УДК 004.052.42 DOI: 10.18372/2073-4751.86.21279

Марковський О. П.,
orcid.org/0000-0003-3483-4233
e-mail: markovskyy@i.ua,

Череватенко О. В.,
orcid.org/0000-0001-9686-0555,
e-mail: chereva@ukr.net,

Вовк В. В.,
orcid.org/0009-0002-2303-3131,
e-mail: vovk.vlad@lll.kpi.ua

ОРГАНІЗАЦІЯ ПРОТИДІЇ АТАКАМ НА КРИПТОГРАФІЧНІ КЛЮЧІ АНАЛІЗОМ ДИНАМІКИ СПОЖИВАННЯ ПОТУЖНОСТІ ТЕРМІНАЛЬНИМИ ПЛАТФОРМАМИ IOT

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

Вступ

Однією з характерник ознак розвитку інформаційних технологій в останнє десятиліття стало динамічне розширення використання систем віддаленого управління об'єктами реального світу на базі технологій Інтернету речей (Internet of Things – IoT) [1].

Теоретична концепція IoT створювалась понад двадцять років тому, як спосіб дешевого дистанційного управління побутовими пристроями, що використовує для обміну даних існуючу інфраструктуру глобальних мереж. Їх технічні характеристики цілком задовольняли потребам управління побутовою технікою. Для цього типу застосувань використання потенційно відкритого середовища Інтернету не створювало істотних проблем з точки зору безпеки.

За останні двадцять років технічні можливості Інтернету досягли якісно більш високого рівня: на порядки збільшилась швидкість передачі даних, зросла надійність. Ці чинники стали рушійною силою широкого застосування технології IoT в різних сферах людської діяльності.

За рахунок простоти, дешевизни та широкого покриття Інтернету, на сьогоднішній день, його використовують у якості середовища обміну даними близько 60% систем дистанційного управління.

Разом з тим, широке застосування потенційно вразливого середовища спричиняє ризики несанкціонованого зовнішнього втручання у процес віддаленого управління. Особливо гостро проступає питання безпеки у контексті тих галузей, для яких зовнішнє втручання несе за собою значні збитки.

У якості прикладів таких сфер можна вказати на моніторинг через Інтернет стану пацієнтів, дистанційні автоматизовані дозатори ліків, віддалене управління інфраструктурними об'єктами, безпілотні транспортні засоби, військові засоби безпілотної розвідки, ураження, евакуації тощо.

Для цих застосувань принципово важливо виключити можливість зовнішнього втручання у процеси управління. На практиці, такі втручання проявляються у формі фальсифікації даних про стан віддалених об'єктів або надсилання хибних команд управління сторонніми особами.

Іншими словами, зовнішнє втручання зводиться до порушення цілісності та автентичності даних, що передаються через середовище Інтернет у таких системах управління.

Класичним засобом запобігання зазначеним ризикам виступає механізм цифрового підпису [2]. Всі відомі алгоритми цифрового підпису базуються на операції модулярного експоненціювання, що виконуються над числами великої розрядності, що на сьогоднішній день стандартизована NIST не меншою, ніж 4096 біт.

Реалізація операції модулярного експоненціювання при такій розрядності потребує близько 10^{11} процесорних операцій. Для малопотужних термінальних мікроконтролерів зазначений об'єм обчислень потребує значних ресурсів, що може стати на заваді забезпеченню функціонування систем управління в реальному часі. Такий стан визначає необхідність пошуку нових шляхів для прискорення виконання операції модулярного експоненціювання.

Водночас, при реалізації модулярного експоненціювання за класичним алгоритмом, послідовність операцій прямо залежить від значення поточного біту експоненти, яка на практиці є секретним ключем. Це створює ідеальні умови для застосування технології реконструкції секретних операндів аналізом динаміки споживання потужності.

Виходячи з цього, можна зробити висновок про те, що сучасний етап розвитку систем віддаленого управління на базі IoT об'єктивно вимагає пошуку нових підходів та методів до прискорення операції модулярного експоненціювання та захисту її операндів від технології реконструкції даних шляхом аналізом динаміки споживання потужності.

Таким чином, наукова задача підвищення рівня захищеності від атак, що здійснюються шляхом аналізу динаміки споживання потужності процесором, на якому реалізується базова операція криптографії з відкритим ключем – модулярного експоненціювання та її прискорення є актуальною з огляду на особливості сучасного етапу розвитку систем віддаленого управління на базі технологій IoT.

