DOI: https://doi.org/10.26642/ten-2024-1(93)-147-152
УДК 004.056

Д.В. Дацюк, асистент
Ю.М. Россінський, к.т.н., доц.
В.В. Воротніков, д.т.н., проф.
Державний університет «Житомирська політехніка»

Безпека комунікації мікросервісів: використання квантово-стійких алгоритмів для підпису JWT-токену

У сучасному світі квантові обчислення стають реальністю, що висуває підвищені вимоги до безпеки комунікації між сервісами. У цьому контексті надзвичайно важливим є використання квантово-стійких алгоритмів для підписування JWT-токенів у мікросервісній архітектурі. Особливий інтерес становить порівняння ефективності квантово-стійких алгоритмів на основі решіток із традиційним RSA, щоб визначити оптимальні методи підписування токенів із точки зору продуктивності, безпеки та стійкості до квантових загроз. Під час дослідження було проведено серію запитів до сервісу та побудовано діаграми для порівняння затримок під час підписування та перевірки підписів. Зокрема, вимірювалися затримки для кожного алгоритму, щоб визначити середній час виконання операцій і зрозуміти вплив їхньої продуктивності на роботу мікросервісної архітектури. Результати дослідження показують, що квантово-стійкий алгоритм Dilithium демонструє значно вищу продуктивність у підписуванні та верифікації JWT-токенів порівняно з RSA. Цей алгоритм побудований на криптографії на ґратках, що забезпечує ефективну генерацію ключів і підписів навіть при високих рівнях безпеки. Однак збільшення розміру ключів і підписів призводить до підвищеного використання пропускної здатності мережі, що слід враховувати під час впровадження. Алгоритм Dilithium виявляється перспективним варіантом для забезпечення високої продуктивності та безпеки в мікросервісних системах. Його здатність швидко генерувати та перевіряти підписи сприяє ефективній та надійній комунікації між мікросервісами, зберігаючи при цьому стійкість до майбутніх квантових загроз. Використання квантово-стійких алгоритмів стає все більш актуальним у світлі зростаючих ризиків, пов'язаних із розвитком квантових обчислень.

Ключові слова: *квантово-стійка криптографія; JWT (JSON Web Token); RSA; Dilithium; криптографія на ґратках; мікросервісна архітектура; квантові загрози*.

Актуальність теми. В сучасній розробці JSON Web token (JWT) доволі часто використовується для забезпечення захищеної комунікації між сервісами в мікросервісній архітектурі. JWT ефективно слугує механізмом для автентифікації та авторизації міжсервісних запитів, дозволяючи сервісам легко ідентифікувати та перевіряти дозволи один одного без потреби в зверненні до центральної бази даних при кожному запиті. Однак існуючі методи підпису JWT-токену, які базуються на традиційних криптографічних алгоритмах, наприклад, RSA або ECC, виявляються вразливими перед потенційними атаками квантових обчислювальних систем. Основна загроза від квантово-обчислювальних систем полягає в їх потенціалі ефективно вирішувати проблеми, які залишаються недоступними для класичних комп'ютерів. Це стосується таких криптографічних алгоритмів, як факторизація великих простих чисел та розв'язання задачі дискретного логарифмування, які є основою багатьох криптографічних протоколів, враховуючи RSA, ECC та інші. Актуальність використання квантово-стійких алгоритмів для підпису JWT- токенів значно зростає у зв'язку з розвитком квантових обчислень.

Аналіз останніх досліджень та публікацій, на які спираються автори. Для відповідного аналізу розглянемо останні дослідження та публікації в цьому напрямі. Національний інститут стандартів та технологій (NIST) [1] відіграє важливу роль у процесі переходу до квантово-стійких криптографічних методів. Інститут запустив довгостроковий проєкт, який складається з визначення, оцінки та стандартизації квантово-стійких криптографічних алгоритмів, призначених для захисту цифрових комунікацій від потенційних загроз з боку квантових комп'ютерів.

