DOI: https://doi.org/10.36910/6775-2524-0560-2025-61-27
УДК 004.056.55:004.383.4
Розломій Інна Олександрівна, к.т.н., доцент
https://orcid.org/0000-0001-5065-9004
Косенюк Григорій Володимирович, к.т.н., доцент
https://orcid.org/0000-0003-2103-3904
Науменко Сергій Васильович, аспірант
https://orcid.org/0000-0002-6337-1605
Черкаський національний університет імені Богдана Хмельницького, м. Черкаси, Україна

МЕХАНІЗМИ КРИПТОГРАФІЧНОГО КОНТРОЛЮ АВТЕНТИЧНОСТІ КОДУ У СЕНСОРНИХ ВУЗЛАХ З ОБМЕЖЕНИМИ ОБЧИСЛЮВАЛЬНИМИ РЕСУРСАМИ

Розломій І. О., Косенюк, Г. В., Науменко С. В. Механізми криптографічного контролю автентичності коду у сенсорних вузлах з обмеженими обчислювальними ресурсами. У статті представлено механізм криптографічного контролю автентичності програмного коду у сенсорних вузлах із обмеженими обчислювальними ресурсами, орієнтований на мікроконтролери класу STM32 та ESP32. Визначено обмеження існуючих підходів до перевірки автентичності коду, таких як значні витрати енергії, обсяг пам'яті та затримки при використанні традиційних алгоритмів SHA-2 і RSA. Запропоновано полегшену архітектуру Secure Boot, яка базується на використанні хеш-функції SPONGENT та цифрового підпису ECDSA з короткими ключами, що дозволяє забезпечити захист без перевищення апаратних обмежень сенсорних пристроїв. Проведено порівняльний аналіз двох конфігурацій механізму, результати якого свідчать про суттєве зниження часу перевірки, енергоспоживання та використання пам'яті у полегшеній версії при збереженні прийнятного рівня криптостійкості. Описано особливості інтеграції рішення у середовище STM32CubeIDE та ESP-IDF, реалізацію механізмів оновлення прошивки з перевіркою автентичності, а також збереження контрольних сум і ключів у захищених областях пам'яті. Запропоноване рішення дозволяє адаптувати перевірку автентичності до специфіки розподілених кіберфізичних систем та закладає основу для побудови енергоефективної та стійкої до атак інфраструктури безпечного програмного забезпечення у вбудованих пристроях.
Ключові слова: Secure Boot, автентичність коду, сенсорні вузли, полегшена криптографія, STM32.

Rozlomii I., Koseniuk, G., Naumenko S. Mechanisms for cryptographic code authentication control in sensor nodes with limited computing resources. The article presents a mechanism for cryptographic authentication control of program code in sensor nodes with limited computational resources, targeting microcontrollers such as STM32 and ESP32. The limitations of existing code authentication approaches are identified, including high energy consumption, memory footprint, and verification delays when using conventional algorithms like SHA-2 and RSA. A lightweight Secure Boot architecture is proposed, based on the SPONGENT hash function and ECDSA digital signature with shortened keys, which enables code protection without exceeding the hardware constraints of sensor devices. A comparative analysis of two configurations of the mechanism demonstrates significant reductions in verification time, energy consumption, and memory usage in the lightweight version while maintaining an acceptable level of cryptographic resistance. The integration features of the solution into STM32CubeIDE and ESP-IDF environments are described, along with the implementation of firmware update procedures with authenticity verification and secure storage of reference hashes and public keys in protected memory regions. The proposed mechanism allows for the adaptation of authenticity control to the specific needs of distributed cyber-physical systems and lays the foundation for building an energy-efficient and attack-resilient secure software infrastructure in embedded devices.
Key words: Secure Boot, code authenticity, sensor nodes, lightweight cryptography, STM32.

Постановка проблеми та її зв'язок із важливими науковими чи практичними завданнями. Забезпечення криптографічної автентичності програмного коду у вбудованих пристроях, зокрема сенсорних вузлах кіберфізичних систем, є критичним завданням в умовах зростаючої кількості атак на рівні програмного забезпечення [1]. Проникнення або модифікація коду в сенсорному вузлі може призвести до некоректної роботи системи загалом, зокрема в галузях із підвищеними вимогами до надійності – таких як охорона здоров'я, енергетика, військова техніка та промислова автоматизація [2].

