Робоча програма дисципліни «WEB-технології»

СПЕЦІАЛЬНІСТЬ 123 · КОМП'ЮТЕРНА ІНЖЕНЕРІЯ

БАКАЛАВР

120

Годин

Загальний обсяг дисципліни

4

Кредити ЄКТС

Відповідно до стандарту

4

Модулі

Змістовних блоки курсу

Загальна інформація про дисципліну

Ідентифікація

Назва: WEB-технології

Спеціальність: 123 Комп'ютерна інженерія

Освітня програма: Організація захисту інформації в інформаційно-комунікаційних системах

Ступінь: Бакалавр

Обсяг: 120 годин / 4 кредити ЄКТС

Модулів: 4 змістовних блоки

Призначення дисципліни

Дисципліна «WEB-технології» формує у студентів системне розуміння принципів побудови та функціонування сучасних веб-додатків, клієнт-серверної архітектури, протоколів передачі даних та технологій захисту інформації у веб-середовищі. Курс охоплює ключові технології розробки веб-застосунків — від розмітки та стилізації до серверної обробки та безпеки.

Дисципліна є обов'язковою для підготовки бакалаврів зі спеціальності «Комп'ютерна інженерія» за освітньою програмою «Організація захисту інформації в інформаційно-комунікаційних системах» та забезпечує формування фахових компетентностей у сфері веб-розробки та захисту веб-застосунків.

Загальні компетентності

Аналітичне мислення

Здатність до аналізу, синтезу та критичного оцінювання технічної інформації: виявлення вразливостей у веб-додатках, порівняння архітектурних рішень, формування обґрунтованих висновків щодо вибору технологічного стеку.

Комунікація

Здатність до усної та письмової комунікації у технічному середовищі: чітко описувати архітектурні рішення, документувати код, складати технічні специфікації та звіти про виявлені вразливості.

Самоорганізація

Уміння самостійно навчатися, планувати роботу та організовувати виконання проектних завдань у визначені строки, зокрема під час розробки веб-застосунків та підготовки аналітичних звітів.

Командна робота

Відповідальність, ініціативність та готовність працювати в команді під час розробки командних проектів, розподілу задач між фронтенд- та бекенд-розробниками, спільного тестування та аудиту безпеки.

Студент повинен вміти: аналізувати технічну документацію та специфікації; виділяти ключові вимоги до веб-застосунків; планувати власну діяльність і дотримуватися дедлайнів; коректно взаємодіяти з викладачами та одногрупниками; брати участь у командній розробці та розподіляти обов'язки між учасниками проекту.

Фахові компетентності

Розробка веб-інтерфейсів

Розуміння сутності та закономірностей побудови клієнтської частини веб-додатків. Знання HTML5, CSS3 та JavaScript для розробки адаптивних, семантичних та доступних веб-інтерфейсів, що відповідають сучасним стандартам W3C.

Серверні технології та API

Здатність розробляти серверну логіку з використанням сучасних фреймворків, проектувати та реалізовувати RESTful API, працювати з базами даних, організовувати автентифікацію та авторизацію у веб-застосунках.

Безпека веб-додатків

Знання основних вразливостей веб-застосунків (OWASP Top 10), методів їх виявлення та усунення. Розуміння принципів захисту від XSS, CSRF, SQL-ін'єкцій, реалізації HTTPS, управління сесіями та контролю доступу.

Протоколи та архітектура

Орієнтування в протоколах передачі даних (HTTP/HTTPS, WebSocket), архітектурних патернах (MVC, SPA, PWA), принципах хмарного розгортання веб-застосунків та використання CDN для забезпечення доступності та безпеки.

Результати навчання

01

Пояснення веб-технологій

Пояснювати сутність ключових понять і явищ: клієнт-серверна архітектура, HTTP-протокол, DOM-модель, RESTful API, автентифікація, авторизація, сесія, cookie, токен, CORS, HTTPS, TLS/SSL.

02

Розробка веб-застосунків

Розробляти повноцінні веб-застосунки з використанням сучасних технологій клієнтської та серверної частини, застосовувати фреймворки та бібліотеки, організовувати взаємодію з базами даних та зовнішніми API.

03

Аналіз та усунення вразливостей

Аналізувати веб-застосунки на наявність вразливостей відповідно до OWASP Top 10; виявляти слабкі місця в архітектурі захисту, пропонувати та реалізовувати заходи з підвищення рівня безпеки.

04

Робота з документацією та стандартами

Орієнтуватися в технічній документації, специфікаціях W3C, RFC та стандартах безпеки (OWASP, ISO/IEC 27001); використовувати професійні ресурси для прийняття обґрунтованих технічних рішень.

МОДУЛЬ 1

Основи веб-розробки: HTML5, CSS3 та клієнтська частина

1 — Тема 1.1

Архітектура веб-застосунків. Протокол HTTP/HTTPS. Структура веб-документа. Стандарти W3C.

2 — Тема 1.2

HTML5: семантична розмітка, форми, мультимедіа, доступність (WCAG). Валідація та SEO-оптимізація.

3 — Тема 1.3

CSS3: каскадність, специфічність, Flexbox, Grid Layout. Адаптивний дизайн та медіа-запити.

4 — Тема 1.4

JavaScript: основи мови, DOM-маніпуляції, події, асинхронність (Promise, async/await), Fetch API.

МОДУЛЬ 1 · ТЕМА 1.1

Архітектура веб–застосунків та протокол HTTP/HTTPS

Лекційний матеріал (2 год.)

Лекція 1: Еволюція веб-технологій. Клієнт-серверна архітектура. Модель OSI та стек TCP/IP. Протокол HTTP: методи (GET, POST, PUT, DELETE), статус-коди, заголовки запиту та відповіді. HTTPS: принципи роботи TLS/SSL, сертифікати, центри сертифікації (CA).

Лекція 2: Браузер як середовище виконання: рушії рендерингу, JavaScript-рушії, DevTools. DNS, IP-адресація, хостинг та доменні імена. Стандарти W3C та WHATWG. Інструменти розробника.

Ключові питання лекції

- Відмінності між HTTP та HTTPS у контексті захисту інформації
- Методи HTTP та їх семантика в контексті безпеки
- Роль сертифікатів TLS/SSL у захисті веб-комунікацій
- Принципи роботи браузера та рушія рендерингу
- Стандарти W3C та їх вплив на безпеку веб-застосунків

Практичне заняття (1 год.)

