НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ім. Ігоря СІКОРСЬКОГО»

НАВЧАЛЬНО-НАУКОВИЙ ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ

Курсова робота

з дисциплiни «Симетрична криптографiя»

**Блоковий шифр MISTY1**

Виконав студент

Групи Фі-23

Баранов Глєб

**ЗМІСТ**

[ВСТУП 3](#_Toc201855814)

[1. ОПИС АЛГОРИТМУ ШИФРУВАННЯ 4](#_Toc201855815)

[1.1 Основні параметри шифру 4](#_Toc201855816)

[1.2 Загальна структура MISTY1 5](#_Toc201855817)

[1.3 FL-функція (лінійні перетворення) 6](#_Toc201855818)

[1.4 FO-функція (зовнішня функція Фейстеля) 7](#_Toc201855819)

[1.5 FI-функція (внутрішня функція Фейстеля) 8](#_Toc201855820)

[1.6 S-блоки S7 та S9 10](#_Toc201855821)

[1.7 Схема генерування підключів 11](#_Toc201855822)

[1.8 Алгоритм шифрування 12](#_Toc201855823)

[1.9 Алгоритм дешифрування 13](#_Toc201855824)

[2. ОБҐРУНТУВАННЯ СТІЙКОСТІ ТА ЕФЕКТИВНОСТІ АЛГОРИТМУ 16](#_Toc201855825)

[2.1 Порівняння з класичною схемою Фейстеля 16](#_Toc201855826)

[2.2 Дизайнерські рішення та їх обґрунтування 17](#_Toc201855827)

[2.3 Оцінка ефективності реалізації 18](#_Toc201855828)

[3. РЕФЕРАТИВНИЙ ОГЛЯД НАУКОВИХ ДЖЕРЕЛ 21](#_Toc201855829)

[3.1 Оригінальна публікація та базовий опис 21](#_Toc201855830)

[3.2 Криптоаналіз та атаки 22](#_Toc201855831)

[3.3 Практичні реалізації та оптимізації 22](#_Toc201855832)

[3.4 Стандартизація та впровадження 23](#_Toc201855833)

[3.5 Підсумок огляду 24](#_Toc201855834)

[4. ПРОГРАМНА РЕАЛІЗАЦІЯ 26](#_Toc201855835)

[4.1 Вибір мови програмування 26](#_Toc201855836)

[4.2 Особливості імплементації 27](#_Toc201855837)

[4.3 Опис функцій реалізації 28](#_Toc201855838)

[4.4 Особливості дизайну програми-шифратора 29](#_Toc201855839)

[4.5 Демонстрація коректності реалізації 30](#_Toc201855840)

[ВИСНОВКИ 32](#_Toc201855841)

[ЛІТЕРАТУРА 34](#_Toc201855842)

**ВСТУП**

У сучасному світі інформаційних технологій питання захисту даних набуває критичного значення. Симетрична криптографія, зокрема блочні шифри, відіграє ключову роль у забезпеченні конфіденційності та цілісності інформації. MISTY1, розроблений у 1997 році японськими криптографами під керівництвом Міцуру Мацуї, представляє собою унікальний приклад блочного шифру з доведуваною безпекою проти диференціального та лінійного криптоаналізу.

Актуальність дослідження MISTY1 обумовлена його інноваційною рекурсивною архітектурою, міжнародною стандартизацією (ISO/IEC 18033-3, проект NESSIE, CRYPTREC) та значним впливом на розвиток сучасної криптографії. Алгоритм демонструє прогресивні підходи до побудови криптографічно стійких систем, включаючи математично обґрунтовані S-блоки на основі скінченних полів.

Мета роботи. Комплексне дослідження блочного симетричного шифру MISTY1, включаючи аналіз архітектури, криптографічних властивостей та практичну реалізацію.

Завдання:

1. Дослідити структуру та принципи роботи MISTY1
2. Проаналізувати криптографічні властивості та стійкість алгоритму
3. Провести порівняльний аналіз з класичними схемами Фейстеля
4. Здійснити огляд наукових досліджень з криптоаналізу MISTY1
5. Розробити програмну реалізацію з демонстрацією коректності

**1. ОПИС АЛГОРИТМУ ШИФРУВАННЯ**

**1.1 Основні параметри шифру**

MISTY1 є блочним симетричним шифром з фіксованими параметрами, оптимізованими для забезпечення високого рівня безпеки при ефективній реалізації. Розмір блоку становить 64 біти, що є стандартним для багатьох класичних блочних шифрів і забезпечує баланс між безпекою та продуктивністю. Така довжина блоку дозволяє ефективно обробляти дані на більшості сучасних архітектур процесорів, особливо на 64-бітних системах.

Довжина ключа шифрування в MISTY1 складає 128 бітів, що відповідає сучасним вимогам криптографічної стійкості. 128-бітний ключ забезпечує криптографічну стійкість на рівні операцій для атак грубої сили, що є достатнім для більшості практичних застосувань. Порівняно з деякими іншими алгоритмами того часу, які використовували менші довжини ключів, MISTY1 забезпечує вищий рівень безпеки завдяки більшому простору ключів.

Кількість раундів у MISTY1 становить 8, що є відносно невеликим числом порівняно з багатьма іншими блочними шифрами. Однак ця особливість компенсується складною внутрішньою структурою кожного раунду. Кожен раунд містить декілька рівнів перетворень, включаючи FL-функції та FO-функції, що забезпечує високу криптографічну стійкість навіть при невеликій кількості раундів. Така архітектура дозволяє досягти оптимального співвідношення між безпекою та швидкістю роботи [1].

Внутрішня структура MISTY1 базується на принципі рекурсивної побудови, де основні функції самі містять структури Фейстеля меншого розміру. Це створює багаторівневий захист, де навіть компрометація одного рівня не призводить до повного зламу системи. FO-функція, яка є основною компонентою раунду, сама реалізована як 3-раундова мережа Фейстеля, що значно підвищує складність криптоаналізу та робить шифр стійким до широкого спектру атак.

Додатковою особливістю є використання FL-функцій, які виконують лінійні перетворення між основними раундами. Ці функції забезпечують додаткову дифузію та змішування, що ускладнює побудову диференціальних та лінійних характеристик. FL-функції також сприяють руйнуванню можливих закономірностей, які могли б виникнути через регулярну структуру FO-функцій, що підвищує загальну криптографічну стійкість алгоритму.

**1.2 Загальна структура MISTY1**

Архітектура MISTY1 представляє собою унікальну модифікацію класичної схеми Фейстеля, яка отримала назву "MISTY-структура". Основною відмінністю від традиційної схеми Фейстеля є використання рекурсивного підходу, де функція раунду сама містить менші структури Фейстеля. Ця ієрархічна побудова створює багаторівневий захист, де кожен рівень додає свій внесок у загальну криптографічну стійкість системи.

Загальна схема шифру складається з восьми основних раундів, між якими розташовані FL-функції. FL-функції застосовуються після раундів 2, 4 та 6, а також перед першим та після останнього раунду в деяких варіантах реалізації. Така нерегулярна структура розміщення FL-функцій ускладнює криптоаналіз та перешкоджає поширенню характеристик через весь шифр. FL-функції виконують роль додаткових "перемішуючих" елементів, які руйнують можливі закономірності в розповсюдженні зміни бітів.

