

SECURITY BACKLOG ЯК ІНСТРУМЕНТ УПРАВЛІННЯ КІБЕРРИЗИКАМИ В SCRUM-ПРОЄКТАХ

Вінницький національний технічний університет

Анотація

У роботі досліджується концепція Security Backlog як окремого механізму управління кіберризиками в межах Scrum-проектів. Обґрунтовується необхідність формалізації безпекових вимог у структурі гнучкої розробки програмного забезпечення, оскільки класичний фреймворк Scrum не передбачає спеціалізованих інструментів для системного управління кіберризиками. Проведено аналіз положень Scrum Guide, міжнародних стандартів безпечної розробки та сучасних практик DevSecOps. Запропоновано модель інтеграції Security Backlog у Scrum-процеси як окремого артефакту або розширення Product Backlog із власними правилами пріоритизації, оцінювання ризиків і контролю виконання. Показано, що впровадження Security Backlog дозволяє зменшити накопичення security debt, підвищити прозорість управління вразливостями та забезпечити відповідність сучасним вимогам кіберстійкості.

Ключові слова: Security Backlog; Scrum; кіберризики; DevSecOps; product backlog; безпечна розробка ПЗ.

Abstract

The article examines the concept of a Security Backlog as a dedicated mechanism for managing cyber risks within Scrum projects. It substantiates the need to formalize security requirements within the structure of agile software development, since the classical Scrum framework does not provide specialized instruments for systematic cyber risk management. The study analyzes the provisions of the Scrum Guide, international secure software development standards, and contemporary DevSecOps practices. A model for integrating the Security Backlog into Scrum processes is proposed, either as a separate artifact or as an extension of the Product Backlog with its own prioritization rules, risk assessment procedures, and execution control mechanisms. It is demonstrated that implementing a Security Backlog reduces the accumulation of security debt, enhances transparency in vulnerability management, and ensures compliance with modern cybersecurity resilience requirements.

Keywords: Security Backlog; Scrum; cyber risks; DevSecOps; product backlog; secure software development.

Вступ

Сучасні підходи до розробки програмного забезпечення дедалі більше ґрунтуються на гнучких методологіях, серед яких провідне місце займає Scrum. Згідно з офіційним Scrum Guide, Scrum визначає ролі, артефакти та події, однак не встановлює конкретних вимог щодо інтеграції безпекових процесів у межах спринтів [1]. Безпека розглядається як складова якості, але не як окремий механізм управління ризиками.

Водночас сучасні кіберзагрози вимагають системного підходу до безпеки на всіх етапах життєвого циклу ПЗ. Рамкова модель Secure Software Development Framework (SSDF), запропонована NIST, наголошує на необхідності інтеграції безпекових практик безпосередньо у процес розробки, а не лише на етапі тестування [2]. Аналогічні підходи простежуються у документації OWASP SAMM, де безпека трактується як безперервний процес управління ризиками [3].

У межах Scrum-проектів безпекові задачі зазвичай інтегруються у загальний Product Backlog без окремої категоризації. Такий підхід може призводити до їх систематичного відкладання через пріоритетність функціональних вимог. У зв'язку з цим постає проблема формалізації управління кіберризиками у межах Scrum.

Результати дослідження

Scrum працює поетапно: робота ділиться на короткі цикли (спринти), і на кожному з них команда поступово додає корисні результати [2]. Однак швидкі релізи можуть сприяти накопиченню технічного та безпекового боргу (security debt), якщо безпекові вимоги не мають чіткої пріоритизації.

Security debt визначається як накопичення невирішених безпекових проблем або компромісних рішень, що підвищують ризик майбутніх інцидентів. У Scrum-практиці такі завдання часто поступаються функціональним вимогам через бізнес-пріоритети.

Таким чином, відсутність структурованого механізму управління кіберризиками в Scrum створює організаційний дисбаланс між швидкістю розробки та рівнем безпеки.

Security Backlog пропонується як структурований набір завдань, що відображають:

- результати аналізу ризиків;
- виявлені вразливості;
- вимоги стандартів безпеки;
- рекомендації OWASP Top 10 [1].

На відміну від звичайних backlog-елементів, завдання Security Backlog мають супроводжуватися:

- оцінкою рівня ризику;
- визначенням потенційного впливу;
- обґрунтованою пріоритизацією.