У вбудованих пристроях переважають мікроконтролери з обмеженим обсягом пам'яті, низькою тактовою частотою та обмеженим енергобюджетом, що унеможливлює застосування класичних криптографічних схем перевірки автентичності коду [3]. Сучасні алгоритми хешування та цифрового підпису мають значні обчислювальні витрати і не придатні для інтеграції у сенсорні модулі без адаптації або спрощення.

Проблема полягає у відсутності комплексного технічного рішення для побудови механізмів перевірки автентичності коду, що можуть працювати в умовах обмежених ресурсів, забезпечуючи при цьому достатній рівень стійкості до атак [4]. Зокрема, актуальною є розробка механізмів

захищеного завантаження (Secure Boot), які поєднують спрощені хеш-функції, ефективні схеми цифрового підпису та протидію поширеним атакам на bootloader і процес оновлення прошивки.

З огляду на зазначене, виникає потреба у створенні спеціалізованого механізму криптографічного контролю автентичності коду, придатного до реалізації у сенсорних пристроях з обмеженими обчислювальними ресурсами. Такий механізм має бути орієнтований на практичну реалізацію у мікроконтролерному середовищі, забезпечувати перевірку автентичності та цілісності програмного коду перед його виконанням або оновленням, а також враховувати обмеження щодо енергоспоживання, продуктивності й пам'яті.

Аналіз останніх досліджень та публікацій. Проблематика перевірки автентичності та цілісності коду у вбудованих пристроях розглядається у низці сучасних досліджень, що зосереджуються на розробці механізмів захищеного завантаження (Secure Boot), схем хешування та цифрового підпису, придатних до виконання в умовах обмежених обчислювальних ресурсів. Зокрема, у [5, 6] запропоновано реалізації Secure Boot для мікроконтролерів STM32 з використанням алгоритмів SHA-2 та RSA/ECC, однак зазначено, що їх інтеграція у сенсорні пристрої супроводжується значними затримками під час верифікації та перевищенням обсягу доступної пам'яті.

Окрему увагу у публікаціях [7, 8] приділено використанню полегшених хеш-функцій, таких як SPONGENT, PHOTON, QUARK та інших, які демонструють кращі характеристики енергоспоживання порівняно з класичними алгоритмами, проте потребують ретельної оцінки на предмет криптостійкості в умовах реального впровадження. У роботах [9, 10] також аналізується застосування компактних цифрових підписів, зокрема варіантів на основі алгоритму Ed25519 або варіацій ECDSA з укороченим ключем, але звертається увага на необхідність компромісу між рівнем безпеки та розміром коду.

В публікаціях [11, 12] розглядають концепцію реалізації багаторівневої перевірки автентичності з урахуванням моделей погроз, які передбачають атаки типу code injection, rollback attack, або модифікацію прошивки при оновленні. При цьому не завжди враховується динамічний контекст використання сенсорних вузлів у розподілених кіберфізичних системах, а отже й обмеження щодо часу відповіді, доступної енергії та відсутності апаратного криптомодуля.

Загалом, незважаючи на наявність значного обсягу робіт, відсутня уніфікована технічна модель, що дозволяє реалізувати криптографічний контроль автентичності коду у сенсорних вузлах з обмеженими ресурсами з урахуванням специфіки мікроконтролерного середовища, типових загроз та вимог до швидкодії й енергоефективності. Це зумовлює необхідність подальших досліджень та розробки практичних рішень у цій галузі.

Мета. Метою статті є розробка механізму криптографічного контролю автентичності програмного коду у сенсорних вузлах з обмеженими обчислювальними ресурсами, що забезпечує перевірку цілісності та автентичності програмного коду на етапі завантаження або оновлення, з урахуванням обмежень щодо пам'яті, енергоспоживання та обчислювальної потужності мікроконтролера.

Виклад основного матеріалу й обґрунтування отриманих результатів дослідження. Одним із ключових етапів забезпечення автентичності програмного коду у сенсорних вузлах є реалізація механізму Secure Boot – процесу безпечного завантаження, який гарантує, що лише перевірений і не модифікований код буде виконано на пристрої.