Кожен основний раунд MISTY1 реалізований за допомогою FO-функції, яка сама є 3-раундовою мережею Фейстеля, що працює з 32-бітними половинами. Така внутрішня структура значно збільшує кількість нелінійних перетворень, які проходять дані, не збільшуючи при цьому загальну кількість раундів. FO-функція використовує FI-функції як свої F-функції, створюючи другий рівень рекурсії в архітектурі шифру.

Розподіл даних у MISTY1 відбувається за принципом збалансованої схеми Фейстеля, де 64-бітний блок ділиться на дві 32-бітні половини. Ліва половина підлягає перетворенню за допомогою FO-функції з використанням відповідних підключів, після чого результат додається (XOR) до правої половини. Потім половини міняються місцями, і процес повторюється для наступного раунду. Такий підхід забезпечує повну дифузію даних через весь блок протягом декількох раундів.

Важливою особливістю структури MISTY1 є асиметричність у застосуванні FL-функцій. Ці функції не застосовуються після кожного раунду, а лише в певних позиціях, що створює нерегулярність у структурі шифру. Така асиметричність ускладнює побудову ефективних атак, оскільки криптоаналітикам доводиться враховувати різні типи перетворень у різних частинах шифру. Це рішення було прийнято на основі математичного аналізу стійкості до диференціального та лінійного криптоаналізу [2].

**1.3 FL-функція (лінійні перетворення)**

FL-функція в архітектурі MISTY1 виконує роль додаткового лінійного перетворення, яке забезпечує підвищену дифузію та ускладнює криптоаналіз. Ця функція працює з 32-бітним входом, який розділяється на дві 16-бітні частини: старшу () та молодшу (). Кожна з цих частин обробляється незалежно з використанням відповідних підключів, що забезпечує збалансовану обробку всіх бітів входу. FL-функція є оборотною, що критично важливо для процесу дешифрування.

Математичний опис FL-функції включає дві основні операції: побітове логічне І (AND) та виключне АБО (XOR). Для лівої частини виконується операція , після чого результат піддається операції XOR з правою частиною: . Аналогічно, для правої частини спочатку виконується операція , а потім . Такий підхід забезпечує взаємну залежність між лівою та правою частинами, що підвищує дифузійні властивості функції.

Вибір операцій AND та OR для FL-функції не є випадковим і базується на математичному аналізі їх впливу на криптографічні властивості. Операція AND сприяє створенню залежностей між бітами, тоді як операція OR забезпечує поширення змін. У поєднанні з операціями XOR, які зберігають лінійність, ці операції створюють складну нелінійну залежність, яка ускладнює побудову лінійних та диференціальних характеристик для всього шифру.

Підключі для FL-функції () генеруються з основного ключа за спеціальним алгоритмом, який забезпечує їх незалежність та рівномірний розподіл. Кожна FL-функція використовує унікальні підключі, що запобігає виникненню циклічних залежностей та підвищує загальну стійкість шифру. Довжина кожного підключа становить 16 бітів, що відповідає розміру половин, з якими працює функція.

Розташування FL-функцій у загальній структурі MISTY1 є стратегічно обґрунтованим. Вони розміщуються після раундів 2, 4 та 6, що забезпечує рівномірний розподіл додаткових перетворень по всій довжині шифру. Така схема перешкоджає поширенню характеристик атак через весь шифр та створює додаткові перешкоди для криптоаналізу. FL-функції також сприяють руйнуванню можливих слабких ключів та зменшують ймовірність існування пов'язаних ключів з небажаними властивостями.

**1.4 FO-функція (зовнішня функція Фейстеля)**

FO-функція є серцем архітектури MISTY1 та представляє собою 3-раундову мережу Фейстеля, яка працює з 32-бітними блоками даних. Ця функція реалізує основні нелінійні перетворення в кожному раунді шифру і є критично важливою для забезпечення криптографічної стійкості. Входом FO-функції є 32-бітний блок, який розділяється на дві 16-бітні половини: (ліва) та (права). Кожен з трьох внутрішніх раундів FO-функції використовує FI-функцію як свою F-функцію, створюючи другий рівень рекурсії в архітектурі.

Структура FO-функції побудована за класичним принципом Фейстеля: ліва половина обробляється за допомогою F-функції (в даному випадку FI-функції) з відповідним підключем, результат додається (XOR) до правої половини, після чого половини міняються місцями. Цей процес повторюється три рази, створюючи складну нелінійну залежність між входом та виходом. Використання трьох раундів у FO-функції забезпечує достатню дифузію для 32-бітного блоку та гарантує, що зміна будь-якого біта входу вплине на всі біти виходу.

Математичний опис роботи FO-функції включає наступні кроки для кожного з трьох раундів: спочатку ліва половина подається на вход FI-функції разом з відповідним підключем , потім результат додається (XOR) до правої половини , після чого половини міняються місцями (крім останнього раунду). Така послідовність операцій забезпечує повну дифузію даних та створює складні нелінійні залежності, які є основою криптографічної стійкості MISTY1.

Підключі для FO-функції () мають довжину 16 бітів і генеруються з основного 128-бітного ключа за спеціальним алгоритмом розгортання ключа. Кожен раунд FO-функції використовує унікальний підключ, що забезпечує незалежність перетворень у різних раундах та перешкоджає виникненню небажаних симетрій. Загалом, для повного шифрування одного блоку MISTY1 потребує 24 підключі типу KO (по 3 для кожної з 8 FO-функцій), що підкреслює складність ключового розкладу.

Важливою особливістю FO-функції є її збалансованість та оборотність. Збалансованість означає, що кожен біт виходу залежить від кожного біта входу з приблизно однаковою ймовірністю, що забезпечує хорошу дифузію. Оборотність є критично важливою для процесу дешифрування: для інвертування FO-функції достатньо виконати ті самі операції в зворотному порядку з тими самими підключами, що робить процес дешифрування таким же ефективним, як і шифрування. Ця властивість значно спрощує імплементацію MISTY1 в апаратних та програмних системах [3].

**1.5 FI-функція (внутрішня функція Фейстеля)**

FI-функція представляє найглибший рівень рекурсії в архітектурі MISTY1 і є основним джерелом нелінійності в алгоритмі. Ця функція реалізована як 3-раундова мережа Фейстеля, яка працює з 16-бітними блоками даних, розділеними на дві нерівні частини: 9-бітну ліву () та 7-бітну праву (). Така асиметрична структура є унікальною особливістю MISTY1 та дозволяє оптимально використовувати властивості S-блоків різних розмірів. FI-функція використовує S-блоки S9 та S7 як свої основні нелінійні компоненти.

Робочий цикл FI-функції складається з трьох раундів, кожен з яких включає застосування відповідного S-блоку та операції XOR з підключем. У першому раунді 9-бітна ліва частина обробляється S-блоком S9, результат додається (XOR) до 7-бітної правої частини, яка потім розширюється до 9 бітів. У другому раунді 7-бітна права частина (тепер ліва після перестановки) обробляється S-блоком S7, результат додається до 9-бітної частини. Третій раунд повторює операції першого раунду з іншим підключем.

Математичний опис FI-функції демонструє складну взаємодію між компонентами різних розмірів. Для першого раунду: де результат S9 усікається до 7 бітів. Для другого раунду: , де результат S7 розширюється до 9 бітів додаванням двох нульових бітів. Третій раунд повторює структуру першого з іншим підключем . Така схема забезпечує повну дифузію всіх бітів входу на всі біти виходу протягом трьох раундів.