Джон Пройс Маттссон, Бен Смітс та Ерік Тормаркер [2] детально розглядають квантово-стійку криптографію, ризики, пов'язані з криптографічно значущими квантовими комп'ютерами, та роль NIST у стандартизації квантово-стійкої публічної криптографії. Це дослідження акцентує увагу на потенціалі квантових ключових розподілів і квантових генераторів випадкових чисел як доповненні до традиційної криптографії.

У своєму дослідженні Лео Дюкас, Ейке Кілц, Танкред Лепоінт, Вадим Любашевський, Пітер Швабе, Грегор Зайлер та Дам'єн Стеле [3] аналізують схему цифрового підпису Dilithium, яка є частиною криптографічного набору CRYSTALS (Cryptographic Suite for Algebraic Lattices) та була запропонована на звернення NIST по стандартах постквантової криптографії. Вони зазначають, що особливістю схеми

Dilithium є її стійкість до квантових атак та висока ефективність на стандартних комп'ютерних архітектурах без спеціалізованого обладнання. Автори наголошують на значному зменшенні розміру публічного ключа в 2,5 раза порівняно з іншими «ґратковими» схемами, при цьому зберігаючи порівняльний розмір підпису. Їхній аналіз безпеки в умовах квантово випадкового оракула показує, що схема підпису Dilithium є не тільки швидкою та компактною, але й безпечною проти очікуваних квантово-комп'ютерних загроз.

У науковій праці Мохаммеда Фаріка та Шавката Алі [4] підкреслюється вразливість сучасних криптографічних методів у світлі розвитку квантових технологій. Основна увага зосереджена на тому, що алгоритми, які залежать від факторизації великих чисел або дискретних логарифмів, можуть бути швидко зламані за допомогою квартових обчислювальних систем. Автори закликають до заміни застарілих алгоритмів, таких як RSA та ECC, на нові, що засновані на криптографії на ґратках.

Наукові праці та публікації, які присвячені квантово-стійким алгоритмам, підкреслюють значний прогрес та нагальну потребу в адаптації до безпечних криптографічних методів у світлі розвитку квантових обчислень.

Метою статті є дослідження використання квантово-стійких алгоритмів для підпису JWT-токену, який буде використовуватися для комунікації між мікросервісами. Особлива увага зосереджується на порівнянні ефективності квантово-стійких алгоритмів із традиційним RSA, щоб визначити оптимальні методи підписування токенів у контексті безпеки, швидкості обробки та стійкості до майбутніх квантових загроз.

Викладення основного матеріалу. JSON Web Token – це відкритий стандарт (RFC 7519), який визначає компактний і самостійний спосіб безпечної передачі інформації між сторонами як JSON-об'єкт. Ця інформація може бути перевірена і довірена, тому що вона підписана цифровим підписом. JWT можуть бути підписані за допомогою секретного ключа або використанням пари ключів [5]. Згенерований JWT- токен – це рядок, який складається з трьох частин – заголовка, корисного навантаження та підпису. Заголовок та корисне навантаження містять у собі дані в JSON-форматі та закодовані в base64. Підпис містить хеш заголовка та корисного навантаження, які закодовані за допомогою алгоритму, який вказаний в заголовку. Список криптографічних алгоритмів, які можна використовувати для підпису JWT-токену, можна переглянути в таблиці 1 [6].

Таблиця 1

Значення параметрів заголовка для JWT (алгоритм)

«alg» параметр	**Цифровий підпис або Message Authentication Code (MAC) алгоритм**	**Вимоги до реалізації**
HS256	HMAC using SHA-256	Обов'язковий
HS384	HMAC using SHA-384	Необов'язковий
HS512	HMAC using SHA-512	Необов'язковий
RS256	RSASSA-PKCS1-v1_5 using SHA-384	Рекомендований
RS384	RSASSA-PKCS1-v1 5 using SHA-384	Необов'язковий
RS512	RSASSA-PKCS1-v1 5 using SHA-512	Необов'язковий
ES256	ECDSA using P-256 curve and SHA-256	Рекомендований
ES384	ECDSA using P-384 curve and SHA-384	Необов'язковий
ES512	ECDSA using P-521 curve and SHA-512	Необов'язковий
PS256	RSASSA-PSS using SHA-256 and MGF1 with SHA-256	Необов'язковий
PS384	RSASSA-PSS using SHA-384 and MGF1 with SHA-384	Необов'язковий
PS512	RSASSA-PSS using SHA-512 and MGF1 with SHA-512	Необов'язковий
none	No digital signature or MAC performed	Необов'язковий

