забезпечує не лише періодичне сканування, але й моніторинг у режимі реального часу. Ключовою перевагою є модуль контролю цілісності файлів (FIM – File Integrity Monitoring), який дозволяє миттєво фіксувати будь-які зміни у файлах конфігурації КСЗІ або системних бінарних файлах. Якщо зловмисник або некомпетентний адміністратор змінить налаштування захисту, система автоматично згенерує інцидент безпеки із зазначенням точного часу, користувача та характеру змін. Це дозволяє реалізувати концепцію безперервної відповідності (Continuous Compliance), коли стан захищеності системи оцінюється не раз на рік комісією, а кожну секунду програмними засобами. Такий підхід повністю відповідає сучасним вимогам до захисту критичної інформаційної інфраструктури та дозволяє трансформувати КСЗІ з формальної паперової конструкції в дієвий механізм кіберзахисту [7].

Економічний та технічний аналіз впровадження автоматизованих засобів контролю показав значне скорочення витрат на проведення внутрішніх аудитів. Замість залучення групи експертів на кілька тижнів для ручної перевірки журналів та налаштувань, організація отримує актуальну картину стану безпеки на єдиній інформаційній панелі (Dashboard). Більше того, інтеграція сканерів безпеки з процесами DevOps дозволяє перевіряти відповідність системи вимогам КСЗІ ще на етапі розробки та тестування, блокуючи розгортання вразливих конфігурацій у продуктивному середовищі. Це створює замкнений цикл управління безпекою, де виявлення невідповідності автоматично ініціює процес її усунення. Таким чином, використання інструментів типу OpenSCAP та Wazuh є необхідною умовою для забезпечення реальної, а не декларативної захищеності сучасних інформаційних систем, дозволяючи підтримувати їх у стані постійної готовності до протидії кіберзагрозам та проходження зовнішніх перевірок контролюючими органами.

Висновки

Проведене дослідження засвідчило, що традиційна парадигма побудови КСЗІ, заснована на ручному створенні документації та ручному налаштуванні засобів захисту, поступово втрачає ефективність в умовах сучасних динамічних кіберзагроз та стрімкого масштабування інформаційних інфраструктур. Практична цінність дослідження полягає в обґрунтуванні комплексного підходу до автоматизації, який охоплює весь життєвий цикл системи захисту – від проектування до експлуатації. Доведено, що впровадження автоматизованої генерації організаційно-розпорядчої документації на основі параметричної моделі об'єкта дозволяє нівелювати розрив між формальними вимогами НД ТЗІ та реальною архітектурою системи, скорочуючи часові витрати на підготовку до державної експертизи на порядок та мінімізуючи семантичні помилки. Водночас застосування концепції «Інфраструктура як код» та систем управління конфігураціями забезпечує детермінованість стану захищеності інформаційної системи та гарантує ідентичність налаштувань безпеки на всіх її вузлах. Важливим елементом запропонованої методології є впровадження безперервного моніторингу відповідності із використанням протоколів автоматизації контенту безпеки. Це дозволяє трансформувати КСЗІ зі статичного стану, зафіксованого атестатом відповідності, у динамічний процес забезпечення кіберстійкості, здатний адаптуватися до змін ландшафту загроз у реальному часі. Таким чином, автоматизація процесів побудови КСЗІ є не лише технічною оптимізацією, а й необхідною умовою забезпечення реального, а не декларативного рівня захищеності державних інформаційних ресурсів. Її впровадження дозволяє суттєво зменшити витрати на експлуатацію системи, підвищити її надійність та забезпечити готовність до протидії сучасним кіберзагрозам.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Закон України «Про захист інформації в інформаційно-телекомунікаційних системах». URL: <https://zakon.rada.gov.ua/laws/show/80/94-%D0%B2%D1%80#Text> (дата звернення: 13.02.2026).
2. NIST (National Institute of Standards and Technology). URL: <https://www.nist.gov/> (дата звернення: 13.02.2026).
3. НД ТЗІ 2.5-004-99. URL: <https://tzi.com.ua/downloads/2.5-004-99.pdf> (дата звернення: 15.02.2026).
4. 2024 ETFA – IEEE 29th International Conference on Emerging Technologies and Factory Automation. URL: <https://bibliographie.ub.rub.de/work/396934> (дата звернення: 16.02.2026).
5. Morris, K. (2016). Infrastructure as Code: Managing Servers in the Cloud. O'Reilly Media. URL: <https://dl.ebooksworld.ir/books/Infrastructure.as.Code.2nd.Edition.Kief.Morris.OReilly.9781098114671.EBooksWorld.ir.pdf> (дата звернення: 19.02.2026).
6. Scarfone, K., & Souppaya, M. (2018). The Technical Specification for the Security Content Automation Protocol (SCAP). NIST Special Publication 800-126 Revision 3. URL: <https://csrc.nist.gov/pubs/sp/800/126/r3/final> (дата звернення: 19.02.2026).
7. Wazuh. URL: <https://documentation.wazuh.com/current/compliance/nist/configuration-assessment.html> (дата звернення: 20.02.2026).

Магденко Анастасія Романівна – студентка групи 1KITC-226, Факультет менеджменту та інформаційної безпеки, Вінницький національний технічний університет, м. Вінниця, e-mail: anastasiimahdenko@gmail.com

Науковий керівник: **Бондаренко Ірина Олексіївна** – асистент кафедри менеджменту та безпеки інформаційних систем, Вінницький національний технічний університет, м. Вінниця, e-mail: bondarenko.i@vntu.edu.ua

Mahdenko Anastasiia R. – student of group 1KITS-22b, Faculty of Management and Information Security, Vinnytsia National Technical University, Vinnytsia, e-mail: anastasiimahdenko@gmail.com

Supervisor: **Bondarenko Iryna O.** – assistant of the Department of Management and Security of Information Systems Vinnytsia National Technical University, Vinnytsia, e-mail: bondarenko.i@vntu.edu.ua