Завдання 1. Аналіз HTTP-запитів та відповідей за допомогою браузерних DevTools та Wireshark: дослідження заголовків, статус-кодів, cookie та механізмів кешування.

Завдання 2. Порівняльний аналіз HTTP та HTTPS-трафіку: виявлення відмінностей у захисті переданих даних.

Лабораторна робота (1 год.)

Лаб. 1. Налаштування локального веб-сервера (Apache/Nginx). Генерація самопідписаного TLS-сертифіката. Аналіз трафіку інструментами браузера.

Самостійна робота (8 год.)

- Опрацювання: Олексій Васильченко, Андрій Кравченко «Веб-технології та веб-дизайн» — Київ: КПІ ім. Ігоря Сікорського, 2023
- Дослідження специфікацій RFC 7230–7235 (HTTP/1.1) та RFC 9110 (HTTP semantics)
- Підготовка реферату: «Еволюція протоколу HTTP від 1.0 до HTTP/3»

Очікуваний результат: студент розуміє архітектуру веб-застосунків, принципи роботи HTTP/HTTPS та вміє аналізувати мережевий трафік.

МОДУЛЬ 1 · ТЕМА 1.2

HTML5: семантична розмітка, форми та доступність

1

Семантика

Теги article, section, nav, header, footer, main. Значення для SEO та доступності.

2

Форми

Нові типи input, атрибути валідації, обробка даних форм. Захист від CSRF.

3

Мультимедіа

Теги audio, video, canvas, SVG. Вбудований контент та його безпека.

4

Доступність

Стандарт WCAG 2.1, ARIA-атрибути. Тестування доступності веб-сторінок.

Лекційний матеріал (2 год.)

Структура HTML5-документа. Семантичні теги та їх роль у SEO та доступності. Форми: нові типи полів, атрибути валідації, безпечна обробка даних. Мультимедіа: теги audio, video, canvas. Вбудований контент (iframe) та ризики безпеки (clickjacking). WCAG 2.1 — стандарт доступності. ARIA-ролі та атрибути. Валідація HTML-коду (W3C Validator).

Практика (1 год.) та Лабораторна (1 год.) і СРС (8 год.)

Практика: Семантична верстка веб-сторінки; аналіз HTML-коду на валідність та доступність за допомогою W3C Validator та Lighthouse.

Лабораторна (Лаб. 2): Розробка семантичної HTML5-сторінки з формами та перевіркою введення. Реалізація захисту від XSS у клієнтській частині.

СРС: Опрацювання: Гуржій А. М., Поворознюк Н. І., Самсонов В. В. «Основи інформаційних технологій» (Київ: Літера ЛТД, 2023); підготовка аналітичної записки «Безпека HTML-форм: типові вразливості та методи захисту».

МОДУЛЬ 1 · ТЕМИ 1.3–1.4

CSS3 та JavaScript: стилізація, інтерактивність та асинхронність

Тема 1.3 — CSS3 (Лекція 5, 2 год.)

Каскадність та специфічність CSS. Блокова модель (box model). Flexbox та Grid Layout — сучасні підходи до розташування елементів. Адаптивний дизайн: медіа-запити, responsive breakpoints, mobile-first підхід. CSS-змінні (custom properties). Препроцесори (Sass/SCSS). Анімації та трансформації. Продуктивність CSS: критичний шлях рендерингу.

Практика (1 год.): Розробка адаптивного макету за допомогою Flexbox та Grid; аналіз продуктивності CSS за допомогою DevTools.

Лабораторна (Лаб. 3, 1 год.): Розробка адаптивного веб-інтерфейсу з використанням CSS Grid та Flexbox. Реалізація темної/світлої теми через CSS-змінні.

СРС (8 год.): Опрацювання: Олексій Васильченко, Андрій Кравченко «Веб-технології та веб-дизайн» (КПІ, 2023) — розділ «CSS3»; підготовка порівняльної таблиці «CSS Flexbox vs Grid: сфери застосування».

Тема 1.4 — JavaScript (Лекція 6–8, 6 год.)

Основи JavaScript ES6+: типи даних, змінні (let, const, var), функції, стрілкові функції, деструктуризація, spread/rest оператори, модулі (ESM). DOM API: вибір елементів, маніпуляції, обробка подій, делегування подій. Асинхронність: callback, Promise, async/await. Fetch API та робота з REST API: формати JSON/XML, обробка помилок. Зберігання даних у браузері: localStorage, sessionStorage, IndexedDB, cookie та їх безпека. Захист від XSS у JavaScript.

Практика (1 год.): Розробка інтерактивного компонента за допомогою DOM API; реалізація асинхронного запиту до публічного API (fetch).

Лабораторна (Лаб. 4, 1 год.): Розробка SPA-компонента з асинхронним отриманням даних через Fetch API. Реалізація захисту від XSS: санітизація введених даних, CSP-заголовки.

СРС (8 год.): Опрацювання документації MDN Web Docs; підготовка аналітичної записки «Вразливості JavaScript-застосунків: XSS, CSRF та методи їх усунення».

МОДУЛЬ 2

Серверні технології та розробка веб-застосунків

Тема 2.1

Серверні мови та фреймворки. Node.js, PHP, Python (Flask/Django). MVC-архітектура веб-застосунків.

Тема 2.2

Бази даних у веб-застосунках. SQL та NoSQL. ORM. Захист від SQL-ін'єкцій та безпечна робота з даними.

Тема 2.3

RESTful API: проектування, документування (OpenAPI/Swagger), версіонування. GraphQL. Автентифікація API.

Тема 2.4

Автентифікація та авторизація: сесії, JWT, OAuth 2.0, OpenID Connect. Управління доступом у веб-застосунках.

МОДУЛЬ 2 · ТЕМА 2.1

Серверні технології та MVC-архітектура

Лекційний матеріал (2 год.)

Серверні мови програмування для веб: PHP, Python, JavaScript (Node.js). Порівняльна характеристика технологічних стеків (LAMP, MEAN, MERN). MVC-архітектура (Model-View-Controller): принципи розподілу відповідальностей. Фреймворки: Express.js (Node.js), Django/Flask (Python), Laravel (PHP). Шаблонізатори (Jinja2, EJS, Blade). Middleware — проміжне програмне забезпечення та його роль у безпеці.

Ключові питання

- Переваги та недоліки різних серверних стеків
- Принципи MVC та їх вплив на безпеку застосунку
- Роль middleware у фільтрації запитів та захисті
- Налаштування безпечного веб-сервера (заголовки безпеки)
- Журналювання та моніторинг серверних подій

Практичне заняття (1 год.)