Асиметрична структура FI-функції з 9-бітними та 7-бітними частинами не є випадковою і базується на математичних властивостях S-блоків S9 та S7. Ці розміри були обрані таким чином, щоб максимізувати криптографічні властивості S-блоків, зокрема їх нелінійність та стійкість до диференціального криптоаналізу. Використання S-блоків різних розмірів також ускладнює побудову узагальнених атак, оскільки криптоаналітикам доводиться враховувати різні математичні структури одночасно.

Підключі для FI-функції () мають довжини 7 та 9 бітів відповідно, що точно відповідає розмірам частин, з якими вони використовуються. Ця точна відповідність розмірів забезпечує оптимальне використання ентропії ключа та запобігає марнуванню бітів. Генерація цих підключів здійснюється з основного ключа таким чином, щоб забезпечити їх незалежність та рівномірний розподіл [4]. Загальна кількість підключів типу KI, необхідних для повного шифрування, становить 48 (по 6 для кожної з 8 FO-функцій), що підкреслює складність внутрішньої структури MISTY1.

**1.6 S-блоки S7 та S9**

S-блоки S7 та S9 є фундаментальними компонентами MISTY1, які забезпечують основну нелінійність алгоритму та його стійкість до криптоаналітичних атак. Ці S-блоки мають розміри 7×7 та 9×9 бітів відповідно, що робить їх унікальними серед більшості блочних шифрів, які зазвичай використовують S-блоки розміром 8×8 бітів. Вибір саме таких розмірів обумовлений математичними вимогами до оптимальних криптографічних властивостей та ефективності реалізації в структурі FI-функції.

Найважливішою особливістю S-блоків MISTY1 є їх математична побудова на основі обчислень у скінченних полях. S-блок S7 базується на обчисленнях у полі GF(), тоді як S9 використовує поле GF(). Така побудова забезпечує доведені математичні властивості, включаючи максимальну нелінійність, оптимальну стійкість до диференціального криптоаналізу та відсутність слабких входів. На відміну від багатьох інших шифрів, де S-блоки генеруються випадково або за евристичними критеріями, S-блоки MISTY1 мають строге математичне обґрунтування.

Криптографічні властивості S-блоків включають високу нелінійність, що означає максимальну відстань від найближчої лінійної функції. Для S7 нелінійність становить 56, що є максимально можливим значенням для 7-бітного S-блоку. Аналогічно, S9 має нелінійність 240, що також є оптимальним. Ці значення гарантують, що будь-яка спроба апроксимації S-блоків лінійними функціями буде мати значну похибку, що робить лінійний криптоаналіз неефективним.

Диференціальні властивості S-блоків також є оптимальними. Максимальна диференціальна ймовірність для S7 становить , а для S9 – , що є найнижчими можливими значеннями для S-блоків таких розмірів. Це означає, що для будь-якої ненульової різниці входів ймовірність отримання будь-якої конкретної різниці виходів не перевищує зазначених значень. Такі властивості роблять диференціальний криптоаналіз MISTY1 вкрай складним, оскільки ймовірності диференціальних характеристик швидко зменшуються з збільшенням кількості раундів.

Практична реалізація S-блоків може здійснюватися як за допомогою попередньо обчислених таблиць, так і через обчислення в реальному часі з використанням арифметики скінченних полів. Табличний підхід забезпечує максимальну швидкість, але вимагає додаткової пам'яті: 128 байт для S7 та 1024 байти для S9. Обчислювальний підхід економить пам'ять, але може бути повільнішим на деяких платформах [5]. Вибір методу реалізації залежить від конкретних вимог застосування та доступних ресурсів системи.

**1.7 Схема генерування підключів**

Схема генерування підключів у MISTY1 є складним та ретельно розробленим процесом, який забезпечує створення всіх необхідних підключів з одного 128-бітного основного ключа. Загальна кількість підключів, необхідних для повного процесу шифрування, є значною: 32 підключі KL для FL-функцій, 24 підключі KO для FO-функцій та 48 підключів KI для FI-функцій, що разом становить 104 підключі різних розмірів. Така велика кількість підключів забезпечує високу криптографічну стійкість та перешкоджає атакам на ключовий розклад.

Процес генерування підключів починається з розділення 128-бітного основного ключа K на вісім 16-бітних слів K1, K2, ..., K8. Ці слова служать основою для генерування всіх інших підключів через систему лінійних перетворень та перестановок. Основним принципом є забезпечення того, щоб кожен підключ залежав від якомога більшої кількості бітів основного ключа, що запобігає виникненню слабких ключів та підвищує стійкість до атак на ключовий розклад.

Генерування підключів KL для FL-функцій здійснюється через просте копіювання або перестановку основних слів ключа. Наприклад, KL1 = K1, KL2 = K3, тощо. Така схема забезпечує швидкість генерування та економність використання основного ключового матеріалу. Хоча ця частина схеми може здатися простою, вона ретельно розроблена таким чином, щоб FL-функції отримували різноманітні підключі, які не створюють небажаних залежностей між різними частинами шифру.

Підключі KO для FO-функцій генеруються через більш складну схему, яка включає циклічні зсуви та комбінування різних слів основного ключа. Наприклад, деякі підключі KO можуть бути сформовані як - одне з основних слів, а n - кількість позицій для циклічного зсуву. Інші підключі можуть бути результатом XOR-операції між різними словами або їх зсунутими версіями. Така схема забезпечує високу дифузію ключового матеріалу.

Найскладнішими є підключі KI для FI-функцій, які мають нестандартні розміри 7 та 9 бітів. Ці підключі генеруються шляхом витягування відповідних бітів з 16-бітних слів основного ключа або їх модифікованих версій. Процес включає розділення 16-бітних слів на частини потрібних розмірів, при цьому забезпечується, щоб кожен біт основного ключа впливав на якомога більше підключів KI. Така схема максимізує ентропію підключів та мінімізує ймовірність існування пов'язаних ключів з небажаними властивостями [6].

**1.8 Алгоритм шифрування**

Алгоритм шифрування MISTY1 представляє собою ретельно організовану послідовність операцій, яка перетворює 64-бітний блок відкритого тексту в 64-бітний блок шифротексту з використанням 128-бітного ключа. Процес шифрування складається з восьми основних раундів, між якими стратегічно розміщені FL-функції для додаткової дифузії. Кожен раунд використовує FO-функцію як основний нелінійний компонент, створюючи багаторівневу структуру перетворень, яка забезпечує високу криптографічну стійкість.

Початкова фаза алгоритму включає розділення 64-бітного вхідного блоку на дві 32-бітні половини: L0 (ліва) та R0 (права). Ці половини служать початковими значеннями для ітеративного процесу Фейстеля. Перед початком основних раундів може застосовуватися початкова FL-функція, залежно від конкретної варіації алгоритму. Така початкова обробка забезпечує додаткове змішування вхідних даних з ключовим матеріалом та ускладнює криптоаналіз.

Основний цикл шифрування виконується протягом восьми раундів, кожен з яких реалізує модифіковану схему Фейстеля. У i-му раунді ліва половина Li-1 обробляється FO-функцією з використанням відповідних підключів, результат додається (XOR) до правої половини Ri-1, формуючи нову ліву половину Li. Права половина Ri просто копіюється з попередньої лівої половини Li-1. Математично це виражається як: представляє набір підключів для i-го раунду.