Як можна побачити в таблиці 1, JWT наразі не підтримує квантово-стійких алгоритмів для підпису. Тому в межах дослідження було розроблено спеціалізовану систему підпису та верифікації для JWT, що відповідає вимогам безпеки в умовах потенційних квантових загроз. Як алгоритм підпису, що відповідає вимогам квантової стійкості, було вибрано алгоритм Dilithium. Dilithium відомий своєю високою стійкістю до різноманітних атак, враховуючи ті, що можуть бути ефективно здійснені квантовими обчислювальними системами.

Розроблена система є набором взаємодіючих компонентів, які впроваджені у вигляді API (інтерфейсу програмування додатків) [7]. Ці компоненти взаємодіють у межах мікросервісної архітектури [10], яка передбачає розподіл системи на невеликі, автономні сервіси, що виконують конкретні функції. Реалізація цієї системи базується на використанні Node.js та Nest.js [8] – потужних інструментів для створення

серверних додатків на мові JavaScript / TypeScript. Nest.js є відомим фреймворком, який сприяє розробці мікросервісних архітектур та надає широкі можливості для створення функціональних додатків. Він містить зручні засоби для обробки HTTP-запитів, керування залежностями та реалізації заходів безпеки. У системі реалізовано кастомний підпис JWT-токенів за допомогою алгоритму Dilithium, для цього було розроблено сервіс, який інтегрується з іншими мікросервісами. Під час запуску системи створюються приватний та публічні ключі, які потім використовуються для підпису JWT-токену. Генерація ключів відбувається за допомогою двох алгоритмів – Dilithium та RSA. Також автоматизовано генерується JWT-токен, який потім використовується під час комунікації між мікросервісами. Послідовність роботи системи показана на рисунку 1.

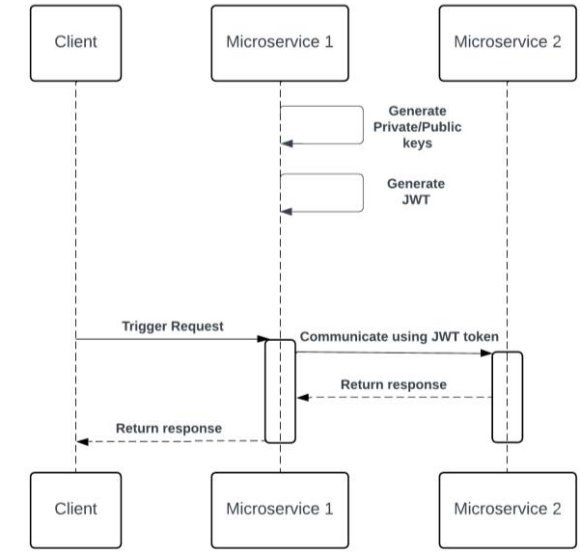

Рис. 1. Схема роботи системи

У межах цього дослідження було проведено порівняльний аналіз швидкодії алгоритмів RSA та Dilithium у контексті підпису JWT-токену. Для кожного алгоритму оцінювалися такі аспекти продуктивності:

- генерація ключів: визначалися показники швидкості та ефективності генерації ключів для кожного з алгоритмів. Це є важливим фактором, враховуючи необхідність регулярної ротації ключів у системах високого рівня безпеки;
- процес підпису: аналіз враховував вимірювання часу, необхідного для підпису JWT-токену, що дозволяє визначити, наскільки швидко кожен алгоритм здатний обробляти великий потік запитів на підписування;
- процес перевірки: досліджувалася продуктивність алгоритмів під час перевірки підписаних токенів, щоб оцінити здатність обробляти великі обсяги запитів на верифікацію в режимі реального часу.

