МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Львівський національний університет імені Івана Франка
Факультет прикладної математики та інформатики
Кафедра кібербезпеки

Затверджено
На засіданні кафедри кібербезпеки
факультету прикладної математики та інформатики
Львівського національного університету імені Івана Франка
(Протокол № 15/23 від 29 серпня 2023 р.)

Завідувач кафедри П.С.Венгерський

Силабус з навчальної дисципліни "Безпечне програмування", що викладається в межах ОПП Кібербезпека першого (бакалаврського) рівня вищої освіти для здобувачів з спеціальності 125 – кібербезпека та захист інформації

Львів 2023 р.

Назва дисципліни	**Безпечне програмування**
Адреса викладання дисципліни	Львівський національний університет імені Івана Франка, вул. Університетська 1, м. Львів, Україна, 79000
Факультет та кафедра, за якою закріплена дисципліна	Факультет прикладної математики та інформатики Кафедра кібербезпеки
Галузь знань, шифр та назва спеціальності	12 – інформаційні технології 125 – кібербезпека та захист інформації
Викладачі дисципліни	Хохлачова Юлія Євгеніївна, доцент кафедри кібербезпеки (лекції та лабораторні заняття)
Контактна інформація викладачів	valeriy.trushevsky@lnu.edu.ua https://ami.lnu.edu.ua/employee/trushevskyj-v-m Головний корпус ЛНУ ім. І. Франка, каб. 380. м. Львів, вул. Університетська, 1
Консультації з питань навчання по дисципліні відбуваються	Консультації проводять раз на тиждень згідно з оприлюдненим розкладом консультацій викладача. Можливі онлайн консультації через Zoom чи Microsoft Teams. Для погодження часу онлайн консультацій слід писати на електронну пошту викладача.
Сторінка курсу	https://ami.lnu.edu.ua/course/bezpechne-prohramuvannia-kb
Інформація про дисципліну	Дисципліна “Безпечне програмування” є вибірковою дисципліною з спеціальності 125 – кібербезпека та захист інформації для освітньої програми Кібербезпека, яка викладається у 8-му семестрі в обсязі 6-ти кредитів (за Європейською Кредитно-Трансферною Системою ECTS).
Коротка анотація дисципліни	Курс спрямований на формування у студентів професійних компетнтностей у галузі безпечного програмування, вивчення теоретичних та практичних основ щодо захисту програмного забезпечення від вразливостей, враховуючи повний цикл його існування. Розглядаються основні принципи проектування та розробки безпечного коду, стратегії уникнення вразливостей, технології тестування програмного забезпечення, реалізація заходів захисту програмного забезпечення, OWASP.
Мета та цілі дисципліни	Метою курсу є вивчення базових принципів та методів захисту програмного забезпечення від вразливостей, основ проектування та розробки безпечного коду, стратегій уникнення вразливостей, вміння застосовувати основні принципи безпечного програмування OWASP на практиці.
Література для вивчення дисципліни	*Основна* 1. Brown B. Cyber Security Program and Policy Using NIST Cybersecurity Framework, 2023. – 169 p. 2. Kohnfelder K. Designing Secure Software, 2021. – 312 p. 3. Raje G. Security and Microservice Architecture on AWS: Architecting and Implementing a Secured, Scalable Solution 1st Edition, 2021. – 394 p. 4. Blokdyk G. Secure Coding Best Practices A Complete Guide – 2021. – 309 p. 5. Wilson G.DevSecOps: A leader’s guide to producing secure software without compromising flow, feedback and continuous improvement, 2020. – 278 p. 6. Magnusson A. Practical Vulnerability Management: A Strategic Approach to Managing Cyber Risk, 2020. – 192 p.