Завдання 1. Розгортання веб-застосунку на основі MVC-фреймворку. Налаштування безпечних HTTP-заголовків (HSTS, CSP, X-Frame-Options, X-Content-Type-Options).

Завдання 2. Аналіз middleware: реалізація логування запитів, обмеження швидкості (rate limiting) та захист від DDoS на рівні застосунку.

Лабораторна робота (Лаб. 5, 1 год.)

Розробка базового веб-застосунку за шаблоном MVC (Node.js + Express або Python + Flask). Налаштування заголовків безпеки. Реалізація логування подій.

Самостійна робота (8 год.)

- Опрацювання: Яременко В. О., Сердюк Б. М. «Серверні веб-технології та захист інформації» (Харків: ХНУРЕ, 2023)
- Підготовка реферату: «Порівняльний аналіз веб-фреймворків: безпека та продуктивність»

Очікуваний результат: студент вміє розробляти серверну частину веб-застосунку на базі MVC-фреймворку та налаштовувати базові заходи безпеки.

МОДУЛЬ 2 · ТЕМА 2.2

Бази даних у веб-застосунках та захист від SQL-ін'єкцій

Реляційні БД

MySQL, PostgreSQL. Структура таблиць, зв'язки, транзакції. Підготовлені запити.

ORM

Sequelize, Mongoose, SQLAlchemy. Абстракція даних та зниження ризику ін'єкцій.

Безпека даних

Шифрування в БД, хешування паролів (bcrypt, Argon2). Резервне копіювання.

NoSQL БД

MongoDB, Redis. Документно-орієнтована модель. Кешування. Сесії у Redis.

SQL-ін'єкції

Типи атак: blind, error-based, UNION-based. Методи виявлення та захисту.

Лекційний матеріал (2 год.)

Реляційні СУБД (MySQL, PostgreSQL) та NoSQL (MongoDB, Redis) у контексті веб-застосунків. ORM-бібліотеки та їх переваги для безпеки. SQL-ін'єкції: класифікація, приклади атак, методи виявлення. Підготовлені запити (prepared statements) та параметризація. Принцип найменших привілеїв для облікових записів БД. Хешування паролів (bcrypt, Argon2). Шифрування чутливих даних у БД.

Практика (1 год.) та Лабораторна (1 год.) і СРС (8 год.)

Практика: Демонстрація SQL-ін'єкції у вразливому застосунку; виявлення та усунення вразливості за допомогою підготовлених запитів та ORM.

Лабораторна (Лаб. 6): Розробка безпечного модуля реєстрації/входу користувача з хешуванням паролів (bcrypt) та захистом від SQL-ін'єкцій.

СРС: Опрацювання: Харченко В. С., Скляр В. В. «Безпека інформаційних технологій та веб-застосунків» (Харків: НАУ, 2022) — розділ «Вразливості баз даних»; підготовка аналітичної записки «SQL-ін'єкції: класифікація та методи захисту».

МОДУЛЬ 2 · ТЕМИ 2.3–2.4

RESTful API, автентифікація та авторизація

Тема 2.3 — RESTful API (Лекція 11, 2 год.)

Принципи REST (Representational State Transfer): stateless, uniform interface, layered system. Проектування ресурсів і маршрутів. HTTP-методи в контексті CRUD. Документування API за допомогою OpenAPI/Swagger. GraphQL як альтернатива REST. Версіонування API. Безпека API: ключі API, обмеження швидкості, CORS-налаштування, запобігання надмірному розкриттю даних.

Практика (1 год.): Проектування та документування RESTful API за допомогою Swagger UI; тестування API інструментом Postman; аналіз типових помилок безпеки API.

Лабораторна (Лаб. 7, 1 год.): Розробка RESTful API з документацією Swagger. Налаштування CORS, rate limiting та захисту від надмірного розкриття даних.

СРС (8 год.): Опрацювання: Яременко В. О., Сердюк Б. М. «Серверні веб-технології та захист інформації» (ХНУРЕ, 2023); підготовка порівняльної таблиці «REST vs GraphQL: безпека та продуктивність».

Тема 2.4 — Автентифікація та авторизація (Лекція 12, 2 год.)

Сесійна автентифікація: серверні сесії, cookie, механізми захисту (HttpOnly, Secure, SameSite). JSON Web Token (JWT): структура, підпис, перевірка, вразливості (alg:none, weak secret). OAuth 2.0: потоки авторизації (Authorization Code, Client Credentials, PKCE). OpenID Connect. Управління доступом: RBAC (Role-Based Access Control) та ABAC. Багатофакторна автентифікація (MFA/2FA) у веб-застосунках. Атаки на автентифікацію: brute force, credential stuffing, session fixation.

Практика (1 год.): Реалізація JWT-автентифікації; аналіз вразливостей OAuth 2.0; налаштування RBAC у веб-застосунку.

Лабораторна (Лаб. 8, 1 год.): Реалізація системи автентифікації на основі JWT з refresh-токенами та RBAC. Реалізація захисту від brute force (account lockout).

СРС (8 год.): Підготовка аналітичної записки «JWT vs сесії: порівняльний аналіз безпеки та сфери застосування».

МОДУЛЬ 3

Безпека веб–застосунків: OWASP та методи захисту

Тема 3.1

OWASP Top 10: класифікація вразливостей, методологія оцінки ризиків. Аналіз вразливостей веб-застосунків.

Тема 3.2

XSS (Cross-Site Scripting) та CSRF (Cross-Site Request Forgery): типи, механізми атак та комплексний захист.

Тема 3.3

Тестування безпеки веб-застосунків: SAST, DAST, пентест. Інструменти: Burp Suite, OWASP ZAP, Nikto.

Тема 3.4

Безпечна розробка (Secure SDLC): DevSecOps, управління залежностями, аудит коду, статичний аналіз.

МОДУЛЬ 3 · ТЕМА 3.1

OWASP Top 10: класифікація вразливостей веб-застосунків

Лекційний матеріал (2 год.)

OWASP (Open Web Application Security Project): місія, документи, методологія. OWASP Top 10 (2021): A01 Broken Access Control, A02 Cryptographic Failures, A03 Injection, A04 Insecure Design, A05 Security Misconfiguration, A06 Vulnerable Components, A07 Auth Failures, A08 Software Integrity Failures, A09 Logging Failures, A10 SSRF. Методологія оцінки ризиків: ймовірність, вплив, пріоритизація. CVSS (Common Vulnerability Scoring System). CVE та NVD бази вразливостей. Приклади реальних інцидентів безпеки у веб-застосунках.