Результати були зафіксовані на основі параметрів, рекомендованих рівнями безпеки NIST № 1 та № 3 [9, с. 56]. Ці рівні порівнюють стійкість шифрів до злому за допомогою атаки перебору ключів. Рівень безпеки NIST № 1 вимагає, щоб будь-яка атака на шифрувальну схему вимагала ресурсів, порівняних із перебором ключів для AES-128 [11]. На відміну від цього, рівень NIST № 3 вимагає ресурсів, еквівалентних тим, що потрібні для перебору ключів AES-192 [11], щоб зламати відповідну криптосистему. Довжини публічних ключів та підписів для кожної схеми цифрового підпису відповідно до різних рівнів безпеки NIST вказані в таблиці 2.

Таблиця 2

Відповідність рівням безпеки NIST № 1 та № 3 (розмір у байтах)

Критерії	**RSA**		**Dilithium**	
Рівень безпеки NIST	1	3	1	3
Розмір публічного ключа	384	512	1312	1952
Розмір підпису	384	512	2420	3309

Для порівняння швидкості генерації ключів було здійснено десять тисяч ітерацій процесу генерації. За результатами виконаних вимірювань було розраховано середнє значення часу, необхідного для генерації однієї пари ключів. Результати представлено в таблиці 3. Результати показують, що Dilithium демонструє значно вищу швидкість генерації ключів. Це пояснюється тим, що алгоритм побудований на криптографії на ґратках, яка забезпечує ефективну та швидку генерацію ключів навіть на високих рівнях безпеки. З іншого боку, RSA використовує великі модулі, які є добутками двох значних простих чисел. Зі збільшенням довжини ключа час його генерації також істотно зростає. Перевага алгоритму Dilithium у швидкості генерації ключів особливо помітна при високих рівнях безпеки, де традиційні методи, такі як RSA, стають значно повільнішими через збільшення розміру ключів. Це дозволяє алгоритму Dilithium забезпечувати надійний захист комунікації між мікросервісами, підтримуючи регулярну ротацію ключів навіть у масштабованих системах із високими вимогами до продуктивності.

Таблиця 3

Порівняння швидкості генерації ключів

	RSA		**Dilithium**	
Рівень безпеки NIST	1	3	1	3
Час (мілісекунди)	89,323	815,653	0,171	0,266

Для оцінки швидкодії підпису та верифікації було розроблено Python-скрипт, що виконує серію запитів до сервісу та будує діаграму на основі отриманих результатів. Цей скрипт дозволяє вимірювати затримки під час підпису та верифікації запитів, визначаючи середній час виконання кожної операції. Зібрані дані допомагають створити наочне порівняння продуктивності різних алгоритмів, виявити закономірності у їхній роботі, а також виявити потенційні вузькі місця, які можуть негативно впливати на швидкість обробки запитів у системі. Рисунок 2 показує кількість запитів на секунду для кожного з алгоритмів з параметрами, що відповідають рівням безпеки NIST № 1 та № 3. Для рівня безпеки NIST № 1 алгоритм Dilithium показує продуктивність підписування 459 запитів за секунду, що в 2,5 раза перевищує продуктивність RSA, який досягає лише 184 запити за секунду. У процесі перевірки підпису продуктивність Dilithium складає 502 запити за секунду, що в 1,7 раза перевищує продуктивність RSA, яка становить 296 запитів за секунду. На рівні безпеки NIST № 3 алгоритм Dilithium має продуктивність підписування в 320 запитів за секунду, що в 2,4 раза перевищує продуктивність RSA, яка дорівнює 132 запитам за секунду. У процесі перевірки підпису Dilithium показує 441 запит за секунду, що вдвічі перевищує продуктивність RSA, який забезпечує 225 запитів за секунду. Таким чином, результати показують, що алгоритм Dilithium демонструє істотну перевагу в продуктивності як під час підписування, так і перевірки підписів порівняно з RSA. Отже, алгоритм Dilithium стає оптимальним вибором для мікросервісних архітектур, де потрібна висока швидкість обробки запитів, забезпечуючи при цьому надійний рівень безпеки. Dilithium надає необхідну продуктивність, забезпечуючи ефективний обмін інформацією між мікросервісами без суттєвого компромісу щодо безпеки, сприяючи тим самим стійкості та масштабованості системи.

