МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

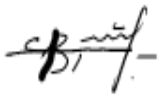
Львівський національний університет імені Івана Франка

Факультет прикладної математики та інформатики

Кафедра кібербезпеки

Затверджено
На засіданні кафедри кібербезпеки
факультету прикладної математики
та інформатики
Львівського національного університету
імені Івана Франка
(Протокол №9/24 від 29 серпня 2024 р.)

Завідувач кафедри



Петро ВЕНГЕРСЬКИЙ

Силабус з навчальної дисципліни

«Аналіз шкідливого програмного забезпечення та розвідка загроз»,

що викладається в межах ОПП Технології штучного інтелекту в кібербезпеці другого (магістерського) рівня вищої освіти для здобувачів з спеціальності 125 Кібербезпека та захист інформації

Львів - 2024

Назва дисципліни	Аналіз шкідливого програмного забезпечення та розвідка загроз
Адреса викладання дисципліни	м. Львів, вул. Університетська 1
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	Венгерський Петро Сергійович, д. ф.-м. н., проф. кафедри кібербезпеки; Щербина Микола Юрійович, асистент кафедри кібербезпеки.
Контактна інформація викладачів	Petro.Venherskyy@lnu.edu.ua https://ami.lnu.edu.ua/employee/venherskyi Mykola.Shcherbyna@lnu.edu.ua https://ami.lnu.edu.ua/employee/shcherbyna-m-yu Головний корпус ЛНУ ім. І. Франка, каб. 380 м. Львів, вул. Університетська, 1
Консультації з питань навчання по дисципліні відбуваються	Консультація проводиться за розкладом консультацій викладача. Можливі дистанційні консультації за попередньою домовленістю.
Сторінка курсу	https://ami.lnu.edu.ua/admission/specializations
Інформація про дисципліну	Дисципліна «Аналіз шкідливого програмного забезпечення та розвідка загроз» є вибірковою дисципліною з спеціальності 125 Кібербезпека та захист інформації для освітньої програми «Технології штучного інтелекту в кібербезпеці», яка викладається в 2-му семестрі в обсязі 3,5 кредитів (за Європейською Кредитно-Трансферною Системою ECTS).
Коротка анотація дисципліни	Дослідження шкідливого ПЗ. Статичний та динамічний аналіз виконуваних файлів в ОС Windows і Linux, для процесорів архітектури x86-64 і Aarch64. Розвідка загроз.
Мета та цілі дисципліни	Метою викладання дисципліни є навчити студентів методам статичного та динамічного аналізу шкідливого ПЗ в ОС Windows і Linux, з використанням відповідних інструментів, забезпечити розуміння на необхідному рівні системи команд актуальних процесорів x86-64 і Aarch64, способів інфікування системи шкідливим ПЗ та приховування його наявності.
Література для вивчення дисципліни	**Базова** 1. Domas S., Domas C. x86 Software Reverse-Engineering, Cracking, and Counter-Measures. Wiley, 2024. 320 p. 2. Duntemann J. x64 Assembly Language Step-by-Step: Programming with Linux [4 ed.] Wiley, 2023. 640 p.

