

Державний заклад
«ПІВДЕННОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ
ПЕДАГОГІЧНИЙ УНІВЕРСИТЕТ
імені К. Д. УШИНСЬКОГО»



ОДЕСЬКИЙ
НАЦІОНАЛЬНИЙ
УНІВЕРСИТЕТ імені І. І. МЕЧНИКОВА

ДВАДЦЯТЬ ДРУГА ВСЕУКРАЇНСЬКА КОНФЕРЕНЦІЯ
СТУДЕНТІВ І МОЛОДИХ НАУКОВЦІВ

ІНФОРМАТИКА, ІНФОРМАЦІЙНІ СИСТЕМИ ТА ТЕХНОЛОГІЇ

25 квітня 2025 р.

Одеса – 2025

Інформатика, інформаційні системи та технології: тези доповідей двадцять другої всеукраїнської конференції студентів і молодих науковців. Одеса, 25 квітня 2025 р. - Одеса, 2025. – 315 с.

Друкується за рішенням Вченої Ради

Університету Ушинського

(протокол № 16 від 29.05.2025 р.)

Організатори конференції продовжують традицію обміну досвідом у сфері освіти та використання інформаційних технологій. У конференції приймають участь студенти, аспіранти та молоді науковці вищих навчальних закладів України.

Тематика конференції охоплює наступне коло питань: сучасні інформаційні технології; інтелектуальні системи; методика викладання інформатики; інформаційні технології в освіті; психолого-педагогічне забезпечення інформатизації навчальної діяльності; дистанційна освіта і глобальні телекомунікаційні мережі; математичне моделювання й інформаційні технології; інформатизація системи керування освітою; інформаційні технології в менеджменті.

Наукові керівники:

завідувачка кафедри прикладної математики та інформатики

фізико-математичного факультету Університету Ушинського, д. т. н., проф. Т. Л. Мазурок,

завідувач кафедри математичного забезпечення комп'ютерних систем факультету математики, фізики та інформаційних технологій ОНУ імені І. І. Мечникова, д. т. н., проф. Є. В. Малахов

Оргкомітет:

Голова:

Ректор Університету Ушинського,

д. і. наук, доц. А. В. Красножон

Заступники голови:

Проректор з наукової роботи Університету Ушинського, д. політ. н., проф. Г.В. Музиченко

Декан факультету математики, фізики та інформаційних технологій ОНУ імені І. І. Мечникова,

д. ф-м. н., проф. Ю. А. Ніщук

Члени оргкомітету:

д. т. н., проф.	Є. В. Малахов	д. т. н., проф.	Т. Л. Мазурок
д. т. н., проф.	Ю. О. Гунченко	к. п. н., доц.	А. О. Яновський
ст. викладач	І. М. Лісіцина	викладач	О. Я. Рубанська
ст. викладач	Н. Ф. Трубіна	к. ф.-м. н.	О. П. Бойко
ст. викладач	В. А. Корабльов	PhD, associated prof. (Poland)	A. Rychlik

© Навчально-науковий інститут природничо-математичних наук, інформатики та менеджменту Державного закладу «Південноукраїнський національний педагогічний університет імені К. Д. Ушинського», кафедра прикладної математики та інформатики, 2025

© Факультет математики, фізики та інформаційних технологій Одеського національного університету імені І. І. Мечникова, кафедра математичного забезпечення комп'ютерних систем, 2025