У традиційних обчислювальних системах Secure Boot реалізується за допомогою комбінації апаратного довіреного якоря (Root of Trust), цифрового підпису прошивки та верифікації з використанням асиметричної криптографії. У контексті пристроїв з обмеженими ресурсами така реалізація потребує адаптації, зокрема використання полегшених алгоритмів та спрощеної ієрархії перевірок.

У межах цього дослідження запропоновано архітектуру Secure Boot, орієнтовану на мікроконтролери класу STM32/ESP32, з урахуванням обмежень щодо обсягу пам'яті, тактової частоти та енергоспоживання. Архітектура охоплює такі етапи:

1) Етап 0 – завантаження з внутрішньої пам'яті Bootloader'а, записаного виробником або розробником, що є недоступним для модифікації після виробництва;

2) Етап 1 – перевірка цифрового підпису або хешу головного завантажувального коду (Main Firmware) з використанням вбудованого відкритого ключа або попередньо обчисленого еталонного хешу;

3) Етап 2 – у разі успішної перевірки – передача керування основному коду; у разі невідповідності – зупинка завантаження або запуск аварійного режиму;

4) Етап 3 – опціональна перевірка оновлень прошивки перед перезаписом у пам'ять (як частина механізму захисту від rollback-атак).

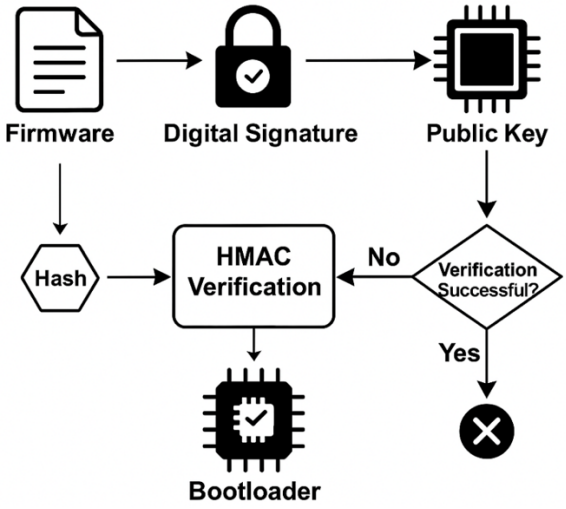

Рис. 1. Структура реалізації Secure Boot у сенсорному пристрої

Зображена на рисунку 1 архітектура реалізації Secure Boot враховує особливості мікроконтролерів, які використовуються в сенсорних пристроях. Її перевагою є можливість інтеграції в програмне середовище без необхідності залучення додаткового апаратного криптографічного модуля, що робить механізм придатним для масового впровадження у КФС.

На практиці важливою частиною реалізації є інтеграція механізмів хешування і перевірки підпису безпосередньо у середовище розробки мікроконтролера, наприклад, у проєкт на базі STM32CubeIDE або ESP-IDF, з використанням бібліотек криптографічного рівня, оптимізованих під обмежені ресурси. Залежно від обраної архітектури, механізм хешування (SPONGENT, PHOTON тощо) може реалізовуватись у вигляді окремого модуля, який обробляє блоки коду під час перевірки або оновлення прошивки.

Крім того, враховано підтримку поетапного оновлення прошивки (staged update), яке передбачає попередню перевірку автентичності нового образу та тимчасове збереження у зовнішній або окремій області пам'яті. Лише після успішного завершення перевірки основна область пам'яті перезаписується, що дозволяє знизити ризик виходу пристрою з ладу внаслідок недопустимого втручання у процес оновлення.

Окрему увагу приділено вибору формату збереження контрольних сум (еталонних хешів) та публічних ключів. У дослідженні розглянуто два варіанти: (1) жорстко зашиті значення у Bootloader, що унеможливлює оновлення ключа, і (2) збереження у захищеній області Flash-пам'яті з однократним записом (One-Time Programmable), що дозволяє гнучкіше керувати безпековою політикою.

Такий підхід дозволяє врахувати специфіку ресурсообмежених сенсорних пристроїв, уникнути надмірних навантажень на процесор під час завантаження, забезпечити контроль автентичності коду на критичних етапах життєвого циклу прошивки та зменшити вікно вразливості у випадку фізичного чи мережевого втручання.

Для перевірки придатності запропонованого механізму до практичного впровадження у сенсорних вузлах з обмеженими ресурсами було проведено тестування на експериментальних стендах із використанням мікроконтролерів STM32F103C8T6 та ESP32-S3. Основну увагу приділено вимірюванню тривалості перевірки автентичності коду, енергоспоживання під час цієї процедури, а також обсягу зайнятої пам'яті.