	3. IDA Help: The Interactive Disassembler Help Index. Hex Rays – State-of-the-art binary code analysis solutions. URL: https://hex-rays.com/products/ida/support/idadoc/index.shtml 4. Bulazel A., Blackthorne J. Three Heads Are Better Than One: Mastering NSA's Ghidra Reverse Engineering Tool. URL: https://raw.githubusercontent.com/0xAlexei/INFILTRATE2019/master/INFILTRATE%20Ghidra%20Slides.pdf 5. Jiang L., An J., Huang H., Tang Q., Nie S., Wu S., Zhang Y. BinaryAI: Binary Software Composition Analysis via Intelligent Binary Source Code Matching. Proceedings of the 46th International Conference on Software Engineering (ICSE'24). URL: https://arxiv.org/pdf/2401.11161 **Допоміжна** 6. Andriesse D. Practical Binary Analysis: Build Your Own Linux Tools for Binary Instrumentation, Analysis, and Disassembly. San Francisco: No Starch Press, 2018. 456 p. 7. Dang B., Gazet A., Bachaalany E. Practical Reverse Engineering x86, x64, ARM, Windows® Kernel, Reversing Tools, and Obfuscation. Wiley, 2014. 384 p. 8. Russinovich M. E., Margosis A. Troubleshooting with the Windows Sysinternals Tools, 2nd Edition. Redmond: Microsoft Press, 2016. 688 p. 9. Intel® 64 and IA-32 Architectures Software Developer's Manual. Combined Volumes: 1, 2A, 2B, 2C, 2D, 3A, 3B, 3C, 3D, and 4. Intel Corporation, 2023. 5066 p. URL: https://software.intel.com/en-us/download/intel-64-and-ia-32-architectures-sdm-combined-volumes-1-2a-2b-2c-2d-3a-3b-3c-3d-and-4 10. Instruction Set Assembly Guide for Armv7 and earlier Arm® architectures. Version 2.0. Reference Guide. Arm Limited, 2019. 590 p. URL: https://documentation-service.arm.com/static/5e7b6a6216d2907d594035c4 11. Arm® A64 Instruction Set Architecture Armv8, for Armv8-A architecture profile. Arm Limited, 2021. 3289 p. URL: https://documentation-service.arm.com/static/61c04c7a2183326f21771ec6 12. dotPeek Documentation. JetBrains s.r.o., 2024. URL: https://www.jetbrains.com/decompiler/documentation/ 13. Levin J. Dalvik and ART. 2024. URL: https://newandroidbook.com/files/Andevcon-DEX.pdf 14. Hacksi. How to Analyze a Malicious Powershell Script & Fileless Malware. 2024. URL: https://youtu.be/nxO32fLoEbs
Обсяг курсу	Загальний обсяг: 105 годин. Аудиторних занять: 32 год., з них 16 год. лекцій та 16 год. лабораторних робіт. Самостійної роботи: 73 год.
Очікувані результати навчання	В результаті вивчення дисципліни фахівець повинен **знати:** – структуру виконуваних файлів формату PE та ELF, мобільних застосунків, способи ін'єкції шкідливого ПЗ; – системи команд процесорів архітектури x86-64 і Aarch64 на необхідному для дизасемблювання рівні, механізми передачі параметрів у підпрограми;

	– методи компресії, обфускації та захисту від відлагодження, що використовуються шкідливим ПЗ. Підготовлений фахівець повинен **вміти**: – здійснювати статичний аналіз шкідливого ПЗ методом зворотної розробки, з використанням інструментів IDA та Ghidra; – здійснювати динамічний аналіз шкідливого ПЗ шляхом аналізу поведінки, з використанням пісочниць та відповідних інструментів, зокрема відлагоджувачів; – застосовувати сучасні засоби ШІ для аналізу підозрілого ПЗ.
Ключові слова	Aarch64, ELF, malware, PE, ransomware, spyware, x86-64, виконуваний файл, відлагоджувач, вірус, декомпілятор, дизасемблер, динамічний аналіз, ін'єкція коду, інфікування, обфускація, пісочниця, статичний аналіз, троян, хробак, шкідливе ПЗ.
Формат курсу	Очний. Проведення лекцій, лабораторних занять і консультацій.
Теми	Теми подані у Схемі курсу нижче
Підсумковий контроль, форма	Залік в кінці 2-го семестру
Пререквізити	Для вивчення курсу студенти потребують базових знань з курсів: – Програмування – Основи операційних систем – Прикладна криптологія
Навчальні методи та тех.-ніки, які будуть використовуватися під час викладання курсу	Презентації, лекції, лабораторні роботи, індивідуальні завдання, індивідуальні доповіді, самостійна робота. Лекційні та лабораторні: інформаційно-рецептивний метод, репродуктивний метод, евристичний метод, метод проблемного викладу. Самостійна робота: репродуктивний метод, дослідницький метод.
Необхідне обладнання	Комп'ютерний клас із вільно-доступним програмним забезпеченням, зокрема IDA Free та Ghidra, ОС Windows та Linux, віртуальні машини, доступ до Internet мережі.
Критерії оцінювання (окремо для кожного виду навчальної діяльності)	Оцінювання проводиться за 100-бальною шкалою. Бали нараховуються за наступним співвідношенням: • лабораторні роботи: 45% семестрової оцінки; максимальна кількість балів – 45; • підсумкове індивідуальне завдання: 45% семестрової оцінки; максимальна кількість балів – 45; • додаткові бали за поточне опитування на лекціях і лабораторних заняттях 10% семестрової оцінки; максимальна кількість балів – 10. Підсумкова максимальна кількість балів – 100. **Академічна доброчесність:** всі студенти отримують принципово різні