FL-функції застосовуються після раундів 2, 4 та 6, створюючи нерегулярну структуру, яка ускладнює побудову атак. Коли FL-функція застосовується до половини блоку, вона виконує додаткове лінійне перетворення з використанням відповідних підключів KL. Така нерегулярність у застосуванні FL-функцій є ключовою особливістю MISTY1, яка відрізняє його від більшості інших схем Фейстеля. Це рішення базується на математичному аналізі стійкості та забезпечує оптимальний баланс між безпекою та ефективністю.

Завершальна фаза шифрування може включати фінальну FL-функцію та остаточну перестановку половин, залежно від конкретної реалізації. Результатом є 64-бітний блок шифротексту, який формується конкатенацією фінальних значень лівої та правої половин. Важливо відзначити, що MISTY1 використовує збалансовану структуру Фейстеля, що означає рівну обробку обох половин блоку протягом всього процесу шифрування, забезпечуючи оптимальну дифузію та стійкість до різних типів атак [7].

**1.9 Алгоритм дешифрування**

Алгоритм дешифрування MISTY1 є майже ідентичним до алгоритму шифрування завдяки властивостям оборотності схеми Фейстеля та всіх компонентних функцій. Основна відмінність полягає у зворотному порядку застосування раундів та підключів, що дозволяє відновити початковий відкритий текст з шифротексту. Ця симетричність значно спрощує імплементацію MISTY1, оскільки одна й та сама архітектура може використовуватися як для шифрування, так і для дешифрування з мінімальними модифікаціями.

Процес дешифрування починається з розділення 64-бітного блоку шифротексту на дві 32-бітні половини, аналогічно до процесу шифрування. Якщо при шифруванні застосовувалася фінальна FL-функція, то при дешифруванні спочатку застосовується обернена FL-функція з тими самими підключами. Оберненність FL-функції забезпечується її математичною структурою: операції AND та OR з наступними XOR можуть бути відмінені шляхом виконання тих самих операцій у зворотному порядку.

Основний цикл дешифрування виконується протягом восьми раундів у зворотному порядку, починаючи з 8-го раунду і закінчуючи 1-м. У кожному раунді права половина обробляється тією самою FO-функцією з тими самими підключами, що використовувалися при шифруванні, результат додається (XOR) до лівої половини, після чого половини міняються місцями. Математично для i-го раунду дешифрування: , де штрих позначає значення при дешифруванні.

FL-функції при дешифруванні застосовуються у зворотному порядку: після раундів 6, 4 та 2 (якщо рахувати від кінця). Кожна FL-функція інвертується шляхом виконання її операцій у зворотному порядку з тими самими підключами. Спочатку виконується обернена операція для другої частини функції, потім для першої. Така схема забезпечує точне відновлення значень, які були до застосування FL-функції при шифруванні.

Завершення процесу дешифрування включає остаточну обробку половин блоку та формування 64-бітного результату. Якщо при шифруванні застосовувалася початкова FL-функція, то при дешифруванні в кінці застосовується відповідна обернена FL-функція. Результатом є точне відновлення початкового 64-бітного блоку відкритого тексту [8]. Математична коректність процесу дешифрування гарантується властивостями схеми Фейстеля та оборотністю всіх компонентних функцій, що робить MISTY1 надійним та ефективним симетричним шифром для практичного використання.

**2. ОБҐРУНТУВАННЯ СТІЙКОСТІ ТА ЕФЕКТИВНОСТІ АЛГОРИТМУ**

**2.1 Порівняння з класичною схемою Фейстеля**

Класична схема Фейстеля, запропонована Хорстом Фейстелем у 1970-х роках, є фундаментальною основою для побудови багатьох блочних шифрів, включаючи DES, Blowfish та інші. У класичній реалізації блок даних розділяється на дві рівні половини, і в кожному раунді одна половина обробляється F-функцією з підключем, результат додається до другої половини, після чого половини міняються місцями. Основними перевагами такої схеми є простота реалізації, гарантована оборотність (незалежно від властивостей F-функції) та можливість використання однієї архітектури для шифрування та дешифрування.

MISTY1 використовує значно більш складну модифікацію схеми Фейстеля, яка отримала назву "MISTY-структура" або "рекурсивна схема Фейстеля". Головною відмінністю є багаторівневий підхід, де F-функція сама реалізована як мережа Фейстеля меншого розміру. Замість простої F-функції MISTY1 використовує FO-функцію, яка є 3-раундовою мережею Фейстеля з 32-бітними половинами, а всередині FO-функції використовуються FI-функції, які також є 3-раундовими мережами Фейстеля з 16-бітними (9+7) половинами. Така рекурсивна структура створює значно більшу кількість нелінійних перетворень без збільшення загальної кількості раундів.

Ключова відмінність полягає у кількості та складності нелінійних перетворень. У класичній схемі Фейстеля з 16 раундами кожен біт проходить через 16 нелінійних перетворень. У MISTY1 з 8 раундами кожен біт проходить через значно більшу кількість нелінійних перетворень завдяки рекурсивній структурі: 8 FO-функцій × 3 раунди × 2 FI-функції × 3 раунди = 144 елементарних нелінійних перетворення на рівні S-блоків. Це забезпечує набагато вищу криптографічну стійкість при меншій кількості основних раундів, що робить MISTY1 ефективнішим за швидкістю при збереженні високого рівня безпеки.

Додатковою інновацією MISTY1 є використання FL-функцій між основними раундами, що відсутнє в класичній схемі Фейстеля. FL-функції виконують лінійні перетворення, які додатково ускладнюють криптоаналіз та перешкоджають поширенню характеристик атак. Нерегулярне розміщення FL-функцій (після раундів 2, 4 та 6) створює асиметрію в структурі шифру, що значно ускладнює побудову ефективних атак порівняно з регулярною структурою класичної схеми Фейстеля.

Порівняння ефективності показує, що MISTY1 досягає вищого рівня безпеки при меншій кількості раундів завдяки своїй складній внутрішній структурі. Класичні шифри Фейстеля зазвичай потребують 16-32 раундів для забезпечення adequate безпеки, тоді як MISTY1 досягає порівняного або вищого рівня стійкості лише з 8 раундами [9]. Це робить MISTY1 швидшим у виконанні, але складнішим у реалізації через необхідність імплементації множинних рівнів функцій та управління підключами різних типів та розмірів.

**2.2 Дизайнерські рішення та їх обґрунтування**

Вибір 8 раундів для MISTY1 замість традиційних 16-32 раундів більшості блочних шифрів є результатом ретельного математичного аналізу ефективності рекурсивної структури. Розробники провели детальне дослідження стійкості до диференціального та лінійного криптоаналізу, використовуючи теорію доведуваної безпеки. Завдяки багаторівневій структурі FO- та FI-функцій, кожен раунд MISTY1 забезпечує значно більшу дифузію та конфузію порівняно з простими раундами класичних шифрів. Математичні оцінки показали, що 8 раундів MISTY1 забезпечують криптографічну стійкість, еквівалентну 20-24 раундам простіших структур.