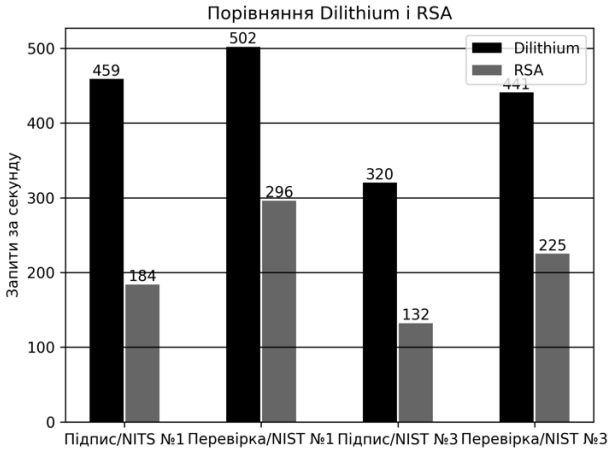

Рис. 2. Порівняння швидкодії підпису та верифікації

Приріст продуктивності досягається шляхом підвищення пропускної здатності. Як показано в таблиці 2, схеми цифрового підпису на основі криптографії на ґратках вимагають більшого розміру ключів і створюють підписи більшого розміру порівняно з RSA. Це збільшення розміру підпису може бути ресурсномістким через підвищене використання вхідної та вихідної пропускної здатності, що впливає на кожен запит API, оскільки JWT завжди передається для аутентифікації.

Висновки та перспективи подальших досліджень. Результати дослідження демонструють, що алгоритм Dilithium має суттєві переваги у швидкості підписування, верифікації та генерації ключів порівняно з RSA для різних рівнів безпеки NIST. Ці переваги обумовлені використанням криптографії на ґратках, яка забезпечує ефективну генерацію ключів і підписів навіть на високих рівнях безпеки. Проте значно більший розмір ключів та підписів Dilithium порівняно з RSA вимагає більшої пропускної здатності мережі. Отже, алгоритм Dilithium є перспективним вибором для підписування JWT-токенів у мікросервісних архітектурах, де важлива висока швидкість обробки запитів, надійність, безпека. Цей алгоритм сприяє ефективній передачі інформації між сервісами, зберігаючи масштабованість і стійкість системи.

Подальші дослідження можуть також охоплювати адаптацію алгоритму для різних мов програмування, що дозволить забезпечити ширшу сумісність, підвищити продуктивність та інтеграцію з різними платформами. Крім того, перспективним напрямом є дослідження інших квантово-стійких алгоритмів для порівняння їхньої ефективності та безпеки в різних середовищах. Це дозволить обрати оптимальні криптографічні методи, що підходять для широкого спектра систем та застосувань.

Ще одним важливим напрямом подальших досліджень є інтеграція квантово-стійких алгоритмів у специфікацію JWT, що гарантуватиме їх сумісність із сучасними стандартами безпеки та можливість захисту комунікації між мікросервісами.

Іншою перспективою є аналіз ефективності алгоритму Dilithium та інших квантово-стійких методів у різних комунікаційних протоколах для визначення їхньої продуктивності та рівня безпеки в умовах зростаючих квантових загроз. Це порівняння дозволить обрати оптимальні методи підпису та перевірки, сприяючи забезпеченню надійної та ефективної комунікації в системах із високими вимогами до продуктивності й безпеки.