Для експерименту реалізовано дві конфігурації Secure Boot:

1) базова конфігурація – SHA-256 + RSA-2048 (референсний варіант);
2) полегшена конфігурація – SPONGENT-128 + ECDSA-160 (пропонований варіант).

Результати подано в таблиці 1.

Табл. 1. Порівняльні характеристики базової та полегшеної конфігурацій механізму криптографічного контролю автентичності коду

Параметр	SHA-256 + RSA-2048	SPONGENT-128 + ECDSA-160
Час перевірки підпису (мс)	219	81
Енергоспоживання під час перевірки (мДж)	4,3	1,6
Обсяг Flash-пам'яті (КБ)	32,7	17,4
Обсяг RAM (КБ)	6,1	2,5
Рівень криптостійкості	Високий	Середній+

Рівень стійкості оцінено відповідно до поточних рекомендацій ENISA та NIST для пристроїв класу IoT у некритичних застосуваннях.

Як показують результати, полегшена конфігурація демонструє істотне зменшення часу перевірки та енергоспоживання при прийнятному рівні криптостійкості, що робить її придатною для використання у сенсорних вузлах із батарейним живленням та обмеженою обчислювальною потужністю. Зменшення обсягів пам'яті також відкриває можливості для інтеграції в пристрої з мінімальними апаратними ресурсами.

З метою аналітичного обґрунтування ефективності використання полегшених криптографічних алгоритмів у рамках запропонованого механізму Secure Boot, проведено формалізацію обчислювального навантаження та енергоспоживання на кожному етапі верифікації.

Загальне криптографічне навантаження на мікроконтролер L_{total} можна подати у вигляді (1).

$$L_{total} = L_{hash} + L_{sig.ver} \quad (1)$$

де L_{hash} – кількість операцій, необхідних для обчислення хешу коду, $L_{sig.ver}$ – кількість операцій для перевірки цифрового підпису.

Кількість елементарних операцій під час хешування залежить від довжини вхідного коду N у байтах та параметрів блоку алгоритму хешування (2).

$$L_{hash} = \left[\frac{N}{B_h}\right] \cdot C_h \quad (2)$$

де B_h – розмір блоку хеш-функції (наприклад, 8 байт для SPONGENT-128), C_h – кількість операцій (ітерацій, побітових перетворень тощо) на один блок.

Для SPONGENT-128 із 80 раундами та невеликим блоком $B_h = 8$, значення C_h оцінюється як константа близько 10^3 інструкцій на блок, тоді як для SHA-256 з $B_h = 64$ і $C_h \approx 10^4$ – у 10 разів більше при тій самій довжині вхідного коду.

Оцінку навантаження на етапі перевірки підпису наведемо як (3).

$$L_{sig.ver} = C_s \cdot log_2(k) \quad (3)$$

де C_s – кількість операцій у криптографічному ядрі залежно від типу алгоритму (наприклад, для ECDSA це переважно операції множення в полі та обчислення зворотного елемента), k – розмір відкритого ключа в бітах.

Для ECDSA-160 $log_2(k) = 160$, тоді як для RSA-2048 – $log_2(k) = 2048$, що пояснює експериментальне зменшення часу верифікації більш ніж у 2,5 рази.

Енергоспоживання E можна аналітично оцінити як (4).

$$E = P_{avg} \cdot \left(\frac{L_{total}}{f}\right) \quad (4)$$

де P_{avg} – середнє споживання потужності під час криптографічних обчислень, f – тактова частота мікроконтролера.

Оскільки Ltotal для SPONGENT + ECDSA-160 є щонайменше в 2 рази меншим за класичну зв'язку SHA-256 + RSA-2048, при рівній тактовій частоті та однаковому P_{avg}, це напряму зменшує E без необхідності знижувати продуктивність системи.

Математична модель підтверджує, що використання полегшених криптографічних алгоритмів є обґрунтованим компромісом між продуктивністю, енергоспоживанням та прийнятною стійкістю до атак, що відповідає поставленій меті дослідження – створення ефективного механізму контролю автентичності у сенсорних пристроях із ресурсними обмеженнями.