Ключові питання

- Категорії OWASP Top 10 (2021) та їх характеристика
- Методологія оцінки ризиків веб-вразливостей
- Використання CVSS для пріоритизації усунення вразливостей
- Реальні кейси експлуатації вразливостей OWASP Top 10

Практичне заняття (1 год.)

Завдання 1. Аналіз вразливого застосунку DVWA (Damn Vulnerable Web Application): ідентифікація вразливостей OWASP Top 10, оцінка ризиків за CVSS.

Завдання 2. Кейс: аналіз реального інциденту безпеки (витік даних великої компанії) — визначення задіяних вразливостей з OWASP Top 10.

Лабораторна робота (Лаб. 9, 1 год.)

Дослідження вразливостей DVWA: виконання практичних атак (SQL Injection, XSS, File Inclusion) в контрольованому середовищі. Складання звіту про знайдені вразливості з оцінкою CVSS.

Самостійна робота (8 год.)

- Опрацювання: Харченко В. С., Скляр В. В. «Безпека інформаційних технологій та веб-застосунків» (НАУ, 2022)
- Підготовка реферату: «OWASP Top 10 (2021): аналіз нових категорій та порівняння з попередньою редакцією»
- Аналіз не менше 2 реальних CVE-вразливостей популярних CMS (WordPress, Drupal)

Очікуваний результат: студент вміє класифікувати вразливості веб-застосунків за OWASP Top 10 та оцінювати їх ризики за методологією CVSS.

МОДУЛЬ 3 · ТЕМА 3.2

XSS та CSRF: атаки та комплексний захист

Лекційний матеріал (2 год.)

XSS (Cross-Site Scripting): класифікація (Reflected, Stored, DOM-based). Механізми атак: викрадення cookie, сесійних токенів, перенаправлення, дефейс. Методи захисту від XSS: вихідне кодування (output encoding), Content Security Policy (CSP), DOMPurify, HttpOnly-cookie. CSRF (Cross-Site Request Forgery): механізм атаки, використання збережених cookie. Захист від CSRF: CSRF-токени, SameSite cookie, перевірка Origin/Referer заголовків, Double Submit Cookie Pattern. Clickjacking: механізм та захист через X-Frame-Options та CSP frame-ancestors.

Практика (1 год.) та Лабораторна (1 год.) і СРС (8 год.)

Практика: Демонстрація Stored XSS та CSRF атак у лабораторному середовищі DVWA; реалізація захисту за допомогою CSP, CSRF-токенів та SameSite cookie; перевірка ефективності захисту.

Лабораторна (Лаб. 10): Впровадження комплексного захисту від XSS та CSRF у тестовому веб-застосунку: налаштування CSP-заголовків, реалізація CSRF-токенів, санітизація введення через DOMPurify.

СРС: Опрацювання: Корченко О. Г., Архипов О. Є. «Методи та засоби захисту веб-ресурсів» (Київ: НАУ, 2022) — розділи «XSS» та «CSRF»; підготовка аналітичної записки «Content Security Policy: налаштування та ефективність захисту від XSS».

Очікуваний результат: студент вміє виявляти та усувати вразливості XSS і CSRF, налаштовувати CSP та реалізовувати захист від clickjacking.

МОДУЛЬ 3 · ТЕМА 3.3

Тестування безпеки веб-застосунків

Розвідка (Reconnaissance)

Збір інформації про цільовий застосунок: технологічний стек, поверхня атаки, відкриті порти, піддомени.

Сканування вразливостей (DAST)

Автоматизоване виявлення вразливостей за допомогою OWASP ZAP, Nikto, Nuclei у режимі чорного ящика.

Ручне тестування (Pentest)

Ручна верифікація вразливостей за допомогою Burp Suite: перехоплення запитів, fuzzing, аналіз відповідей.

Лекційний матеріал (2 год.)

Методологія тестування безпеки веб-застосунків: OWASP Testing Guide (OTG), PTES. Класифікація методів: SAST (статичний аналіз коду), DAST (динамічне тестування), IAST, RASP. Інструменти: Burp Suite Community/Pro (перехоплення та модифікація запитів), OWASP ZAP (автоматизований сканер), Nikto (сканер сервера), sqlmap (автоматизація SQL-ін'єкцій у тестових середовищах). Правові та етичні аспекти тестування. Складання звіту з пентесту.

Практика (1 год.) та Лабораторна (1 год.) і СРС (8 год.)

Практика: Налаштування Burp Suite для перехоплення HTTP-трафіку; сканування тестового застосунку за допомогою OWASP ZAP; аналіз отриманих звітів та пріоритизація вразливостей.

Лабораторна (Лаб. 11): Проведення комплексного пентесту тестового веб-застосунку (DVWA або WebGoat): розвідка, сканування OWASP ZAP, ручна верифікація Burp Suite, складання структурованого звіту.

СРС: Опрацювання: Корченко О. Г., Архипов О. Є. «Методи та засоби захисту веб-ресурсів» (НАУ, 2022); підготовка аналітичної записки «Інструменти тестування безпеки веб-застосунків: порівняльний огляд».

МОДУЛЬ 3 · ТЕМА 3.4

Безпечна розробка: Secure SDLC та DevSecOps

Лекційний матеріал (2 год.)

Secure Software Development Life Cycle (Secure SDLC): інтеграція безпеки на кожному етапі розробки. DevSecOps: принципи «безпека як код» (security-as-code), зсув ліворуч (shift-left security). Управління залежностями: SCA (Software Composition Analysis), виявлення вразливих бібліотек (npm audit, Snyk). Статичний аналіз коду (SAST): інструменти (ESLint security plugins, Semgrep, SonarQube). Code Review з фокусом на безпеку: чеклісти та best practices. CI/CD pipeline з інтегрованими перевірками безпеки. Управління секретами: змінні середовища, Vault, безпечне зберігання ключів.

Ключові питання

- Принципи Secure SDLC та DevSecOps
- Shift-left security: раннє виявлення вразливостей
- Управління залежностями та вразливими компонентами
- Автоматизація перевірок безпеки у CI/CD

Практичне заняття (1 год.)

Завдання 1. Аудит залежностей проекту за допомогою npm audit та Snyk: виявлення вразливих пакетів, аналіз CVE, стратегії оновлення.