References:

1. «NIST to Standardize Encryption Algorithms That Can Resist Attack by Quantum Computers» (2023), [Online], available at: https://www.nist.gov/news-events/news/2023/08/nist-standardize-encryption-algorithms-can-resist-attack-quantum-computers
2. Mattsson, J.P., Smeets, B. and Thormarker, E. (2021), «Quantum-Resistant Cryptography», [Online], available at: https://arxiv.org/pdf/2112.00399.pdf
3. Ducas, L., Kiltz, E., Lepoint, T. et al. (2017), «CRYSTALS-Dilithium: A Lattice-Based Digital Signature Scheme», [Online], available at: https://eprint.iacr.org/2017/633.pdf
4. Farik, M. and Ali, S. (2016), «The Need for Quantum-Resistant Cryptography in Classical Computers», [Online], available at: https://ieeexplore.ieee.org/document/7941947
5. «Introduction to JSON Web Tokens», [Online], available at: https://jwt.io/introduction
6. Jones, M. (2015), «JSON Web Algorithms (JWA)», [Online], available at: https://www.hjp.at/doc/rfc/rfc7518.html
7. «What is an API (Application Programming Interface)?», [Online], available at: https://aws.amazon.com/what-is/api/?nc1=h_ls
8. «Nest.js», [Online], available at: https://docs.nestjs.com
9. Barker, E. (2020), *Recommendation for Key Management: Part 1 – General*, [Online], available at: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-57pt1r5.pdf
10. «Microservices», [Online], available at: https://aws.amazon.com/microservices/
11. *Specification for the ADVANCED ENCRYPTION STANDARD (AES)* (2001), [Online], available at: https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197.pdf

Дацюк Денис Васильович – асистент кафедри комп’ютерної інженерії та кібербезпеки Державного університету «Житомирська політехніка».

https://orcid.org/0009-0009-2250-1694.

Наукові інтереси:

- інформаційні технології;
- технології розробки;
- вебтехнології;
- кібербезпека програмного забезпечення.

Россінський Юрій Михайлович – кандидат технічних наук, доцент, доцент кафедри комп’ютерної інженерії та кібербезпеки Державного університету «Житомирська політехніка».

https://orcid.org/0009-0009-6767-6666.

Наукові інтереси:

- програмування;

– комп’ютерна графіка;
– кібербезпека програмного забезпечення.

Воротніков Володимир Володимирович – доктор технічних наук, доцент, професор кафедри комп’ютерної інженерії та кібербезпеки Державного університету «Житомирська політехніка».
https://orcid.org/0000-0001-8584-3901.

Наукові інтереси:
– комп’ютерні мережі та мережні технології;
– мережна безпека;
– кібербезпека;
– керування складними інформаційними системами.

Datsiuk D.V., Rossinskyi Yu.M., Vorotnikov V.V.

Microservices Communication Security: Using Quantum-Resistant Algorithms for JWT Token Signing

In today's world, quantum computing is becoming a reality, which places increased demands on the security of communication between services. In this context, the use of quantum-resistant algorithms for signing JWT tokens in microservice architecture is extremely important. The comparison of the performance of lattice-based quantum-resistant algorithms with traditional RSA has the particular interest to determine the optimal token signing methods in terms of performance, security, and resistance to quantum threats. During the research, a series of requests to the service were made and charts were constructed to compare the delays during signing and verification of signatures. In particular, the latencies for each algorithm were measured to determine the average execution time of operations and to understand the impact of their performance on the operation of the microservice architecture. The research results show that the quantum-resistant Dilithium algorithm shows significantly higher performance in signing and verifying JWT tokens compared to RSA. This algorithm is built on lattice cryptography, which provides efficient generation of keys and signatures even with high levels of security. However, increasing the size of keys and signatures leads to increased use of network bandwidth, which should be considered during implementation. The Dilithium algorithm turns out to be a promising option for ensuring high performance and security in microservice systems. Its ability to rapidly generate and verify signatures facilitates efficient and reliable communication between microservices, while remaining resilient to future quantum threats. The use of quantum-resistant algorithms is becoming more and more relevant in light of the growing risks associated with the development of quantum computing.

Keywords: quantum-resistant cryptography; JWT (JSON Web Token); RSA; dilithium; lattice cryptography; microservice architecture; quantum threats.

Стаття надійшла до редакції 19.04.2024.