Реалізація запропонованого механізму у вигляді бібліотеки мовою C дозволяє легко інтегрувати його у середовища STM32CubeIDE та ESP-IDF без залучення зовнішніх апаратних криптомодулів. Проведене тестування також показало відсутність суттєвого впливу механізму на час загального старту системи, що зберігає її реальну продуктивність.

Розроблений механізм може бути застосований у галузях, де сенсорні пристрої виконують критичні функції та не допускають втручання в програмне забезпечення, зокрема:

1) у медичних пристроях (монітори життєвих показників, інфузійні насоси), де критично важливо гарантувати достовірність коду;

2) в системах промислової автоматизації, що використовують сенсори з автономним живленням;

3) у військових та енергетичних кіберфізичних, де втручання в мікропрограму може призвести до збоїв безпекового рівня.

Низький рівень енергоспоживання та компактність запропонованого рішення дозволяють масштабувати його для застосування у великих розподілених мережах IoT без значних витрат на обчислювальні ресурси.

Висновки та перспективи подальшого дослідження. У статті представлено механізм криптографічного контролю автентичності коду, спеціально адаптований до умов сенсорних вузлів з обмеженими обчислювальними ресурсами. Запропонована архітектура Secure Boot враховує обмеження щодо енергоспоживання, пам'яті та обчислювальної потужності, що дозволяє інтегрувати перевірку цілісності та автентичності програмного коду без залучення апаратних криптомодулів.

Порівняльна оцінка показала, що використання полегшених криптографічних алгоритмів (SPONGENT-128 та ECDSA-160) дозволяє суттєво знизити час перевірки та енергоспоживання у порівнянні з класичними схемами (SHA-256 та RSA-2048), зберігаючи прийнятний рівень криптостійкості для більшості некритичних IoT-застосувань. Крім того, продемонстровано практичну реалізацію механізму у мікроконтролерних середовищах STM32 та ESP32, що підтверджує його придатність до використання у реальних проєктах.

Подальший розвиток дослідження доцільно зосередити на створенні комплексних моделей, що враховують енергетичний профіль мікроконтролерів різних архітектур, а також розширеній оцінці стійкості полегшених алгоритмів у контексті багаторівневих загроз. Перспективним напрямом є інтеграція моделей обчислювальної вартості та ризик-орієнтованих методів, здатних враховувати динамічні умови функціонування вбудованих пристроїв у реальному часі. Такий підхід відкриває можливості для створення нових ресурсоефективних рішень, придатних для широкого спектра сучасних кіберфізичних систем.

Список бібліографічного опису

1. Garg T., Khullar S., Kaur G. Security Issues and Challenges for Cyber-Physical Systems. Cyber-Physical Systems. Chapman and Hall/CRC, 2022. P. 161–176.
2. Laktionov I., Diachenko G., Moroz D., Getman I. A Comprehensive Review of Cybersecurity Threats to Wireless Infocommunications in the Quantum-Age Cryptography. *IoT*. 2025. Vol. 6, No. 4. P. 61.
1. Homaei M., Mogollón-Gutiérrez Ó., Sancho J. C., Ávila M., Caro A. A review of digital twins and their application in cybersecurity based on artificial intelligence. *Artificial Intelligence Review*. 2024. Vol. 57, No. 8. P. 201.
2. Розломій І. О., Фауре Е. В., Науменко С. В. Методи аутентифікації у вбудованих системах з обмеженими обчислювальними ресурсами. *Вимірювальна та обчислювальна техніка в технологічних процесах*. 2025. № 1. С. 29–35. DOI: https://doi.org/10.31891/2219-9365-2025-81-4
3. Bhasin S., De Santis F. (Eds.). *Constructive Side-Channel Analysis and Secure Design: 12th International Workshop, COSADE 2021, Lugano, Switzerland, October 25–27, 2021, Proceedings*. Vol. 12910. Springer Nature, 2021.
4. Phong N. D., Tuyen U. Q., Osinski P. Risk Warning Systems for Underground Mining Using IoT Solutions: A Case Study. *GEOMATE Journal*. 2024. Vol. 27, No. 119. P. 100–111.
5. Chakraborty R., Mondal U. K., Debnath A., Ghosh U., Roy B. B. Lightweight micro-architecture for IoT & FPGA security. *International Journal of Information Technology*. 2023. Vol. 15, No. 7. P. 3899–3905.
6. Luzhetskyi V. Hardware implementation of the HDG hash function. *Вісник Черкаського державного технологічного університету*. 2025. Т. 30, № 2. С. 10.
7. Shim K. A survey on post-quantum public-key signature schemes for secure vehicular communications. *IEEE Transactions on Intelligent Transportation Systems*. 2021. Vol. 23, No. 9. P. 14025–14042.
8. Nouma S. E., Yavuz A. A. Trustworthy and efficient digital twins in post-quantum era with hybrid hardware-assisted signatures. *ACM Transactions on Multimedia Computing, Communications and Applications*. 2024. Vol. 20, No. 6. P. 1–30.
9. Nittala E. P. Mitigating Cyber-Physical Attacks in ERP-Controlled Infrastructures through AI-Based Intrusion Response Systems. *International Journal of AI, BigData, Computational and Management Studies*. 2025. Vol. 6, No. 1. P. 151–160.
10. Li Z., Mashima D., Ong W. S., Esiner E., Kalbarczyk Z., Chang E. C. On Practicality of Using ARM TrustZone Trusted Execution Environment for Securing Programmable Logic Controllers. *Proceedings of the 19th ACM Asia Conference on Computer and Communications Security*, 2024. P. 947–961.