Завдання 2. Налаштування базового CI/CD pipeline (GitHub Actions) з автоматичним запуском SAST та перевіркою залежностей.

Лабораторна робота (Лаб. 12, 1 год.)

Налаштування DevSecOps pipeline: SAST аналіз (Semgrep), аудит залежностей (npm audit), DAST сканування (OWASP ZAP) у GitHub Actions CI/CD. Реалізація безпечного управління секретами через змінні середовища.

Самостійна робота (8 год.)

- Опрацювання: Харченко В. С., Скляр В. В. «Безпека інформаційних технологій та веб-застосунків» (НАУ, 2022) — розділ «Secure SDLC»
- Підготовка аналітичної записки «DevSecOps: інтеграція безпеки в процес розробки програмного забезпечення»
- Складання чеклісту безпечного Code Review для веб-застосунків

Очікуваний результат: студент вміє інтегрувати перевірки безпеки у процес розробки, виконувати аудит залежностей та налаштовувати CI/CD з автоматизованими security-перевірками.

МОДУЛЬ 4

Сучасні веб–технології: фреймворки, хмари та безпека інфраструктури

Тема 4.1

Сучасні JS-фреймворки (React, Vue, Angular). SPA, PWA та SSR-архітектури. Безпека фронтенд-фреймворків.

Тема 4.2

Хмарні технології та контейнеризація: Docker, Nginx, CDN. Розгортання та безпека хмарної інфраструктури.

Тема 4.3

WebSocket, Server-Sent Events, Web Workers. Реального часу веб-застосунки та їх захист.

Тема 4.4

Комплексний проект: розробка та аудит безпеки повнофункціонального захищеного веб-застосунку.

МОДУЛЬ 4 · ТЕМА 4.1

Сучасні JavaScript-фреймворки та архітектури SPA/PWA

Лекційний матеріал (2 год.)

Огляд сучасних JS-фреймворків: React (компонентна модель, Virtual DOM, hooks), Vue.js (реактивність, Options/Composition API), Angular (TypeScript, DI, модулі). Порівняльна характеристика з точки зору безпеки. SPA (Single Page Application): переваги, недоліки, вразливості (DOM-based XSS, відкриті маршрути). PWA (Progressive Web App): Service Workers, Web App Manifest, кешування. SSR (Server-Side Rendering) та SSG (Static Site Generation): Next.js, Nuxt.js. Безпека state management (Redux, Vuex, Pinia): зберігання чутливих даних.

Ключові питання

- Порівняння React, Vue та Angular у контексті безпеки
- Вразливості SPA та методи їх усунення
- Безпека Service Workers у PWA
- XSS у контексті JS-фреймворків: dangerouslySetInnerHTML, v-html
- Захист маршрутизації та захищені маршрути (protected routes)

Практичне заняття (1 год.)

Завдання 1. Аналіз безпеки React-застосунку: виявлення небезпечного використання dangerouslySetInnerHTML, небезпечних redirect та відкритих маршрутів.

Завдання 2. Реалізація захищених маршрутів (Protected Routes) у React/Vue з перевіркою JWT-токена та управлінням ролями.

Лабораторна робота (Лаб. 13, 1 год.)

Розробка SPA-застосунку на React або Vue.js із JWT-автентифікацією, RBAC та захистом від DOM-based XSS. Аудит безпеки реалізованого фронтенду за допомогою Burp Suite.

Самостійна робота (8 год.)

- Опрацювання: Олексій Васильченко, Андрій Кравченко «Веб-технології та веб-дизайн» (КПІ, 2023) — розділ «Сучасні фреймворки»
- Підготовка реферату: «Безпека SPA-застосунків: огляд вразливостей та методів захисту»
- Порівняльний аналіз підходів до захисту від XSS у React, Vue та Angular

Очікуваний результат: студент вміє розробляти SPA з використанням сучасних фреймворків та реалізовувати захист від типових вразливостей фронтенд-застосунків.

МОДУЛЬ 4 · ТЕМА 4.2

Хмарні технології та контейнеризація веб-застосунків

Docker
Контейнеризація застосунків. Dockerfile. Безпека образів. Docker Compose для мультисервісних застосунків.

CDN
Content Delivery Network: принципи роботи, безпека (DDoS-захист, WAF), кешування статичних ресурсів.

1 2 3 4

Nginx
Веб-сервер та reverse proxy. Налаштування HTTPS, заголовків безпеки, rate limiting, балансування навантаження.

Хмарні платформи
AWS, Azure, GCP: моделі розгортання (IaaS, PaaS, SaaS). Спільна відповідальність за безпеку (shared responsibility).

Лекційний матеріал (2 год.)

Контейнеризація та Docker: переваги, Dockerfile, Docker Compose, безпека контейнерів (rootless, read-only filesystem, seccomp). Nginx як веб-сервер та reverse proxy: налаштування HTTPS/TLS, HTTP/2, заголовків безпеки (HSTS, CSP, X-Frame-Options). WAF (Web Application Firewall): принципи роботи, правила ModSecurity, хибно позитивні спрацювання. CDN та DDoS-захист (Cloudflare, AWS CloudFront). Хмарні платформи та модель спільної відповідальності за безпеку.

Практика (1 год.) та Лабораторна (1 год.) і СРС (8 год.)

Практика: Налаштування Nginx як reverse proxy з HTTPS та заголовками безпеки; контейнеризація веб-застосунку за допомогою Docker Compose; сканування Docker-образу на вразливості (Trivy).

Лабораторна (Лаб. 14): Розгортання повного стеку веб-застосунку (Frontend + Backend + DB + Nginx) у Docker Compose з налаштуванням HTTPS, заголовків безпеки та базових WAF-правил.

СРС: Опрацювання: Яременко В. О., Сердюк Б. М. «Серверні веб-технології та захист інформації» (ХНУРЕ, 2023); підготовка аналітичної записки «Docker Security: best practices для захищеної контейнеризації веб-застосунків».

МОДУЛЬ 4 · ТЕМА 4.3

WebSocket, реального часу технології та їх захист

Лекційний матеріал (2 год.)

WebSocket: принципи роботи, рукостискання (handshake), фрейми, протокол wss://. Порівняння WebSocket з HTTP long polling, SSE (Server-Sent Events). Безпека WebSocket: відсутність SOP, CSRF-вразливості, перехоплення з'єднання, DoS-атаки. Захист WebSocket: перевірка Origin, автентифікація через токени, TLS (wss://), rate limiting. Web Workers та SharedArrayBuffer: продуктивність та безпека. WebRTC: peer-to-peer з'єднання та питання безпеки. Приклади застосування: чати, онлайн-ігри, системи моніторингу в реальному часі.