ВИКОРИСТАННЯ ІНКРЕМЕНТАЛЬНО ПІДТРИМУВАНИХ МАТЕРІАЛІЗОВАНИХ ПОДАЇВ У КОРПОРАТИВНИХ ЗАСТОСУНКАХ.....	239
Пасенченко Т. О., Гунченко Ю. О.	239
АРХІТЕКТУРА ІТ-РІШЕННЯ ЗАБЕЗПЕЧЕННЯ КРОВ'Ю ВІЙСЬКОВИХ МЕДИЧНИХ ШПИТАЛІВ.....	240
Кашуба М. Д., Чиркова К. С.	240
КОНЦЕПЦІЯ АВТОМАТИЗАЦІЇ УПРАВЛІННЯ ЗАПАСАМИ КОМПОНЕНТІВ КРОВІ	244
Чиркова К. С.	244
ДОСЛІДЖЕННЯ МЕТОДІВ І АЛГОРИТМІВ МАШИННОГО НАВЧАННЯ У СИСТЕМАХ REAL-TIME BIDDING.....	246
Іванов О. О., Мартинович Л. Я.	246
TEACHERDESMOS ЯК ІНСТРУМЕНТ АКТИВНОГО НАВЧАННЯ: СТВОРЕННЯ ІНТЕРАКТИВНИХ УРОКІВ З МАТЕМАТИКИ	248
Лобушко М. Є., Рикова Л. Л.	248
ОГЛЯД ТЕХНОЛОГІЙ РОЗРОБКИ СИСТЕМИ ВИЯВЛЕННЯ АНОМАЛІЙ У КОМП'ЮТЕРНИХ МЕРЕЖАХ	250
Свиридов І. І., Шпинарева І. М.	250
АНАЛІЗ ПРОБЛЕМИ ВИСОКОГО РІВНЯ ХИБНОПОЗИТИВНИХ СПРАЦЮВАНЬ У СИСТЕМАХ ВИЯВЛЕННЯ МЕРЕЖЕВИХ АНОМАЛІЙ	252
Свиридов І. І., Шпинарева І. М.	252
ІННОВАЦІЙНІ ТЕХНОЛОГІЇ В РОЗРОБЦІ МАСШТАБОВАНИХ РЕКОМЕНДАЦІЙНИХ ВЕБ-ПЛАТФОРМ	254
Привалов А. Г., Рудніченко М. Д.	254
ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ МОДЕЛЮВАННЯ ЗБУРЕНОГО РУХУ ТВЕРДОГО ТІЛА У СЕРЕДОВИЩІ З ОПОРОМ	256
Явдошук І. С., Рачинська А. Л.	256
КОМП'ЮТЕРНЕ МОДЕЛЮВАННЯ РУХУ СУПУТНИКА-ГІРОСТАТА З ПОРОЖНИНАМИ	257
Кобзар К. В., Рачинська А. Л.	257
ІНФОРМАЦІЙНА СИСТЕМА ПРОГНОЗУВАННЯ ПОПИТУ НА МЕДИЧНІ СТРАХОВКИ СЕРЕД ЛІКАРІВ	258
Федорова К. А.	258
ПРОГРАМУВАННЯ "РОЗУМНОГО АКВАРІУМА" З ВИКОРИСТАННЯМ ARDUINO ЯК ЗАСОБУ НАВЧАННЯ ІНФОРМАТИКИ.....	260
Реулець М., Корабльов В. А.	260
ЦИВІЛЬНЕ ЗАСТОСУВАННЯ ДРОНІВ У СФЕРІ ОСВІТИ: МІЖНАРОДНИЙ ДОСВІД ТА УКРАЇНСЬКІ РЕАЛІЇ.....	262
Ковальчук Б., Корабльов В. А.	262

Якщо гравець, обравши правильні двері, змінить свій вибір – він програє, і навпаки. Тому можна сказати, що якщо не змінити вибір – вірогідність виграти $1/3$, а якщо змінити – $2/3$.

Це легше пояснити розглянувши наступну таблицю (вважається, що гравець у всіх 3 випадках обирає двері № 1):

Двері № 1	Двері № 2	Двері № 3	Номер двері, котру відкрив ведучий	Результат, якщо змінити вибір	Результат, якщо не змінити вибір
Авто	Коза	Коза	2 або 3	Коза	Авто
Коза	Авто	Коза	3	Авто	Коза
Коза	Коза	Авто	2	Авто	Коза

Цей парадокс демонструє, що наша інтуїція може помилятися щодо ймовірностей, і викликав бурхливі дискусії, оскільки багато людей, навіть математиків, не могли повірити в правильність цього рішення.

Робота демонструє, що теорія ймовірностей – це не лише сукупність абстрактних формул, але й потужний інструмент для аналізу реальних життєвих ситуацій. Завдяки використанню практичних прикладів, таких як гральні карти та парадокс Монті Голла, можна значно підвищити інтерес учнів до цієї дисципліни. Важливо враховувати вікові особливості школярів, пропонуючи їм завдання, які відповідають їхнім інтересам та рівню розвитку.