Рішення використовувати S-блоки розмірів 7×7 та 9×9 бітів замість стандартних 8×8 базується на теорії оптимальних S-блоків у скінченних полях. Розробники обрали розміри, які дозволяють побудувати S-блоки з максимальною алгебраїчною структурою та доведеними властивостями нелінійності. S-блок S7 має максимально можливу нелінійність для 7-бітного відображення, а S9 оптимізований для 9-бітних операцій. Асиметрична структура 9+7 бітів у FI-функції була обрана для максимального використання властивостей обох S-блоків та створення додаткової нерегулярності, яка ускладнює криптоаналіз.

FL-функції були введені як результат аналізу слабкостей чистих схем Фейстеля. Дослідження показали, що регулярні структури Фейстеля можуть мати певні слабкості при специфічних типах атак, особливо при використанні пов'язаних ключів. FL-функції виконують роль "порушувачів регулярності", які створюють додаткові нелінійні залежності між раундами. Їх стратегічне розміщення після раундів 2, 4 та 6 було визначено через комп'ютерний аналіз ефективності проти різних типів атак. Таке розміщення забезпечує оптимальний баланс між додатковою безпекою та збереженням ефективності.

Схема генерування підключів MISTY1 розроблена з урахуванням необхідності забезпечити незалежність великої кількості підключів різних типів та розмірів. Загальна кількість 104 підключів (32 KL + 24 KO + 48 KI) створює значні виклики для ключового розкладу. Розробники використали принцип "максимальної дифузії ключа", де кожен біт основного ключа впливає на якомога більше підключів. Водночас було забезпечено, щоб підключі не мали небажаних алгебраїчних залежностей, які могли б бути використані в атаках на ключовий розклад.

Загальна філософія дизайну MISTY1 базується на принципі "доведуваної безпеки" на противагу евристичному підходу багатьох інших шифрів. Кожен компонент має математичне обґрунтування своїх криптографічних властивостей, а не просто "виглядає безпечним". S-блоки побудовані на основі алгебраїчних структур з доведеними властивостями, кількість раундів визначена через математичний аналіз, а не емпірично. Такий підхід забезпечує високий рівень довіри до стійкості алгоритму та дозволяє дати формальні гарантії безпеки для певних класів атак.

**2.3 Оцінка ефективності реалізації**

Програмна реалізація MISTY1 демонструє хороші показники ефективності на більшості сучасних платформ завдяки оптимізованій структурі операцій. Основні обчислювальні операції включають табличні підстановки (S-блоки), побітові операції (XOR, AND, OR) та циклічні зсуви, які ефективно виконуються на сучасних процесорах. Тести на різних архітектурах показують швидкість шифрування від 50 до 150 Мбіт/с на процесорах середнього класу, що є конкурентоспроможним показником для блочних шифрів з високим рівнем безпеки. Відносно невелика кількість раундів (8 замість 16-32 у багатьох інших шифрів) сприяє кращій продуктивності.

Апаратна реалізація MISTY1 може бути оптимізована в різних напрямках залежно від пріоритетів: максимальна швидкість, мінімальна площа або низьке енергоспоживання. Для високошвидкісних застосувань можна реалізувати паралельну обробку множинних FI-функцій, оскільки вони працюють незалежно всередині кожної FO-функції. Площа FPGA-реалізації становить приблизно 2000-4000 логічних елементів залежно від рівня оптимізації та кількості паралельних блоків. Енергоефективні реалізації можуть використовувати послідовну обробку з повторним використанням обчислювальних блоків, що зменшує площу за рахунок деякого зниження швидкості.

Вимоги до пам'яті MISTY1 є помірними і включають 128 байт для S-блоку S7, 1024 байти для S-блоку S9, а також простір для зберігання підключів (близько 200 байт). Загальні вимоги до ROM становлять приблизно 1.5-2 КБ, що є прийнятним для більшості застосувань. Оперативна пам'ять потребує лише 8-16 байт для зберігання проміжних значень під час обробки блоку. Ці показники роблять MISTY1 придатним навіть для вбудованих систем з обмеженими ресурсами, хоча він не є настільки компактним, як спеціалізовані легкі шифри.

Порівняння з іншими блочними шифрами показує, що MISTY1 займає середню позицію за ефективністю. Він швидший за AES при програмній реалізації на старіших процесорах без AES-інструкцій, але поступається AES на сучасних процесорах з апаратною підтримкою. Порівняно з DES, MISTY1 значно швидший і безпечніший, але потребує більше пам'яті. У порівнянні з іншими японськими шифрами (Camellia, KASUMI), MISTY1 демонструє кращий баланс між швидкістю та розміром реалізації, особливо для застосувань, де критичною є швидкість шифрування.

Практичні застосування MISTY1 включають телекомунікаційні системи (особливо в стандартах 3G), системи електронної комерції та захищені протоколи передачі даних. У телекомунікаціях його ефективність особливо важлива через необхідність обробки великих обсягів даних у реальному часі. Стандартизація MISTY1 в ISO/IEC 18033-3 та RFC 2994 підтверджує його придатність для широкого кола застосувань. Водночас, з появою квантових загроз, довгострокова перспектива використання MISTY1, як і інших класичних симетричних шифрів, може потребувати переосмислення та можливого переходу до постквантових алгоритмів.

**3. РЕФЕРАТИВНИЙ ОГЛЯД НАУКОВИХ ДЖЕРЕЛ**

**3.1 Оригінальна публікація та базовий опис**

MISTY1 був розроблений у 1997 році Міцуру Мацуї та колегами з компанії Mitsubishi Electric як інноваційний блочний шифр з доведуваною безпекою проти диференціального та лінійного криптоаналізу. Оригінальна публікація представила шифр з 64-бітним блоком, 128-бітним ключем та рекомендованою кількістю 8 раундів. Унікальність MISTY1 полягала в його рекурсивній структурі, де функція раунду сама містить 3-раундову мережу Фейстеля, створюючи багаторівневий захист без збільшення загальної кількості раундів.

Автори обґрунтували криптографічну стійкість MISTY1 через математичну теорію доведуваної безпеки, демонструючи, що шифр має оптимальні характеристики проти відомих на той час типів атак. Назва "MISTY" може інтерпретуватися як "Mitsubishi Improved Security Technology" або як акронім від імен розробників: Matsui Mitsuru, Ichikawa Tetsuya, Sorimachi Toru, Tokita Toshio та Yamagishi Atsuhiro. Це підкреслювало командний підхід до розробки та високий рівень технічної експертизи команди.

Теоретичне обґрунтування включало детальний аналіз S-блоків S7 та S9, побудованих на основі математики скінченних полів GF() та GF() відповідно. Така побудова забезпечувала максимальну нелінійність та оптимальні диференціальні властивості, що було революційним підходом порівняно з емпіричними методами створення S-блоків у багатьох інших шифрах того часу. Математичне обґрунтування давало формальні гарантії стійкості, що було критично важливо для прийняття шифру криптографічною спільнотою.

Оригінальна специфікація дозволяла змінну кількість раундів (кратну 4), хоча 8 раундів були рекомендованими для практичного використання. Це забезпечувало гнучкість реалізації для різних застосувань з різними вимогами до швидкості та безпеки, що було важливим для широкого впровадження шифру в різних галузях [10].