Ключові питання

- Відмінності WebSocket від HTTP та їх безпековий вплив
- Вразливості WebSocket та методи захисту
- Автентифікація у WebSocket з'єднаннях через JWT
- Захист від DoS-атак на WebSocket-сервер

Практичне заняття (1 год.)

Завдання 1. Аналіз WebSocket-трафіку за допомогою Burp Suite: перехоплення повідомлень, виявлення вразливостей (відсутність автентифікації, передача чутливих даних).

Завдання 2. Реалізація захищеного WebSocket-сервера: перевірка Origin, JWT-автентифікація при підключенні, rate limiting повідомлень.

Лабораторна робота (Лаб. 15, 1 год.)

Розробка захищеного чату реального часу на основі WebSocket (socket.io або ws): JWT-автентифікація, перевірка Origin, шифрування wss://, захист від спаму через rate limiting.

Самостійна робота (8 год.)

- Опрацювання: Корченко О. Г., Архипов О. Є. «Методи та засоби захисту веб-ресурсів» (НАУ, 2022) — розділ «Безпека протоколів реального часу»
- Підготовка аналітичної записки «WebSocket Security: вразливості та заходи захисту»
- Огляд та аналіз інцидентів безпеки, пов'язаних із WebSocket-застосунками

Очікуваний результат: студент вміє розробляти та захищати веб-застосунки реального часу на основі WebSocket, виявляти та усувати типові вразливості протоколу.

МОДУЛЬ 4 · ТЕМА 4.4

Комплексний проект: розробка та аудит захищеного веб–застосунку

Лекційний матеріал (2 год.)

Підсумкова лекція: комплексна архітектура захищеного веб-застосунку. Secure by Design принципи: defense in depth, fail secure, principle of least privilege, complete mediation. Чеклісти безпеки веб-застосунку: OWASP ASVS (Application Security Verification Standard) рівні 1–3. Управління інцидентами: виявлення, реагування, відновлення. Логування та моніторинг безпеки: SIEM, централізований збір логів (ELK Stack). Документація безпеки: Threat Modeling (STRIDE), модель загроз для веб-застосунку.

Ключові питання

- OWASP ASVS: рівні верифікації безпеки
- Defense in depth у контексті веб-застосунків
- Threat Modeling методологія STRIDE
- Організація моніторингу та реагування на інциденти
- Перспективи розвитку безпеки веб-технологій

Практичне заняття (1 год.)

Завдання 1. Threat Modeling для розробленого в рамках курсу веб-застосунку за методологією STRIDE: побудова діаграми потоків даних (DFD), ідентифікація загроз, оцінка ризиків, план митигації.

Завдання 2. Фінальний аудит безпеки власного проекту за чеклістом OWASP ASVS (рівень 1). Презентація результатів.

Лабораторна робота (Лаб. 15 — фінальна, 1 год.)

Комплексний аудит безпеки власного веб-застосунку: SAST аналіз (Semgrep), DAST сканування (OWASP ZAP), перевірка залежностей (npm audit), тест автентифікації та авторизації. Підготовка фінального звіту безпеки з рекомендаціями.

Самостійна робота (8 год.)

- Доопрацювання комплексного проекту відповідно до результатів аудиту
- Підготовка аналітичної записки «Threat Modeling власного веб-застосунку: ідентифіковані загрози та заходи митигації»
- Складання фінального портфоліо виконаних лабораторних робіт та звіту про набуті компетентності

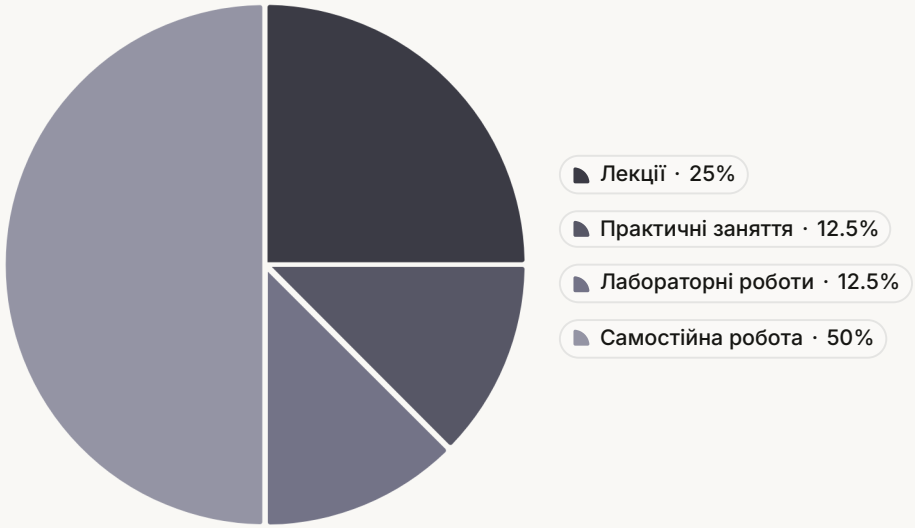
Очікуваний результат: студент вміє проводити комплексний аудит безпеки веб-застосунку, будувати модель загроз та складати фаховий звіт з рекомендаціями щодо підвищення рівня захисту.

Розподіл навчального часу — 120 годин

Модуль	Тема	Лекції (год.)	Практ. (год.)	Лаб. (год.)	СРС (год.)
Модуль 1	Тема 1.1 — HTTP/HTTPS, архітектура	2	1	1	8
	Тема 1.2 — HTML5	2	1	1	8
	Тема 1.3 — CSS3	2	1	1	8
	Тема 1.4 — JavaScript	6	1	1	8
Модуль 2	Тема 2.1 — Серверні технології	2	1	1	8
	Тема 2.2 — Бази даних, SQL-ін'єкції	2	1	1	8
	Тема 2.3 — RESTful API	2	1	1	8
	Тема 2.4 — Автентифікація, JWT, OAuth	2	1	1	8
Модуль 3	Тема 3.1 — OWASP Top 10	2	1	1	8
	Тема 3.2 — XSS та CSRF	2	1	1	8
	Тема 3.3 — Тестування безпеки	2	1	1	8
	Тема 3.4 — Secure SDLC, DevSecOps	2	1	1	8
Модуль 4	Тема 4.1 — JS-фреймворки, SPA/PWA	2	1	1	8
	Тема 4.2 — Docker, Nginx, CDN	2	1	1	8
	Тема 4.3 — WebSocket, реальний час	2	1	1	8
	Тема 4.4 — Комплексний проект	2	1	1	8
Разом	**16 тем**	**30**	**15**	**15**	**60**

Загальний обсяг: 120 годин = 4 кредити ЄКТС. Розподіл є орієнтовним і може коригуватися відповідно до специфіки навчального процесу та технічного оснащення лабораторій.