	зразки підсумкового індивідуального завдання, що унеможливлює пряме запозичення. Очікується, що роботи студентів будуть оригінальними дослідженнями чи міркуваннями. Списування та втручання в роботу інших студентів становлять, але не обмежують, приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності в написанні завдань є підставою для її незарахуванння викладачем, незалежно від масштабів плагіату чи обману. Жодні форми порушення академічної доброчесності не толеруються. **Відвідання занять** є важливою складовою навчання. Очікується, що всі студенти відвідають усі лекції та лабораторні заняття курсу. Студенти повинні інформувати викладача про неможливість відвідати заняття. У будь-якому випадку студенти зобов'язані дотримуватися термінів визначених для виконання всіх видів робіт, 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 контролі та бали за виконання лабораторних робіт і підсумкового індивідуального завдання. При цьому обов'язково враховуються присутність на заняттях та активність студента під час лабораторного 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. **Оцінювання на лекційних заняттях:** поточне опитування по 1-2 бали за правильну відповідь, сумарно – до 5 балів. **Оцінювання роботи на лабораторних заняттях:** 1) поточне опитування по 1-2 бали за правильну відповідь, сумарно – до 5 балів; 2) виконання 5-ти лабораторних робіт по 9 балів, сумарно – 45 балів. Бали оцінювання лабораторної роботи нараховуються пропорційно кількості виконаних завдань з лабораторної роботи з урахуванням відповідей на поставлені запитання. За оригінальне виконання лабораторної роботи може додаватися 1-2 бали. **Оцінювання самостійної роботи.** За рахунок годин самостійної роботи студенти освоюють теоретичний матеріал, виконують лабораторні роботи та індивідуальне завдання. Спеціальне оцінювання не проводиться. **Оцінювання індивідуального завдання:** Індивідуальне завдання полягає в аналізі двох програм-імітаторів шкідливого ПЗ для Windows (архітектура x86-64) та Linux (архітектура Aarch64), з використанням декомпіляторів IDA Free та Ghidra відповідно, та інших допоміжних інструментів. По кожному зразку необхідно відповісти на 4 питання (2, 3, 4 та 5 балів), які потребують різного рівня аналізу. Надається проєкт IDA Free/Ghidra, якість та повнота декомпіляції оцінюється до 6 балів. Робота по одному з двох зразків (на вибір викладача) захищається шляхом доповіді (до 5 балів). Максимальна оцінка – 45 балів (20 + 20 + 5).

	Критерії оцінювання результатів неформальної освіти: Нарахування балів відбувається за публікацію студентом тез доповідей на конференціях, наукових статей, за участь студента у діяльності наукових гуртків, семінарів, круглих столів, конкурсів (CTF-змагань), участь у заходах неформальної освіти, за отримання сертифікатів про проходження навчання на різних освітніх платформах (Coursera, Prometheus тощо). Кількість балів визначається відсотком покриття результатів відповідної активності до вимог результатів навчання з навчальної дисципліни.
Питання до контрольної роботи. Залік.	Питання підсумкового індивідуального завдання відповідають темам курсу. Залік – за результатами поточного контролю протягом семестру і, за потреби, додаткове усне опитування за тематикою курсу.
Опитування	Анкету-оцінку з метою оцінювання якості курсу буде надано по завершенню курсу.