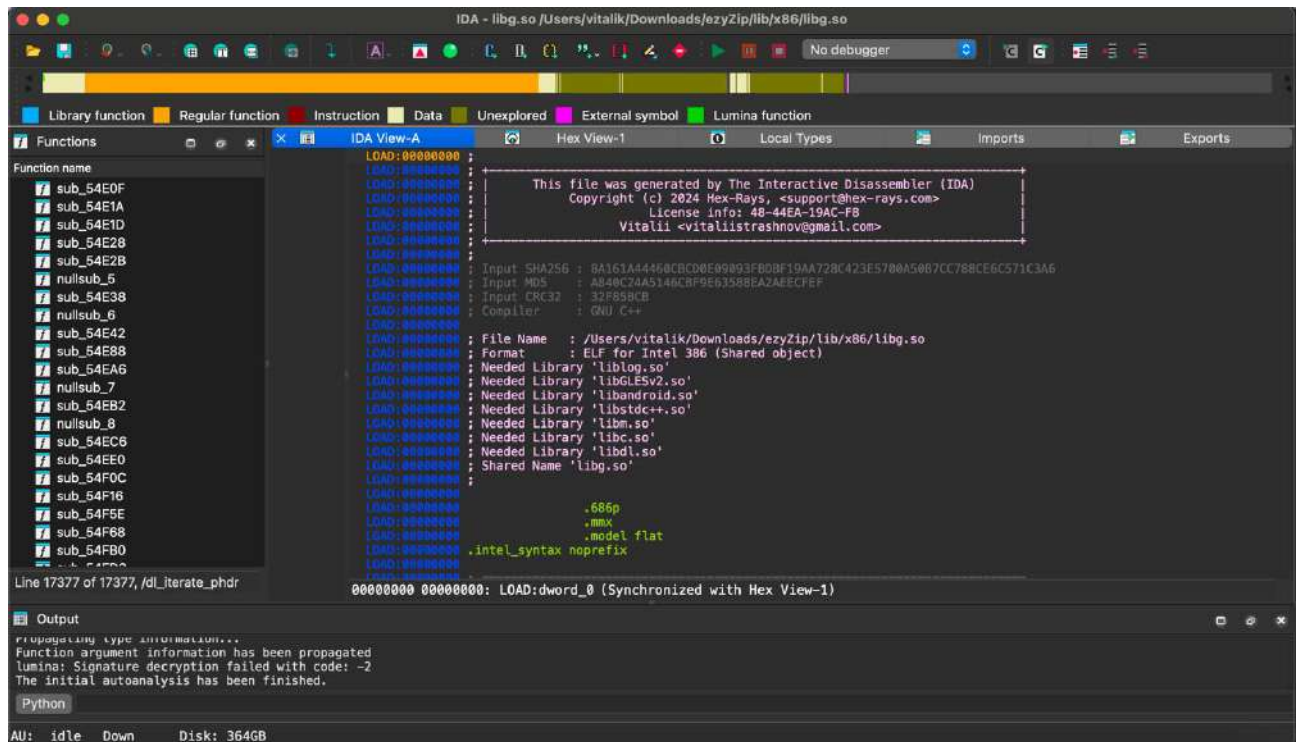
EMULATION OF GAME SERVER LOGIC THROUGH DYNAMIC BINARY MODULE LOADING

Strashnov V., Gunchenko Y.

Odesa I.I. Mechnikov National University

Keywords: dynamic library, simulation, reverse engineering, Rust, reverse-engineering.

In the context of closed-source client-server games, an important task is the ability to study and replicate their logic for the purposes of analysis, testing, or developing a compatible server environment. This paper presents an approach to emulating game logic implemented as a dynamic library (libg.so) by calling its internal functions from custom code written in the Rust programming language.