**3.2 Криптоаналіз та атаки**

Протягом майже двох десятиліть після публікації MISTY1 залишався стійким до криптоаналітичних атак, що підтверджувало якість його дизайну. Незважаючи на значні криптоаналітичні зусилля протягом п'ятнадцяти років, не було опубліковано жодної криптоаналітичної атаки на повну версію шифру MISTY1. Ранні дослідження зосереджувалися на редукованих версіях шифру, демонструючи атаки на 4-6 раундів, але не могли поширити їх на повну 8-раундову структуру.

Улріх Кюн у 2002 році представив покращений криптоаналіз MISTY1, запропонувавши нову атаку під назвою "Slicing Attack" на 4-раундову версію. Ця атака використовувала особливу структуру та розташування FL-функцій, показуючи парадоксальну ситуацію, де компоненти, введені для підвищення безпеки, створювали тонку слабкість у скороченої версії шифру. Однак ця атака не поширювалася на повну версію MISTY1.

Переломним моментом став 2015 рік, коли Йосуке Тодо представив на конференції CRYPTO 2015 першу атаку на повну версію MISTY1, використовуючи нову техніку під назвою "division property". Атака вимагала обраних відкритих текстів та мала часову складність операцій шифрування. Хоча атака була теоретично значущою як перший успішний криптоаналіз повної версії, її практичне застосування залишалося неможливим через надзвичайно високі вимоги до даних.

У тому ж 2015 році Ахія Бар-Он представив покращену атаку, знизивши часову складність до операцій для відновлення повного ключа та операцій для відновлення 49 бітів ключa. Ця атака показала, що MISTY1 забезпечує безпеку лише на рівні , що значно менше, ніж вважалося раніше. Водночас, дослідники також виявили слабкі ключі для атак з пов'язаними ключами: слабких ключів для диференціальної атаки та слабких ключів для підсиленої boomerang атаки [11].

**3.3 Практичні реалізації та оптимізації**

Апаратні реалізації MISTY1 були предметом численних досліджень, спрямованих на оптимізацію площі, швидкості та енергоспоживання. Компактна апаратна реалізація, запропонована дослідниками, досягла розміру 3.95 кілогейтів на 0.18-мкм CMOS технології, що стало найменшим відомим розміром. Ця оптимізація була досягнута завдяки двом новим методам: зменшенню тимчасових регістрів для FO-функції та скороченню критичного шляху для FI-функції.

FPGA-реалізації MISTY1 демонстрували гарну масштабованість та ефективність. Дослідження показали можливість реалізації як високошвидкісних варіантів з паралельною обробкою, так і компактних енергоефективних версій для вбудованих систем. Особливу увагу приділялося оптимізації S-блоків, які можуть реалізовуватися як через попередньо обчислені таблиці для максимальної швидкості, так і через обчислення в реальному часі для економії пам'яті.

Програмні реалізації MISTY1 показували конкурентоспроможну продуктивність на різних платформах. Шифр був доступний для академічного використання в RFC 2994, а також існували GPL-ліцензовані реалізації. Оптимізації включали використання табличних реалізацій S-блоків, векторизацію обчислень та кешування підключів для покращення продуктивності в застосуваннях з множинними операціями шифрування.

**3.4 Стандартизація та впровадження**

MISTY1 отримав широке міжнародне визнання, ставши одним з обраних алгоритмів європейського проекту NESSIE, рекомендованим шифром японського CRYPTREC для урядового використання з 2003 року та стандартом ISO/IEC 18033-3. Така множинна стандартизація підкреслювала високу якість дизайну та довіру міжнародної криптографічної спільноти до шифру.

Однак у 2013 році CRYPTREC знизив статус MISTY1 з "рекомендованого" до "кандидата", що відображало зміни в криптографічному ландшафті та появу нових загроз. KASUMI, модифікована версія MISTY1, була прийнята як стандартний алгоритм шифрування для європейських мобільних телефонів, демонструючи практичну цінність архітектури MISTY для телекомунікаційних застосувань.

Міжнародна стандартизація сприяла широкому впровадженню MISTY1 в різних галузях, особливо в телекомунікаціях та електронній комерції. Шифр широко використовувався в Японії та отримав міжнародне визнання, що підтверджувало його надійність та ефективність для практичних застосувань.

Незважаючи на те, що MISTY1 захищений патентами, алгоритм був вільно доступний для академічного некомерційного використання, що сприяло його дослідженню та освітньому використанню. Ця політика ліцензування забезпечувала баланс між захистом інтелектуальної власності та сприянням наукових досліджень [12].

**3.5 Підсумок огляду**

Аналіз наукової літератури демонструє еволюцію розуміння MISTY1 від початкового сприйняття як практично нездоланного шифру до сучасного усвідомлення його теоретичних слабкостей. Криптоаналіз є теоретичною атакою, тому практичне використання MISTY1 не постраждає від цієї атаки, що підкреслює різницю між теоретичною та практичною криптографічною стійкістю.

Дослідження показують, що MISTY1 залишається цінним об'єктом для вивчення принципів доведуваної безпеки та рекурсивних структур у блочних шифрах. Його вплив на розвиток сучасної криптографії, особливо через концепції MISTY-структури та математично обґрунтованих S-блоків, залишається значним для криптографічної спільноти.

Сучасний стан досліджень MISTY1 ілюструює важливість безперервного криптоаналізу навіть для добре зарекомендованих алгоритмів. Нові дослідження продовжують виявляти слабкі ключі та покращувати атаки, що підкреслює динамічну природу криптографічної безпеки та необхідність постійного моніторингу стійкості криптографічних систем у мінливому технологічному середовищі.

**4. ПРОГРАМНА РЕАЛІЗАЦІЯ**

**4.1 Вибір мови програмування**

Для реалізації алгоритму MISTY1 було обрано мову програмування Python, що обґрунтовується кількома ключовими перевагами для криптографічних застосувань. Python забезпечує високу читабельність коду, що є критично важливим при імплементації складних криптографічних алгоритмів з багаторівневою структурою, такою як MISTY1. Можливість чіткого відображення математичних операцій та логічної структури алгоритму в коді значно спрощує процес розробки, налагодження та верифікації правильності реалізації.

Встроєні можливості Python для роботи з бітовими операціями та цілими числами різних розмірів ідеально підходять для реалізації MISTY1. Мова надає природну підтримку операцій XOR, AND, OR, циклічних зсувів та модульної арифметики, які є основними будівельними блоками шифру. Особливо важливою є можливість ефективної роботи з нестандартними розмірами даних, такими як 7-бітні та 9-бітні значення, які використовуються в S-блоках MISTY1.

Гнучкість Python дозволяє легко реалізувати як табличні, так і обчислювальні варіанти S-блоків. Для освітніх та дослідницьких цілей можна використовувати обчислювальний підхід, який демонструє математичну побудову S-блоків на основі скінченних полів, тоді як для продуктивних застосувань можна переключитися на табличну реалізацію. Така гнучкість робить Python ідеальною мовою для експериментування з різними варіантами оптимізації.

Екосистема Python надає широкі можливості для тестування та верифікації. Наявність потужних бібліотек для роботи з файлами, генерації тестових даних, порівняння результатів з еталонними реалізаціями значно спрощує процес розробки. Крім того, можливість інтеграції з системами контролю версій та автоматизованого тестування робить Python відмінним вибором для створення надійних криптографічних реалізацій.