Аналіз вразливості операції модулярного експоненціювання до атак аналізом динаміки споживання потужності

Базова операція обчислення модулярної експоненти $A^E \bmod M$ над n-розрядними числами реалізується за одним з двох різновидів класичного алгоритму [2]. Останній полягає в виконанні n циклів, послідовність операцій у яких визначається значенням бітів експоненти: $E = 2^n \cdot e_n + ... + e_1 + 2 \cdot e_2 + 4 \cdot e_3 + ... + 2^{n-1} \cdot e_n$, $\forall i \in \{1,2,...,n\}$: $e_i \in \{0,1\}$. В залежності від Послідовність, в якій здійснюється аналіз бітів коду експоненти E визначає один із двох різновидів класичного алгоритму обчислення модулярної експоненти.

Перший різновид полягає у скануванні коду експоненти E у напрямку зі старших розрядів до молодших. Він передбачає використання однієї змінної R, в якій формується результат модулярного експоненціювання і котра перед початком циклу встановлюється рівною одиниці: $R = 1$. В кожному i-ому циклі ($\boldsymbol{i}$ пробігає значення від n до 1) реалізується модулярне піднесення до квадрату поточного значення результату R: $R = R^2 \bmod M$. При значенні $e_i = 1$, змінна результату R множиться на значення A по модулю M: $R = R \cdot A \bmod M$.

Ключовою відмінністю другого різновиду класичного алгоритму модулярного експоненціювання виступає використання двох змінних: D та R, яким перед початком циклу присвоюються значення A та одиниці, відповідно: $D = A$, $R = 1$. При цьому змінна D використовується для обчислення послідовних значень A у степенях двійки по модулю M.

Проходження розрядів експоненти E реалізується у зворотному напрямку: від молодших до старших. В кожному i-тому циклі (i змінюється від 1 до n) при одиничному значенні поточного біту експоненти (e_i = 1) здійснюється модулярне множення поточного значення результату R та змінної D: $R = R \cdot D \bmod M$. Після цього, реалізується модулярне піднесення до квадрату змінної D: $D = D^2 \bmod M$. По проходженню всіх n бітів експоненти, результат модулярного експоненціювання $A^E \bmod M$ розміщується у змінній R.

Аналіз обох розглянутих різновидів класичного алгоритму свідчить про те, що порядок виконання операцій суттєвим чином залежить від розрядів коду експоненти. Це зумовлює потенційну небезпеку успішних атак на реалізацію обчислення модулярної експоненти шляхом вимірювання та аналізу динаміки споживання потужності обчислювальною платформою, на якій реалізується ця операція.

Сучасні технології реконструкції даних шляхом аналізу динаміки споживання потужності обчислювальними платформами, на яких ці дані оброблюються, поділяються на ряд різновидів: простий аналіз споживання потужності (Simple Power Analysis - SPA), диференційний аналіз (Differential Power Analysis – DPA) та кореляційний аналіз споживання потужності (Correlation Power Analysis - CPA) [3]. Перший, найпростіший і, разом з тим, найбільш ефективний із наведених різновидів орієнтований на ситуації коли дані, що підлягають реконструкції, впливають на послідовність команд програми. Відповідно, технологія SPA базується на розпізнаванні команд за осцилограмою споживання потужності. Послідовність команд дозволяє реконструювати дані, від яких ця послідовність залежить. Цілком очевидно, що в обох різновидах класичного алгоритму обчислення модулярної експоненти послідовність операцій модулярного множення та модулярного піднесення до квадрату повністю визначає розряди секретного коду експоненти Е. Тобто, базова задача SPA при порушенні захисту криптографії з відкритим ключем полягає в розпізнаванні за осцилограмою споживання потужності операцій модулярного множення та модулярного піднесення до квадрату [4]. При розрядності 4096 ці операції реалізуються послідовностями з сотень мільйонів процесорних команд [5] і за даними [4] успішно розпізнаються. Особливо вразлива до SPA реалізація криптографічних протоколів з відкритим ключем на малопотужних термінальних мікроконтролерах систем дистанційного контролю та управління, що побудовані з використанням технологій IoT[6]. На практиці в таких системах найчастіше застосовуються для захисту від зовнішнього втручання в їх роботу механізми цифрового підпису DSA, базовою обчислювальною операцією яких є модулярне експоненціювання.

З наведеного випливає, що реалізація криптографічних протоколів з відкритим ключем в побудованих за технологією IoT системах контролю віддалених об'єктів, до яких потенційно може мати фізичний доступ зловмисник, потребує спеціальних засобів захисту.

До теперішнього часу запропоновано ряд методів та підходів, які дозволяють або повністю виключити можливість застосування технологій SPA для зламу механізмів захисту даних на основі криптографії з відкритим ключем, або на порядки зменшити її ефективність.