	7. Janca T. Alice and Bob Learn Application Security, 2020. – 288 p. *Додаткова* 8. Deogun D. Secure By Design First Edition, 2019. – 410 p. 9. Martin R. Clean Architecture: A Craftsman's Guide to Software Structure and Design, 2017. – 432 p. 10. Robert C. Seacord, Secure coding in C and C++ - 2nd ed, 2013. – 569 p. 11. Howard M., LeBlanc D. Writing Secure Code - 2nd ed, 2015. – 768 p. 12. Fundamental Practices for Secure Software Development, 2018. – 38p. *Рекомендовані онлайн ресурси* 1. Udemy: Principles of Secure Coding 2. Udemy: Secure Coding - Secure application development 3. Udemy: Secure Coding in C/C++ 4. Udemy: Secure Coding & Design Best Practices in Python 5. Udemy: Secure Coding and Design Best Practices in C# 6. Udemy: Software Architecture Security - The Complete Guide 7. Coursera: Principles of Secure Coding 8. OWASP Foundation Developer Guide project 9. OWASP Application Security Verification Standard 10. SAFECode
Обсяг курсу	Загальний обсяг: 180 годин. Аудиторних занять: 70 год., з них 28 год. лекцій та 42 год. лабораторних робіт. Самостійної роботи: 110 год.
Очікувані результати навчання	У результаті вивчення навчальної дисципліни студент має набути таких компетентностей: **знати:** - вимоги до безпеки програмного забезпечення; - основні стратегії уникнення вразливостей; - основні принципи проектування безпечного коду; - типові помилки програмування та методи їх усунення; - типові моделі атак; - рекомендації щодо запобігання вразливостей; - методи захисту даних користувача; - концепції статичного та динамічного аналізу коду; - засоби автоматизованого тестування; - основні практики OWASP; - безпечна обробка помилок та безпечне логування; - тестування безпеки. **вміти:** - застосовувати принципи проектування та розробки безпечного коду; - виявляти ризики безпеки у коді та зовнішніх залежностях; - застосовувати шаблони атак для виявлення вразливостей; - визначати заходи протидії загрозам; - запобігати вразливостям конфедиційності; - проводити code review для виявлення потенційних вразливостей системи; - виявляти проблеми в коді використовуючи PyLint Tool; - застосовувати основні принципи OWASP на практиці. **Курс забезпечує набуття таких компетентностей: КЗ 1, КЗ 2, КЗ 3, КЗ 4, КЗ 5, КЗ 7, КФ 1, КФ 5, КФ 7, КФ 9, КФ 10; та програмних результатів**

	навчання: ПРН 1, ПРН 2, ПРН 3, ПРН 4, ПРН 5, ПРН 6, ПРН 7, ПРН 8, ПРН 9, ПРН 16, ПРН 22, ПРН 27, ПРН 31, ПРН 33, ПРН 34, ПРН 36, ПРН 37, ПРН 38, ПРН 40, ПРН 44, ПРН 45, ПРН 46, ПРН 47, ПРН 48.
Ключові слова	Безпека програмного забезпечення, безпечний код, моделі атак, вразливості, ризики, статичні та динамічні аналізатори коду, загрози, переповнення буферу, безпечна обробка помилок, безпечне логування, PyLint Tool, Code review, SQL-ін'єкції, OWASP challenges.
Формат курсу	Очний
Теми	Теми подані у схемі курсу нижче
Підсумковий контроль, форма	Залік у кінці семестру.
Пререквізити	Для вивчення курсу студенти потребують базові знання з таких дисциплін: 1) Програмування; 2) Основи кібербезпеки; 3) Основи криптології; 4) Прикладна криптологія.
Навчальні методи та техніки, які будуть використовуватися під час викладання курсу	Презентації, лекції Модульний контроль Індивідуальні завдання RangeForce платформа
Необхідне обладнання	Лабораторія з обладнаними робочими станціями. IDE для програмування мовою C++, C#, Python або Java.
Критерії оцінювання (окремо для кожного виду навчальної діяльності)	Оцінювання проводиться за 100-бальною шкалою. Бали нараховуються за наступним співвідношенням: • модульний контроль, тестування, усне опитування: 50% семестрової оцінки; максимальна кількість балів 50 • лабораторні роботи: 50% семестрової оцінки; максимальна кількість балів 50 Підсумкова максимальна кількість балів 100. **Академічна доброчесність**: Очікується, що роботи студентів будуть їх оригінальними дослідженнями чи міркуваннями. Відсутність посилань на використані джерела, фабрикування джерел, списування, втручання в роботу інших студентів становлять, але не обмежують, приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності в письмовій роботі студента є підставою для її незарахуванння викладачем, незалежно від масштабів плагіату чи обману. **Відвідання занять** є важливою складовою навчання. Очікується, що всі студенти відвідають усі лекції та практичні зайняття курсу. Студенти повинні інформувати викладача про неможливість відвідати заняття. У будь-якому випадку студенти зобов'язані дотримуватися термінів визначених для виконання всіх видів письмових робіт та індивідуальних завдань, передбачених курсом. **Література.** Уся література, яку студенти не зможуть знайти самостійно, буде надана викладачем виключно в освітніх цілях без права її передачі третім особам. Студенти заохочуються до використання також й іншої літератури та джерел, яких немає серед рекомендованих. **Політика виставлення балів.** Враховуються бали набрані при поточному тестуванні, самостійній роботі та бали підсумкового тестування. При цьому обов'язково враховуються присутність на заняттях та активність студента під час практичного заняття; недопустимість пропусків та запізнень на заняття; користування мобільним телефоном, планшетом чи іншими мобільними