Схема курсу

Тиж.	Тема, план, короткі тези	Форма діяльності (заняття)	Література	Завдання, год.	Термін виконання
1	**Процесори архітектури x86-64 для реверс-інженера**. Перехід від 32-розрядної архітектури x86 до 64-розрядної. Регістри. Доступ до пам'яті. Операції обробки даних. Умовні та безумовні переходи. Стек та підпрограми. Погодження викликів. Функції Windows API. Формат виконуваних файлів PE. Релокації. DLL бібліотеки. Прості інструменти командного рядка (DUMPBIN). Способи ін'єкції коду.	лекція, лаб., самостійна робота	[1, 2, 8, 9]	1 1 8	1 тиждень
2-3	**Дизасемблювання та декомпіляція PE файлів за допомогою IDA Free**. Робота з вікнами програми. Основні операції (навігація по коду, найменування, зміна типів, коментування, імпортовані ф-ції, перехресні зв'язки тощо). Декомпіляція у С-псевдокод.	лекція, лаб., самостійна робота	[1, 4, 8]	2 2 12	2 тижні
4-5	**Процесори архітектури Aarch64 для реверс-інженера**. Перехід від 32-розрядної архітектури ARM до 64-розрядної. Регістри. Доступ до пам'яті. Операції обробки даних. Умовні та безумовні переходи. Стек та підпрограми. Погодження	лекція, лаб., самостійна робота	[8, 10, 11]	2 2 8	2 тижні

	викликів. Системні виклики Linux в Aarch64. Формат виконуваних файлів ELF (особливості Aarch64). Релокації. Спільні бібліотеки. Прості інструменти командного рядка (binutils). Способи ін'єкції коду.				
6-7	**Дизасемблювання та декомпіляція ELF файлів за допомогою Ghidra**. Робота з вікнами програми. Основні операції (навігація по коду, найменування, зміна типів, коментування, імпортовані ф-ції, перехресні зв'язки тощо). Декомпіляція у С-псевдокод.	лекція, лаб., самостійна робота	[5, 8]	2 2 12	2 тижні
8-9	**Доповнення та закріплення знань**. Формат виконуваних файлів ELF (особливості x86-64). Системні виклики Linux в x86-64. Релокації. Спільні бібліотеки. Способи ін'єкції коду. Дизасемблювання та декомпіляція ELF файлів за допомогою IDA Free. Дизасемблювання та декомпіляція PE файлів за допомогою Ghidra.	лекція, лаб., самостійна робота	[1, 2-5, 8]	2 2 8	2 тижні
10-11	**Динамічне дослідження підозрілого ПЗ у Windows та Linux**. Виділені, віртуальні машини та пісочниці (на прикл. Cuckoo Sandbox та ін.) Трасування системних викликів (Process Monitor, strace тощо). Використання відлагоджувачів (на прикл. IDA Free та ін.)	лекція, лаб., самостійна робота	[1, 3, 4, 7]	2 2 8	2 тижні
12-14	**Вдосконалення майстерності реверс-інженера**. Пошук криптографічних операцій. Використання ШІ для дослідження підозрілого ПЗ (на прикл. BinaryAI та ін.) (Де)компресія коду. (Де)обфускація коду. Захист від відлагодження. Поліморфні генератори. Приховування в системі.	лекція, лаб., самостійна робота	[1, 3, 6, 8]	3 3 9	3 тижні
15-16	**Інші платформи розповсюдження шкідливого ПЗ**. Скриптові мови. Дослідження та декомпіляція .NET, Java та мобільних застосунків. Розвідка загроз.	лекція, лаб., самостійна робота	[12-14]	2 2 8	2 тижні