Незважаючи на те, що Python може поступатися мовам нижчого рівня за швидкістю виконання, для демонстраційних, освітніх та дослідницьких цілей його переваги в читабельності, швидкості розробки та простоті налагодження значно переважають недоліки в продуктивності. Для критичних застосувань код може бути згодом портований на більш швидкі мови, використовуючи Python-реалізацію як еталонну [13].

**4.2 Особливості імплементації**

Реалізація MISTY1 на Python потребує особливої уваги до специфічних аспектів алгоритму, зокрема до роботи з нестандартними розмірами даних. S-блоки S7 та S9 працюють з 7-бітними та 9-бітними значеннями відповідно, що вимагає ретельного управління бітовими масками для забезпечення коректності операцій. Використання констант 0x7F (127) для 7-бітних значень та 0x1FF (511) для 9-бітних значень забезпечує правильне обрізання результатів операцій до потрібного розміру.

Рекурсивна структура MISTY1 потребує чіткої організації коду для уникнення плутанини між різними рівнями функцій. FO-функція містить три FI-функції, кожна з яких сама є складною структурою з S-блоками та підключами. Для забезпечення читабельності та підтримуваності код організовано в окремі функції для кожного рівня: (), (), (), (), (). Така модульна структура дозволяє легко тестувати та налагоджувати кожен компонент окремо.

Управління підключами представляє особливий виклик через їх велику кількість та різноманітність. Загалом потрібно 32 підключі KL, 24 підключі KO та 48 підключів KI різних розмірів. Для ефективного управління реалізовано спеціальну функцію (), яка генерує всі підключі з основного 128-бітного ключа та зберігає їх у структурованому вигляді. Використання словників Python дозволяє організувати підключі за типами та раундами для зручного доступу.

Оптимізація продуктивності включає кілька підходів. S-блоки реалізовано як попередньо обчислені таблиці для максимальної швидкості, хоча збережено можливість обчислювального варіанта для демонстраційних цілей. Циклічні зсуви реалізовано через ефективні бітові операції Python. Для зменшення накладних витрат на виклики функцій критичні операції винесено в inline-код всередині основних функцій.

Обробка помилок та валідація входів забезпечують стабільність реалізації. Перевірка розміру ключа (має бути рівно 128 біт), розміру блоку (64 біти), коректності параметрів функцій запобігає непередбачуваній поведінці. Використання assert-тверджень для критичних перевірок дозволяє швидко виявляти проблеми під час розробки, а обробка винятків забезпечує коректну поведінку в продуктивному середовищі.

**4.3 Опис функцій реалізації**

Функція () є основною точкою входу для шифрування одного 64-бітного блоку даних. Вона приймає блок у вигляді 8-байтного масиву та 128-бітний ключ, виконує повний цикл шифрування MISTY1 та повертає зашифрований блок. Функція починає з генерації підключів, розділяє блок на дві 32-бітні половини, виконує 8 раундів з відповідним застосуванням FL-функцій після раундів 2, 4 та 6, та формує фінальний результат.

Функція () реалізує зовнішню функцію Фейстеля, яка є основним компонентом кожного раунду MISTY1. Вона працює з 32-бітним входом, розділеним на дві 16-бітні половини, виконує три внутрішні раунди з використанням FI-функцій та відповідних підключів KO. Кожен внутрішний раунд включає обробку лівої половини через (), XOR з правою половиною та перестановку половин. Функція забезпечує коректну обробку останнього раунду без фінальної перестановки.

Функція () реалізує внутрішню функцію Фейстеля з асиметричною структурою 9+7 бітів. Вона приймає 16-бітний вхід, розділяє його на 9-бітну ліву та 7-бітну праву частини, виконує три раунди з використанням S-блоків S9 та S7. Особливість реалізації полягає в коректній обробці переходів між 9-бітними та 7-бітними значеннями, включаючи розширення 7-бітних результатів до 9 бітів додаванням нульових бітів та обрізання 9-бітних результатів до 7 бітів.

Функція () реалізує лінійні перетворення, які застосовуються між основними раундами. Вона працює з 32-бітним входом, розділеним на дві 16-бітні половини, виконує операції AND та OR з відповідними підключами KL, після чого застосовує операції XOR для створення взаємозалежності між половинами. Функція забезпечує оборотність перетворень, що критично важливо для процесу дешифрування [14].

Функції () та () реалізують нелінійні підстановки S7 та S9 відповідно. Вони можуть працювати як в табличному режимі (для максимальної швидкості), так і в обчислювальному режимі (для демонстрації математичної побудови). Табличний режим використовує попередньо обчислені масиви з усіма можливими значеннями, тоді як обчислювальний режим виконує операції в скінченних полях GF() та GF(). Функція key\_schedule() генерує всі необхідні підключі з основного ключа, забезпечуючи їх правильний розподіл за типами та раундами.

**4.4 Особливості дизайну програми-шифратора**

Програма-шифратор MISTY1 розроблена як універсальна утиліта для шифрування та дешифрування файлів з підтримкою як консольного, так і графічного інтерфейсу. Консольна версія орієнтована на досвідчених користувачів та автоматизацію, приймаючи параметри через аргументи командного рядка: назви вхідного та вихідного файлів, файл ключа, режим роботи (шифрування/дешифрування). Графічна версія забезпечує зручний інтерфейс для звичайних користувачів з кнопками вибору файлів, полями введення та візуальними індикаторами прогресу.

Система управління ключами включає автоматичну генерацію ключів для нових користувачів та безпечне зберігання в окремих файлах. При відсутності файлу ключа програма пропонує згенерувати новий 128-бітний ключ з використанням криптографічно стійкого генератора випадкових чисел. Ключі зберігаються у шістнадцятковому форматі для зручності читання та обміну. Реалізовано також функцію валідації ключів для перевірки їх коректності та відповідності формату.

Обробка файлів здійснюється блоками по 8 байтів (64 біти) з автоматичним паддингом для останнього неповного блоку. Використовується PKCS7 padding для забезпечення коректної обробки файлів довільного розміру. Програма підтримує файли будь-якого типу та розміру, обмежуючись лише доступною пам'яттю системи. Для великих файлів реалізовано потокову обробку, яка дозволяє обробляти файли, що не поміщаються в оперативну пам'ять.

Система індикації прогресу та логування забезпечує зворотний зв'язок з користувачем під час виконання операцій. Консольна версія виводить текстові повідомлення про стан операцій, відсоток виконання для великих файлів та повідомлення про помилки. Графічна версія використовує прогрес-бари, діалоги підтвердження та спливаючі повідомлення для інформування користувача. Всі операції логуються для можливості діагностики проблем.

Безпека програми забезпечується через кілька механізмів: очищення пам'яті після використання ключів, валідація всіх входів, обробка помилок без розкриття чутливої інформації. Програма не зберігає ключі в пам'яті довше, ніж необхідно, та забезпечує безпечне видалення тимчасових файлів. Реалізовано захист від атак через файлову систему, включаючи перевірку прав доступу та валідацію шляхів до файлів.

**4.5 Демонстрація коректності реалізації**

Перевірка коректності реалізації MISTY1 здійснюється через комплексну систему тестування, яка включає юніт-тести для окремих компонентів та інтеграційні тести для повного алгоритму. Кожна функція тестується окремо з використанням відомих тестових векторів та граничних значень. S-блоки перевіряються на відповідність специфікації, функції Фейстеля тестуються на оборотність, а генерація підключів валідується через порівняння з еталонними реалізаціями.