	пристроями під час заняття в цілях не пов'язаних з навчанням; списування та плагіат; несвоєчасне виконання поставленого завдання і т. ін. Жодні форми порушення академічної доброчесності не толеруються.
Питання до контролю	1. Основні елементи безпеки та вразливості програмного забезпечення. 2. Типові моделі атак програмного забезпечення. Пошук вразливостей. 3. Виправлення вразливостей хешу пароля. Злам хешу пароля та захист від цього. 4. Основні стратегії уникнення вразливостей. Шаблони атак для виявлення вразливостей. Вразливості спричинені людським фактором. 5. Оцінка функціональності програмного забезпечення. Виявлення ризиків безпеки у коді та зовнішніх залежностях. 6. Основні принципи проектування безпечного коду. Керування ризиками. 7. Моделювання та ранжування загроз. Визначення заходів протидії загрозам. 8. Використання OWASP Threat Dragon для моделювання загроз, Microsoft Threat Modeling Tool. 9. Типові помилки програмування та методи їх усунення. Переповнення буферу. Рекомендації щодо запобігання вразливостей. 10. Виявлення поширених веб-вразливостей. Запобігання вразливостям конфедиційності. 11. Безпечне керування інтернет сесіями та розподіленим доступом користувачів. 12. Методи захисту даних користувача. Безпечна обробка помилок та безпечне логування. 13. Тестування безпеки. Code review для виявлення потенційних вразливостей системи. Концепції статичного та динамічного аналізу коду. 14. Виявлення проблем в коді використовуючи PyLint Tool, засоби автоматизованого тестування. Використання OWASP Zed Attack Proxy (ZAP). 15. Моніторинг та логування для підтримки безпеки. 16. OWASP: Порушення контролю доступу. 17. OWASP: Криптографічні вразливості. 18. OWASP: Вразливості ін'єкцій. 19. OWASP: Вразливий дизайн. 20. OWASP: Вразлива конфігурація веб-аплікації. 21. OWASP: Вразливі та застарілі компоненти. 22. OWASP: Помилки ідентифікації та автентифікації. 23. OWASP: Порушення цілісності програмного забезпечення та даних. 24. OWASP: Помилки логування та моніторингу безпеки. Основні види атак та загроз цифрового підпису. 25. OWASP: Підробка запитів сервера.
Опитування	Анкету-оцінку з метою оцінювання якості курсу буде надано по завершенню курсу.

Тиж.	Тема, план, короткі тези	Форма діяльності (заняття)	Література	Завдання, год.	Термін виконання
1	**Тема 1. Вимоги до безпеки програмного забезпечення**. (Термінологія та основні елементи безпеки програмного забезпечення. Вразливості програмного забезпечення. Типові моделі атак. Виправлення вразливостей хешу пароля.)	лекція, самостійна робота	[1-7]	2 6	1 тиждень

	Тема 1. Вимоги до безпеки програмного забезпечення. (Визначення основних вимог безпеки програмного забезпечення. Пошук вразливостей. Злам хешу пароля та захист від цього)	лаб.	[1-7]	2	
2	**Тема 2. Стратегії уникнення вразливостей.** (Основні стратегії уникнення вразливостей. Оцінка функціональності програмного забезпечення. Виявлення ризиків безпеки у коді та зовнішніх залежностях.)	лекція, самостійна робота	[1-7]	2 8	1 тиждень
	Тема 2. Стратегії уникнення вразливостей. (Шаблони атак для виявлення вразливостей. Вразливості спричинені людським фактором)	лаб.	[1-7]	4	
3	**Тема 3. Проектування безпечного коду.** (Основні принципи проектування безпечного коду. Моделювання та ранжування загроз. Керування ризиками.)	лекція, самостійна робота	[1-7]	2 8	1 тиждень
	Тема 3. Проектування безпечного коду. (Визначення заходів протидії загрозам. Використання OWASP Threat Dragon для моделювання загроз, Microsoft Threat Modeling Tool)	лаб.		2	
4	**Тема 4. Розробка безпечного коду.** (Типові помилки програмування та методи їх усунення. Переповнення буферу. Рекомендації щодо запобігання вразливостей.)	лекція, самостійна робота	[1-7]	2 8	1 тиждень
	Тема 4. Розробка безпечного коду. (Виявлення поширених веб-вразливостей. Запобігання вразливостям конфедиційності.)	лаб.	[1-7]	4	
5	**Тема 5. Реалізація заходів захисту програмного забезпечення.** (Безпечне керування інтернет сесіями та розподіленим доступом користувачів)	лекція, самостійна робота	[1-7]	2 8	1 тиждень
	Тема 5. Реалізація заходів захисту програмного забезпечення. (Методи захисту даних користувача. Безпечна обробка помилок та безпечне логування.)	лаб.	[1-7]	2	
6	**Тема 6. Тестування безпеки програмного забезпечення.** (Тестування безпеки. Code review для виявлення потенційних вразливостей системи. Концепції статичного та динамічного аналізу коду.)	лекція, самостійна робота	[1-7]	2 8	1 тиждень
	Тема 6. Тестування безпеки програмного забезпечення. (Виявлення проблем в коді використовуючи PyLint Tool, засоби автоматизованого тестування. Використання OWASP Zed Attack Proxy (ZAP))	лаб.	[1-7]	4	
7	**Тема 7. Підтримка безпеки програмного забезпечення.** (Моніторинг та логування для підтримки безпеки.)	лекція, самостійна робота	[1-7]	2 8	1 тиждень
	Тема 7. Підтримка безпеки програмного забезпечення.	лаб.	[1-7]	2	