Ще один підхід до зменшення ефективності SPA полягає в зміщенні в часі виконання операцій модулярного множення [7], що утруднює прив'язку виконання цих операцій до значень розрядів секретного коду експоненти. Реалізація цього підходу потребує пам'яті для тимчасового зберігання операндів операції модулярного

множення. При застосуванні методу [8] існує можливість реконструкції бітів експоненти за виявленням операції запису операндів множення в пам'ять. Сама ця операція запису двох n-розрядних чисел складається на практиці з близько сотні команд, які можуть бути успішно розпізнані. Тобто метод [8] лише ускладнює проведення SPA. В роботі [9] запропоновано удосконалений варіант цього підходу, який практично виключає можливість застосування SPA, проте не виключає можливості реконструкції коду експоненти з використанням CPA [4].

В якості дієвого методу протидії SPA можна розглядати розподілене обчислення модулярної експоненти на термінальних мікроконтролерах систем дистанційного контролю на базі IoT з залученням хмарних технологій [10]. Крім захисту від SPA цей підхід дозволяє на порядок прискорити обчислення модулярної експоненти. З іншого боку, відомі технології розподіленого модулярного експоненціювання орієнтовані на концентрації частини обчислень, пов'язаної з обробкою секретного коду експоненти саме на термінальному мікроконтролері. Це відкриває певні можливості ефективного реконструювання коду експоненти методами аналізу динаміки споживання потужності термінальним мікроконтролером.

Один із найбільш ефективних підходів до зменшення дієвості застосування SPA до базової операції криптографії з відкритим ключем – модулярного експоненціювання $A^E \bmod M$, полягає в організації обробки коду експоненти E групами по k розрядів, починаючи зі старших, з передобчисленнями всіх 2^k значень A, $A^2 \bmod M$, $A^3 \bmod M$,..., $A^{2^k-1} \bmod M$, що зберігаються в таблиці. Експоненціювання організовано у вигляді n/k циклів, в кожному із яких попереднє значення поточного результату R k раз підноситься до квадрату A і множиться на табличне значення, адресація якого здійснюється значенням поточної групи розрядів коду експоненти [11]. Тобто при обробці групи з k розрядів операція модулярного множення, незалежно від k-бітового коду групи виконується точно один раз. Це ускладнює відновлення бітів коду експоненти за фактом виконання операції модулярного множення. Проте, методами CPA можна встановити [4] факт ідентичності k-бітових фрагментів коду E. В силу того, що код експоненти E являє собою частину закритого ключа, тобто є практично незмінним, технологіями DPA [12] можна встановити послідовність індексів табличних значень, що використовуються в процесі обчислення модулярної експоненти. Тобто, вказаний підхід суттєвим чином ускладнює незаконну реконструкцію секретного коду експоненти E методами аналізу споживання потужності, але не виключає її успішне здійснення.

Проведений аналіз відомих методів протидії незаконній реконструкції ключів криптографічних алгоритмів, що базуються на операції модулярного експоненціювання, технологією SPA, засвідчив, що вони лише знижують ефективність її застосування, і при цьому, практично не блокують можливість ефективного використання альтернативних технологій зламу ключових елементів аналізу динаміки споживання потужності, таких як CPA та DPA.

З огляду на те, що технології CPA та DPA базуються на статистичній обробці динаміки споживання потужності при реалізації одного і того ж самого криптографічного алгоритму, найбільш дієвий спосіб захисту від цих технологій полягає в організації поліморфної реалізації криптографічних алгоритмів [13]. Аналіз свідчить про те, що така реалізація може бути виконана для базової процедури криптографічних алгоритмів з відкритим ключем – модулярного експоненціювання.

Мета досліджень

Мета досліджень полягає в підвищенні ефективності захисту від атак, що здійснюються аналізом динаміки споживання потужності комп'ютерними платформами при реалізації модулярного експоненціювання – базової операції криптографії з відкритим ключем і направлені на незаконну реконструкцію секретних ключів, за рахунок організації поліморфної обробки розрядів коду експоненти групами зі змінною довжиною.

Організація поліморфного обчислення модулярної експоненти

Для досягнення поставленої мети запропонована організація поліморфної реалізації модулярного експоненціювання, в основі якої покладено різновид класичного алгоритму з аналізом розрядів коду експоненти починаючи зі старших. Як і в методі [11] пропонується організувати групову обробку розрядів коду експоненти. На відміну від згаданого методу, в якому довжина кожної групи постійна, пропонується зробити довжину груп змінною: тільки за цієї умови можна організувати поліморфне обчислення модулярної експоненти.