Структура навчального часу



Коментар до розподілу

Загальний обсяг — **120 годин (4 кредити ЄКТС)** — розподілено між чотирма формами роботи для забезпечення балансу між теорією, практикою та самостійним дослідженням.

Лекції — 30 год. (25%)

Системне подання теоретичного матеріалу з використанням наочних схем, діаграм архітектури та демонстрацій атак і захисту у безпечному середовищі.

Практичні — 15 год. (12,5%)

Аналіз вразливостей, кейс-стаді реальних інцидентів, дискусії та захист аналітичних завдань. Акцент на критичному аналізі технічних рішень.

Лабораторні — 15 год. (12,5%)

Виконання 15 індивідуальних лабораторних робіт у спеціалізованому середовищі: розробка, тестування та аудит безпеки реальних веб-застосунків.

Самостійна робота — 60 год. (50%)

Поглиблене вивчення технологій, виконання аналітичних завдань, підготовка рефератів, дослідження CVE та розробка проектів. СРС становить **50% обсягу дисципліни**.

Самостійна робота студентів — Детальний опис

Опрацювання літератури

Поглиблене вивчення теоретичного матеріалу за підручниками та посібниками з веб-технологій та захисту інформації (Харченко, Корченко, Яременко, Васильченко та ін.), специфікаціями RFC, стандартами OWASP та документацією MDN Web Docs. Студент конспектує ключові положення та порівнює підходи різних авторів.

Робота з інформаційними ресурсами

Опрацювання офіційних ресурсів: OWASP (**owasp.org**), MDN Web Docs (**developer.mozilla.org**), NVD NIST (**nvd.nist.gov**), W3C (**w3.org**). Аналіз CVE-баз, пошук актуальних вразливостей та їх опис у звітах.

Індивідуальні завдання

Виконання аналітичних завдань (порівняльний аналіз технологій, аналіз CVE-вразливостей, характеристика архітектурних рішень); творчих завдань — реферати, аналітичні записки, технічні огляди та міні-дослідження з безпеки веб-застосунків.

Підготовка до занять

Підготовка до практичних та лабораторних занять (вивчення документації інструментів, повторення теорії); підготовка до тестувань (ключові терміни, вразливості, протоколи); підготовка до модульного контролю (систематизація знань, вирішення типових задач).

Форми контролю знань

1	2	3
Поточний контроль	Проміжний контроль	Підсумковий контроль
Завдання: тестові та аналітичні вправи за темами змістовного модуля. Оцінюються правильність виконання лабораторних робіт, якість звітів про виявлені вразливості та рівень засвоєння матеріалу. Проводиться після кожної теми у формі тестування (10–15 питань) або захисту лабораторної роботи.	Модульна контрольна робота — комплексне завдання, що охоплює теми змістовного модуля (теоретичні питання, аналіз вразливостей, практичні задачі з захисту веб-застосунку). Оцінюються повнота відповіді, технічна коректність та обґрунтованість рішень. Проводиться після завершення кожного модуля.	Залік/іспит відповідно до навчального плану. Форма проведення — тестування та практичне завдання (аудит безпеки тестового веб-застосунку). Підсумкова оцінка враховує результати поточного, проміжного та підсумкового контролю: **100-бальна система** з переведенням у національну шкалу та ЄКТС (A, B, C, D, E, FX, F).

Методи викладання

Викладання дисципліни «WEB-технології» поєднує класичні академічні та практично-орієнтовані методи, що забезпечує формування не лише міцної теоретичної бази, а й практичних навичок розробки та аудиту безпеки веб-застосунків, необхідних для майбутньої професійної діяльності у сфері захисту інформації.

Лекції

Системне подання теоретичного матеріалу з демонстраціями атак і захисту у безпечному середовищі; студенти ведуть конспекти та аналізують архітектурні схеми.

Лабораторні роботи

Практична розробка та тестування захищених веб-застосунків; виконання пентестів у контрольованому середовищі; складання фахових звітів про вразливості.

Кейс-стаді

Аналіз реальних інцидентів безпеки та CVE-вразливостей; студенти виявляють причини, вектори атак та пропонують заходи усунення на основі OWASP.

Технічні дискусії

Обговорення архітектурних рішень, порівняння технологій та підходів до захисту; розвиток критичного мислення та аргументованого технічного обґрунтування.

Самостійне дослідження

Підготовка аналітичних записок, рефератів та технічних оглядів; самостійний пошук CVE-вразливостей, аналіз документації та стандартів OWASP/W3C.

Code Review

Взаємний аналіз коду студентів з фокусом на виявлення вразливостей; формування навичок безпечної розробки та критичного сприйняття чужих технічних рішень.

Усі методи спрямовані на формування компетентностей, передбачених освітньо-професійною програмою «Організація захисту інформації в ІКС» спеціальності 123 «Комп'ютерна інженерія».

Рекомендована література — Підручники та посібники (2022–2024 рр.)

1. Харченко В. С., Скляр В. В.
Безпека інформаційних технологій та веб-застосунків: навч. посібник. — Харків : НАУ «ХАІ», 2022. — 312 с. *(Рекомендується для модулів 2–3.)*

2. Корченко О. Г., Архипов О. Є.
Методи та засоби захисту веб-ресурсів: навч. посібник. — Київ : НАУ, 2022. — 268 с. *(Рекомендується для модулів 3–4.)*

3. Яременко В. О., Сердюк Б. М.
Серверні веб-технології та захист інформації: навч. посібник. — Харків : ХНУРЕ, 2023. — 280 с. *(Рекомендується для модулів 2 та 4.)*

4. Васильченко О. С., Кравченко А. В.
Веб-технології та веб-дизайн: навч. посібник для студентів ЗВО. — Київ : КПІ ім. Ігоря Сікорського, 2023. — 320 с. *(Рекомендується для модуля 1.)*