The game we've been exploring here is implemented as an Android application, with its logic partially or entirely located in a .so file—a native library likely compiled from C or C++. The contents of the .so file were analyzed using the IDA Pro disassembler (as shown in the figure), which made it possible to identify key functions responsible for calculating game state and generating packets sent to the server. The analysis revealed that the game uses a “mirror server” model—meaning the entire game logic is repeated on the server side, while the client only sends checksums of the game state.

```
pub fn get_module_base(shared_object_name: &str) -> Option<usize> {
    let path = "/proc/self/maps";
    let file = File::open(path).ok()?;
    let reader = io::BufReader::new(file);

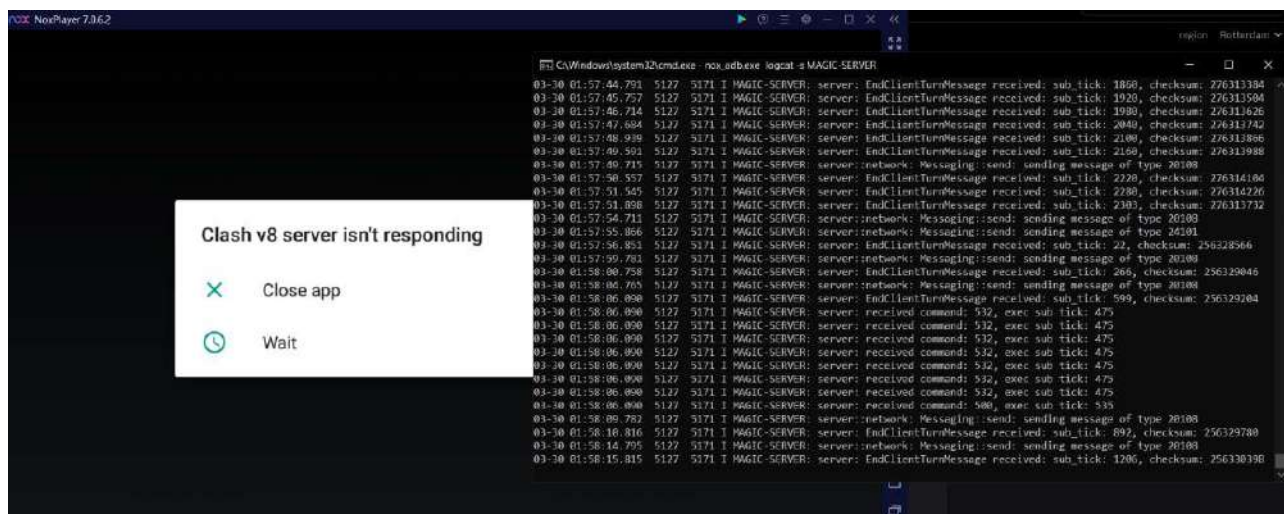
    for line in reader.lines() {
        let line = line.ok()?;
        if line.contains(shared_object_name) {
            let address_str = line.split_whitespace().next().unwrap_or("");
            let address = usize::from_str_radix(&address_str.split('-').next().unwrap_or(""), 16)
                .ok()?;
            return Some(address);
        }
    }

    None
}
```

To simulate the server side, a library named libserver.so was created, which is loaded alongside libg.so. The load address of libg.so is read via the /proc/self/maps filesystem. Using the relative function offsets obtained from IDA Pro, the absolute addresses of the functions are calculated and invoked from Rust using FFI (Foreign Function Interface).

Before calling each function, the necessary memory is allocated using malloc, and then filled with the required values—determined based on disassembly analysis. After invoking a function, its result is passed to the next one, forming a complete chain of game logic execution. In this way, a server logic simulator has been implemented that functions as a "black box": it does not require understanding of the internal game logic but replicates its behavior through direct function calls to the original .so file.

A key advantage of this approach is the ability to precisely replicate the game logic even without access to the original source code. However, it should be noted that this method depends on the stability of function addresses, which may change with game updates or changes in the compilation architecture.



The figure shows the server structure running the game through NOX to emulate libg.so. One can observe how the server receives and processes requests from the client.

The developed approach can be useful for creating tools to analyze game behavior, study protection mechanisms, build standalone servers or bots, as well as for testing game logic without connecting to the original server.

References

1. Hex-Rays docs URL: <https://docs.hex-rays.com/> (accessed: 20.03.2025).
2. The Rust Programming Language URL: <https://doc.rust-lang.org/book/> (accessed: 20.03.2025).
3. <https://github.com/VitalikObject/magic-rs/> - source code of following program

МЕТОДИЧНІ ОСОБЛИВОСТІ ВПРОВАДЖЕННЯ ЕЛЕМЕНТІВ СТЕМ-ОСВІТИ НА УРОКАХ ІНФОРМАТИКИ

Богданова Т. А., Мазурок Т. Л.

ПНПУ ім. К.Д. Ушинського, м. Одеса

СТЕМ-освіта є одним з пріоритетних напрямків, розвиток якого пов'язаний з реформуванням освіти на основі впровадження основних принципів Концепції

Державний заклад
«ПІВДЕННОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ
ПЕДАГОГІЧНИЙ УНІВЕРСИТЕТ
імені К. Д. УШИНСЬКОГО»



ОДЕСЬКИЙ
НАЦІОНАЛЬНИЙ
УНІВЕРСИТЕТ імені І. І. МЕЧНИКОВА

ДВАДЦЯТЬ ДРУГА ВСЕУКРАЇНСЬКА КОНФЕРЕНЦІЯ
СТУДЕНТІВ І МОЛОДИХ НАУКОВЦІВ

ІНФОРМАТИКА, ІНФОРМАЦІЙНІ
СИСТЕМИ ТА ТЕХНОЛОГІЇ

Збірник робіт

Збірник робіт надрукований в авторській редакції
без внесення суттєвих змін оргкомітетом

Підписано до друку 25.04.2025
Здано у виробництво 25.04.2025
Формат 60x84/16. Папір офсетний. Друк офсетний.
Тираж 50 примірників

Надруковано з готового оригінал-макета