References

1. Garg T., Khullar S., Kaur G. Security Issues and Challenges for Cyber-Physical Systems. *Cyber-Physical Systems*. Chapman and Hall/CRC, 2022. P. 161–176.
2. Laktionov I., Diachenko G., Moroz D., Getman I. A Comprehensive Review of Cybersecurity Threats to Wireless Infocommunications in the Quantum-Age Cryptography. *IoT*. 2025. Vol. 6, No. 4. P. 61.
3. Homaei M., Mogollón-Gutiérrez Ó., Sancho J. C., Ávila M., Caro A. A review of digital twins and their application in cybersecurity based on artificial intelligence. *Artificial Intelligence Review*. 2024. Vol. 57, No. 8. P. 201.
4. Rozlomii I. O., Faure E. V., Naumenko S. V. Authentication methods in embedded systems with limited computational resources. *Measuring and Computing Equipment in Technological Processes*. 2025. No. 1. P. 29–35. DOI: https://doi.org/10.31891/2219-9365-2025-81-4
5. Bhasin S., De Santis F. (Eds.). *Constructive Side-Channel Analysis and Secure Design: 12th International Workshop, COSADE 2021, Lugano, Switzerland, October 25–27, 2021, Proceedings*. Vol. 12910. Springer Nature, 2021.
6. Phong N. D., Tuyen U. Q., Osinski P. Risk Warning Systems for Underground Mining Using IoT Solutions: A Case Study. *GEOMATE Journal*. 2024. Vol. 27, No. 119. P. 100–111.
7. Chakraborty R., Mondal U. K., Debnath A., Ghosh U., Roy B. B. Lightweight micro-architecture for IoT & FPGA security. *International Journal of Information Technology*. 2023. Vol. 15, No. 7. P. 3899–3905.
8. Luzhetskyi V. Hardware implementation of the HDG hash function. *Bulletin of Cherkasy State Technological University*. 2025. Vol. 30, No. 2. P. 10.
9. Shim K. A survey on post-quantum public-key signature schemes for secure vehicular communications. *IEEE Transactions on Intelligent Transportation Systems*. 2021. Vol. 23, No. 9. P. 14025–14042.
10. Nouma S. E., Yavuz A. A. Trustworthy and efficient digital twins in post-quantum era with hybrid hardware-assisted signatures. *ACM Transactions on Multimedia Computing, Communications and Applications*. 2024. Vol. 20, No. 6. P. 1–30.
11. Nittala E. P. Mitigating Cyber-Physical Attacks in ERP-Controlled Infrastructures through AI-Based Intrusion Response Systems. *International Journal of AI, BigData, Computational and Management Studies*. 2025. Vol. 6, No. 1. P. 151–160.
12. Li Z., Mashima D., Ong W. S., Esiner E., Kalbarczyk Z., Chang E. C. On Practicality of Using ARM TrustZone Trusted Execution Environment for Securing Programmable Logic Controllers. *Proceedings of the 19th ACM Asia Conference on Computer and Communications Security*, 2024. P. 947–961.