5. Гуржій А. М., Поворознюк Н. І., Самсонов В. В.
Основи інформаційних технологій: навч. посібник для здобувачів вищої освіти. — Київ : Літера ЛТД, 2023. — 296 с. *(Рекомендується для теми 1.1.)*

6. Дорошенко А. Ю., Жабін В. І.
Технології розробки програмного забезпечення для веб: підручник. — Київ : Інститут програмних систем НАН України, 2023. — 344 с. *(Рекомендується для модуля 2.)*

7. Андрейчук В. О., Стисло О. В., Куліш С. М.
Методики тестування веб-додатків та забезпечення їх безпеки: монографія. — Івано-Франківськ : Університет Короля Данила, 2023. — 153 с. *(Рекомендується для теми 3.3.)*

8. Кузьмінська О. Г., Мацапура О. В.
Сучасні технології розробки клієнт-серверних застосунків: навч. посібник. — Київ : Академвидав, 2024. — 256 с. *(Рекомендується для модулів 1–2.)*

Додаткова література та наукові статті

9. Батіг В. М., Стисло Т. О.

Типологія веб-сайтів та їх особливості в умовах вимог безпеки. — Матеріали I Всеукраїнської науково-практичної інтернет-конференції «IT-екосистема». — Івано-Франківськ, 2023. — С. 13–17. *Рекомендується для теми 1.1.*

10. Князевич М. Р., Ващишак С. П., Бандура В. М.

Класифікаційні моделі визначення сигнатур програмного коду для виявлення вразливостей. — IT-екосистема: цифровізація бізнес-процесів. — Івано-Франківськ : Університет Короля Данила, 2023. — С. 36–39. *Рекомендується для теми 3.4.*

11. OWASP Foundation

OWASP Top 10 — 2021: The Ten Most Critical Web Application Security Risks. — OWASP Foundation, 2021. — Режим доступу: **owasp.org/www-project-top-ten**. *Рекомендується для тем 3.1–3.3.*

12. Куриш В. О., Кітура В. М.

Роль сучасних веб-технологій у забезпеченні кібербезпеки інформаційних систем. — Матеріали науково-практичної конференції «Сучасні цифрові технології та інноваційні методики навчання». — Тернопіль: ТНПУ, 2024. — С. 38–42. *Рекомендується для теми 4.2.*

Нормативно-правова база

- Конституція України

  Прийнята 28 червня 1996 р. Основний Закон держави. **zakon.rada.gov.ua**

- Закон «Про основні засади забезпечення кібербезпеки України» від 05.10.2017 № 2163-VIII

  Визначає правові та організаційні засади у сфері кібербезпеки. **zakon.rada.gov.ua**

- Закон «Про захист персональних даних» від 01.06.2010 № 2297-VI

  Регулює обробку та захист персональних даних у інформаційних системах. **zakon.rada.gov.ua**

- Закон «Про інформацію» від 02.10.1992 № 2657-XII (у редакції 2022 р.)

  Визначає основи правового режиму інформації та інформаційних відносин. **zakon.rada.gov.ua**

- Стандарт ДСТУ ISO/IEC 27001:2015 — Системи управління інформаційною безпекою

  Національний стандарт управління інформаційною безпекою. **zakon.rada.gov.ua**

Офіційні ресурси та фахові видання

OWASP Foundation

owasp.org — провідна міжнародна організація з безпеки веб-застосунків: Top 10, ASVS, Testing Guide, Cheat Sheets.

MDN Web Docs (Mozilla)

developer.mozilla.org — офіційна документація веб-стандартів: HTML, CSS, JavaScript, Web APIs, HTTP.

NVD — National Vulnerability Database

nvd.nist.gov — база даних CVE-вразливостей NIST для аналізу актуальних загроз веб-застосункам.

W3C — World Wide Web Consortium

w3.org — офіційні стандарти та специфікації HTML, CSS, SVG, ARIA, WebRTC та інших веб-технологій.

Верховна Рада України

zakon.rada.gov.ua — нормативно-правова база у сфері кібербезпеки, захисту інформації та персональних даних.

Національна бібліотека України ім. В. І. Вернадського

nbuv.gov.ua — електронні ресурси, наукові статті та монографії з інформаційних технологій та кібербезпеки.

Журнал «Захист інформації»

zi.nau.edu.ua — фахове наукове видання НАУ з питань захисту інформації та кібербезпеки.

Міністерство освіти і науки України

mon.gov.ua — освітні стандарти, навчальні програми, методичні матеріали для ЗВО.

Зв'язок дисципліни з ОП «Організація захисту інформації в ІКС»

Чому WEB–технології для захисту інформації?

Дисципліна «WEB-технології» є невід'ємною складовою підготовки бакалаврів за освітньою програмою «Організація захисту інформації в інформаційно-комунікаційних системах». Сучасні ІКС базуються на веб-стеку, тому розуміння принципів розробки та захисту веб-застосунків є ключовим для фахівця з кібербезпеки.

Веб-застосунки є найбільш поширеним вектором кібератак: за даними OWASP, понад 75% атак спрямовані саме на рівень застосунків. Тому майбутній фахівець із захисту інформації зобов'язаний розуміти як розробку, так і аудит безпеки веб-систем.

Практичне значення для майбутньої кар'єри

- Проведення аудиту безпеки та пентесту веб-застосунків організації
- Розробка технічних вимог до безпеки корпоративних веб-систем
- Виявлення та усунення вразливостей у критичній веб-інфраструктурі ІКС
- Налаштування WAF, HTTPS та заголовків безпеки для корпоративних веб-сервісів
- Забезпечення безпеки API та мікросервісної архітектури в ІКС
- Проведення Threat Modeling та документування ризиків безпеки веб-систем

Ключові компетентності курсу — Підсумок

Курс формує у студентів цілісне розуміння сучасного веб-стеку та практичні навички розробки й аудиту захищених веб-застосунків — від базової розмітки до комплексного пентесту відповідно до стандартів OWASP.

Бажаємо успіхів у вивченні WEB-технологій!

120

Годин

Загальний обсяг дисципліни

4

Кредити ЄКТС

Відповідно до стандарту

15

Лабораторних

Практичних робіт з безпеки

10+

Джерел

Рекомендованої літератури 2022–2024 рр.

> «Безпека — це не продукт, а процес.» — Брюс Шнайєр

Спеціальність 123 · Комп'ютерна інженерія · Бакалавр · 4 кредити ЄКТС · ОП «Організація захисту інформації в ІКС»