NIST SSDF підкреслює необхідність інтеграції управління ризиками в процес розробки та документування безпекових вимог [3]. Впровадження Security Backlog дозволяє реалізувати ці вимоги в межах Scrum без зміни його базових принципів.

Запропоновано три можливі моделі:

1. Розширення Product Backlog – завдання безпеки маркуються окремими тегами та мають власну систему оцінювання ризиків.
2. Окремий Security Backlog – підтримується паралельно з Product Backlog, але інтегрується в планування спринтів.
3. Гібридна модель – критичні ризики інтегруються до Sprint Backlog негайно, менш критичні – накопичуються в Security Backlog.

OWASP SAMM рекомендує формалізувати процес управління завданнями безпеки в рамках організаційних практик [4]. Запропонована модель відповідає цим рекомендаціям.

Впровадження Security Backlog трансформує підхід до безпеки з епізодичних перевірок на безперервний керований процес. Замість того, щоб сприймати вразливості як раптові перешкоди, команда отримує можливість системно зменшувати «security debt» у сфері безпеки шляхом їхньої чіткої фіксації та класифікації. Це перетворює абстрактні загрози на конкретні робочі завдання, інтегровані в загальний життєвий цикл продукту.

Такий підхід забезпечує високий рівень прозорості ризиків для Product Owner-ів та ключових стейкхолдерів. Коли безпекові вимоги стають частиною загального беклогу, бізнес може приймати зважені рішення щодо пріоритетів, поєднуючи нові можливості та захист системи. Крім того, наявність структурованого переліку завдань гарантує відповідність міжнародним стандартам безпечної розробки, що є критичним для сертифікації та довіри клієнтів.

Практична користь Security Backlog полягає в тому, що він підвищує безпеку продукту: регулярне додавання завдань з безпеки у спринти допомагає уникнути накопичення серйозних вразливостей, які могли б спричинити великі проблеми. В результаті DevSecOps стає постійною практикою в компанії. Безпека перестає бути зоною відповідальності окремого підрозділу і стає невід'ємною складовою кожної фази розробки, перетворюючи Backlog з простого інструменту обліку на потужний механізм стратегічного управління кіберризиками.

Висновки

Отже, впровадження Security Backlog у Scrum-проектах є ефективним інструментом системного управління кіберризиками в умовах гнучкої розробки. Формалізація безпекових завдань, їх ризик-орієнтована пріоритизація та інтеграція в процес спринтів дозволяють мінімізувати security debt, підвищити прозорість управління вразливостями та забезпечити відповідність міжнародним стандартам безпечної розробки. Запропонована модель може стати основою подальших досліджень у напрямі поєднання Agile-методологій та управління кібербезпекою.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. OWASP Top Ten Web Application Security Risks | OWASP Foundation. OWASP Foundation, the Open Source Foundation for Application Security | OWASP Foundation. [Електронний ресурс]. Режим доступу: <https://owasp.org/www-project-top-ten/> (дата звернення: 01.03.2026).
2. Home | Scrum Guides. [Електронний ресурс]. Режим доступу: <https://scrumguides.org/> (дата звернення: 01.03.2026).
3. Secure Software Development Framework | CSRC. NIST Computer Security Resource Center | CSRC. [Електронний ресурс]. Режим доступу: <https://csrc.nist.gov/Projects/ssdf> (дата звернення: 02.03.2026).
4. OWASP SAMM | OWASP Foundation. OWASP Foundation, the Open Source Foundation for Application Security | OWASP Foundation. [Електронний ресурс]. Режим доступу: <https://owasp.org/www-project-samm/> (дата звернення: 02.03.2026).

Марущак Анастасія Віталіївна – студентка групи КІТС-25м, Факультет менеджменту та інформаційної безпеки, Вінницький національний технічний університет, м. Вінниця, e-mail: anamar349@gmail.com

Науковий керівник: **Карпинець Василь Васильович** – кандидат технічних наук, доцент, завідувач кафедри менеджменту та інформаційної безпеки, Вінницький національний технічний університет, м. Вінниця, e-mail: karpinets@gmail.com

Marushchak Anastasiia V. – student of group CITS-25m, Faculty of Management and Information Security, Vinnytsia National Technical University, Vinnytsia, e-mail: anamar349@gmail.com

Supervisor: **Karpinets Vasyl V.** – Ph. D., assistant professor, Head of the Department of Management and Security of Information Systems, Vinnitsa National Technical University, Vinnytsia, e-mail: karpinets@gmail.com