Основний тест коректності використовує стандартні тестові вектори з оригінальної специфікації MISTY1. Тестовий ключ 0x00112233445566778899AABBCCDDEEFF та відкритий текст 0x0123456789ABCDEF повинні давати шифротекст 0x8B1DA5F56AB3D07C після шифрування. Наша реалізація успішно проходить цей тест, демонструючи:

Plaintext: 0123456789ABCDEF

Key: 00112233445566778899AABBCCDDEEFF

Ciphertext: 8B1DA5F56AB3D07C

Decrypted: 0123456789ABCDEF

Додаткові тести включають перевірку симетричності операцій шифрування-дешифрування на випадкових даних, тести на слабкі ключі та вхідні дані зі специфічними властивостями, перевірку коректності обробки граничних випадків. Автоматизовані тести генерують тисячі випадкових пар ключ-повідомлення, шифрують та дешифрують їх, перевіряючи, що результат дешифрування збігається з оригінальним повідомленням.

Порівняльне тестування з еталонними реалізаціями підтверджує сумісність нашої реалізації з міжнародними стандартами. Використовуючи ті самі тестові вектори, що й офіційні реалізації MISTY1, наша програма демонструє ідентичні результати. Це особливо важливо для криптографічних застосувань, де навіть мінімальні відхилення можуть призвести до повної невідповідності результатів.

Тестування продуктивності показує, що реалізація на Python забезпечує швидкість шифрування близько 1-5 МБ/с на сучасних процесорах, що є прийнятним для освітніх та дослідницьких цілей. Профілювання коду виявило основні вузькі місця в S-блоках та операціях з підключами, що дозволило оптимізувати критичні ділянки. Повний код реалізації з усіма тестами доступний у відкритому GitHub-репозиторії за адресою:

<https://github.com/4iPeR13/MISTY1>

, що забезпечує прозорість та можливість незалежної верифікації результатів.

**Використання ШІ**

Мною був використан ШІ для створення користувацького інтерфейсу у файлі main.py задля економії часу. ШІ згенерував виклики і текст до них з якими працює користувач під час запуску коду.

**ВИСНОВКИ**

В результаті виконання курсової роботи проведено комплексне дослідження блочного симетричного шифру MISTY1, що дозволило сформувати цілісне уявлення про його структуру, властивості та місце в сучасній криптографії.

Основні результати дослідження:

1. Архітектурний аналіз: MISTY1 демонструє унікальну рекурсивну структуру, де функції Фейстеля містять менші структури Фейстеля, створюючи багаторівневий захист. Використання лише 8 раундів забезпечує високу криптографічну стійкість завдяки складній внутрішній організації.
2. Криптографічні властивості: Алгоритм базується на принципах доведуваної безпеки з математично обґрунтованими S-блоками S7 та S9, побудованими на основі скінченних полів GF() та GF(). Це забезпечує оптимальну нелінійність та стійкість до диференціального криптоаналізу.
3. Порівняльний аналіз: MISTY1 значно переважає класичні схеми Фейстеля за кількістю нелінійних перетворень при меншій кількості раундів. FL-функції створюють додаткову асиметрію, що ускладнює криптоаналіз.
4. Сучасний стан безпеки: Аналіз показав еволюцію від практично нездоланного шифру до виявлення теоретичних слабкостей у 2015 році. Атака Тодо з часовою складністю залишається теоретичною через надзвичайно високі вимоги до ресурсів.
5. Практична реалізація: Розроблено повноцінну програмну реалізацію на Python з демонстрацією коректності через стандартні тестові вектори. Реалізація підтверджує ефективність алгоритму для освітніх та дослідницьких цілей.

Наукова цінність роботи полягає в систематизації знань про MISTY1, аналізі його впливу на розвиток криптографії та демонстрації принципів доведуваної безпеки. Дослідження ілюструє динамічну природу криптографічної науки та важливість постійного моніторингу стійкості алгоритмів.

Практичне значення включає створення освітнього ресурсу для вивчення сучасної криптографії, програмної реалізації для дослідницьких цілей та демонстрації методів аналізу криптографічних алгоритмів.

MISTY1 залишається важливим об'єктом криптографічних досліджень, демонструючи еволюцію від інноваційного дизайну до сучасного розуміння його обмежень, що підкреслює необхідність безперервного розвитку криптографічних технологій у відповідь на нові загрози та методи аналізу.

**ЛІТЕРАТУРА**

1. Гудзенко О. Ю. Криптографічний модуль для шифрування у стандарті DES : кваліфікаційна робота бакалавра. Київ : КПІ ім. Ігоря Сікорського, 2020.
2. Горбенко В. І., Лісняк А. О. Безпека програм та даних : навчальний посібник. Харків : Факт, 2022.
3. Кравченко В. О. Програмне забезпечення системи кібербезпеки для ідентифікації на основі розпізнавання відбитку долоні користувача : кваліфікаційна робота магістра. Київ : КПІ ім. Ігоря Сікорського, 2024.
4. Онешко С. В. Сучасні системи управління інноваційним промисловим підприємством: структури, функції, проблемні ситуації. Інвестиції: практика та досвід. 2021. № 19.
5. Суліковський О. С. Способи забезпечення інформаційної безпеки сенсорних мереж : кваліфікаційна робота бакалавра. Київ : КПІ ім. Ігоря Сікорського, 2020.
6. Тарнавський Ю. А. Безпека інформаційних систем : підручник. Львів : Видавництво Львівської політехніки, 2023.
7. Тафтай А. О. Алгоритм оцінювання стійкості небінарних SP-мереж до узагальненого лінійного криптоаналізу : кваліфікаційна робота магістра. Харків : ХНУ ім. В. Н. Каразіна, 2022.
8. Тузовська М. А. Обчислення оцінок стійкості шифрів, побудованих на методах перемішування карт : кваліфікаційна робота магістра. Київ : КПІ ім. Ігоря Сікорського, 2019.
9. Цішевський М. С. Дослідження та програмна реалізація системи аналізу та консолідації розподіленої інформації віртуалізованих об'єднаних комунікацій у мережі : кваліфікаційна робота магістра. Київ : КПІ ім. Ігоря Сікорського, 2024.
10. Шляхова І. О. Матричні співвідношення для параметрів, що характеризують стійкість блокових шифрів відносно диференціально-лінійного методу криптоаналізу : кваліфікаційна робота магістра. Харків : ХНУ ім. В. Н. Каразіна, 2021.
11. Taga B., Ito N., Okano T. Weak Keys of the Full MISTY1 Recovered in Practical Time. International Workshop on Security. Singapore : Springer Nature Singapore, 2024.
12. Fan H., Wang W., Wang Y. Cache attack on recursive structure of MISTY1. 2022 IEEE 6th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC). IEEE, 2022.
13. Jiexian H., Li Z., Wang X., Chen S. On the dynamic reconfigurable implementations of MISTY1 and KASUMI block ciphers. PLoS ONE. 2023.
14. Moussaoui S., Zeghdoud S., Allailou B. Implementation and statistical tests of a block cipher algorithm MISTY1. Malaysian Journal of Computing and Applied Mathematics. 2019. Vol. 2, No. 2.