	(Реалізація логування. Підтримка безпеки після деплоймента.)				
8	**Тема 8. OWASP: Порушення контролю доступу.** (Вразливість порушення контролю доступу.)	лекція, самостійна робота	[1-7]	2 8	1 тиждень
	Тема 8. OWASP: Порушення контролю доступу. **(**Direct Request OWASP Challenge, Sensitive File in Web Root OWASP Challenge**)**	лаб.	[1-7]	4	
9	**Тема 9. OWASP: Криптографічні вразливості.** (Задача зламу слабкої хеш-функції.) **Тема 10. OWASP: Вразливості ін'єкцій.** (Вразливість ін'єкцій команд та шаблонів)	лекція, самостійна робота	[1-7]	2 8	1 тиждень
	Тема 9. OWASP: Криптографічні вразливості. (Злам системи за допомогою криптографічних ключів в коді.) **Тема 10. OWASP: Вразливості ін'єкцій.** (Вразливість SQL-ін'єкцій)	лаб.	[1-7]	2	
10	**Тема 11. OWASP: Вразливий дизайн.** (Вразливості небезпечного дизайну. Unrestricted File Upload OWASP Challenge) **Тема 12. OWASP: Вразлива конфігурація веб-аплікації.** (Directory Listing OWASP Challenge)	лекція, самостійна робота	[1-7]	2 8	1 тиждень
	Тема 11. OWASP: Вразливий дизайн. (HTTP Request Smuggling OWASP Challenge) **Тема 12. OWASP: Вразлива конфігурація веб-аплікації.** (XML External Entities OWASP Challenge)	лаб.	[1-7]	4	1 тиждень
11	**Тема 13. OWASP: Вразливі та застарілі компоненти.** (Vulnerable Component OWASP Challenge, Backdoored Component OWASP Challenge)	лекція, самостійна робота	[1-7]	2 8	1 тиждень
	Тема 13. OWASP: Вразливі та застарілі компоненти. (Log4Shell OWASP Challenge)	лаб.	[1-7]	2	
12	**Тема 14. OWASP: Помилки ідентифікації та автентифікації.** (Missing Authentication for Critical Function OWASP Challenge, Brute Forcing OWASP Challenge)	лекція, самостійна робота	[1-7]	2 8	1 тиждень
	Тема 14. OWASP: Помилки ідентифікації та автентифікації. (Session Fixation OWASP Challenge)	лаб.	[1-7]	4	1 тиждень

13	**Тема 15. OWASP: Порушення цілісності програмного забезпечення та даних.** (Remote File Inclusion OWASP Challenge)	лекція, самостійна робота	[1-7]	2 8	1 тиждень
	Тема 15. OWASP: Порушення цілісності програмного забезпечення та даних. (Code Download Without Integrity Check OWASP Challenge)	лаб.	[1-7]	2	
14	**Тема 16. OWASP: Помилки логування та моніторингу безпеки. (**Sensitive Information in Log File OWASP Challenge, **)**	лекція, самостійна робота	[1-7]	2 8	1 тиждень
	Тема 17. OWASP: Підробка запитів сервера. (Server-Side Request Forgery OWASP Challenge, AWS Metadata SSRF OWASP Challenge, SSRF Insecure Allowlist Bypass OWASP Challenge**)**	лаб.	[1-7]	4	1 тиждень