Згідно з запропонованим методом в коді E експоненти виділяються групи $G_1,G_2,\ldots,G_m$ розрядів, кожна із яких містить відповідно $\eta_1,\eta_2,\ldots,\eta_m$ бітів. На відміну від методу, запропонованого в роботі [14] запропонована організація передбачає наявність в коді експоненти окремих нульових чи одиничних бітів, що не належать ні одній із груп $G_1,G_2,\ldots,G_m$. Фактично, процес обробки групи розрядів $e_h,e_{h-1},\ldots,e_{h-}$ коду експоненти E, $h\in\{\eta+1,\eta+2,\ldots,n\}$, за умови, що поточний результат дорівнює R, розкладається на три фази:

- послідовне піднесення поточного результату R до модулярного квадрату η раз;
- обчислення $Y=A^F \bmod M$, де $F=e_{h-\eta}+2\cdot e_{h-\eta+1}+\ldots+2^{\eta-2}\cdot e_{h-1}+2^{\eta-1}\cdot e_h$;
- обчислення нового значення поточного результату R у вигляді модулярного добутку: $R=R\cdot Y \bmod M$;

Виконання першої фази пропонується реалізувати шляхом реалізації обчислення модулярного квадрату R на кожному із n кроків експоненціювання. Виконання другої фази пропонується шляхом використання передобчислень. Виходячи із того, що використання передобчислень не має мати наслідком сповільнення модулярного експоненціювання, кількість ε одиничних бітів в групі має бути більшою одного, тобто два і більше: $\varepsilon>1$. З тих же міркувань очевидно, що перший та останній біт групи мають дорівнювати одиниці.

Фактично, при обчисленні значення Y виконується η операцій модулярного піднесення до квадрату та ε операцій модулярного множення. При цьому η операцій модулярного піднесення до квадрату фактично дублюють аналогічну кількість таких операцій, що здійснюються в рамках виконання першої фази обробки групи. Це означає, що якщо передобчислення використовуються лише один раз в процесі модулярного експоненціювання. то це має наслідком сповільнення обчислень. Часова ефективність передобчислень визначається економією операцій модулярного множення при багатократному їх використанні в процесу модулярного експоненціювання. Вона забезпечується, якщо число зекономлених операцій модулярного множення перевищує кількість η дублюючих операцій модулярного піднесення до квадрату при здійсненні передобчислень: $(N-1)\cdot\varepsilon > \eta$, тобто, кількість N повторів групи в коді експоненти E має задовольняє умові

$$N > \frac{\eta}{\varepsilon} + 1. \qquad (1)$$

При цьому, середня кількість $N(\eta)$ однакових бітових груп довжиною η за

умови, що $n>>2^{\eta}$ визначається за формулою:

$$N(\eta)=\frac{n}{2^{\eta}}. \qquad (2)$$

В криптографічних механізмах цифрового підпису код експоненти E являє собою закритий ключ, який міняється відносно рідко [15], так, що його можна вважати сталим. Це надає значні резерви часу для підбору найбільш ефективного набору груп розрядів постійного коду експоненти E, для яких здійснюється передобчислення.

Наприклад, при розрядності n=12 і значенні коду експоненти E = 3031 =101 111 010 111_2 може бути використані одна група (m=1): G_1={1011}, для якої η_1=4, або група G_1 = {111}, для якої η_1=3, або дві групи (m=2): G_1={101} та G_2= {11}, довжини η_1 і η_2 яких дорівнюють відповідно: η_1=3 та η_2=2.

Для безпосереднього обчислення експоненти пропонується використовувати дві таблиці. Перша T з них містить в собі сформовані перед кожним модулярним експоненціюванням $A^E \bmod M$ коди $Y_1,Y_2,\ldots,Y_m$ результатів обчислення часткових модулярних експонент , показником для яких слугують двійкові коди груп $G_1,G_2,\ldots,G_m$:

$\forall l \in \{1,2,...,m\}$:

$$T[l]=A\cdot A^{g_1+2\cdot g_2+\ldots+2^{m-2}\cdot g_{\eta}}\cdot A^{2^{m-1}} \bmod M \quad (3)$$

Наприклад, якщо A=2026, модуль M = 3953 = 1111 0111 0001_2, а G_1={101} та G_2= {11}, то $T[1] = Y_1 = A^5 \bmod M = 2026^5 \bmod 3953 = 1433$; $T[2] = Y_1 = A^3 \bmod M = 2026^{32} \bmod 3953 = 1215$. Нульовий елемент таблиці T містить в собі код числа A, на яким здійснюється операція модулярного експоненціювання: $T[0]=A=2026$.

Друга, з запропонованих до використання таблиць – Q містить n записів, кожен із яких відноситься до відповідного, починаючи за старших, біту коду експоненти. Кожен із записів складається із тегового біту b та $\log_2 m$-розрядного коду u адресації таблиці T. Теговий біт b_j, $j\in\{1,2,,...,n\}$ який дорівнює нулю коли обробка j-го біті експоненти полягає лише в модулярному піднесенні до квадрату поточного результату: $R=R^2 \bmod M$; якщо тегів біт b_j = 1, то обробка j-го розряду коду E експоненти полягає в здійсненні двох модулярних мультиплікативних операцій: модулярного піднесення до квадрату поточного результату: $R=R^2 \bmod M$ та модулярного множення отриманого в результаті цієї операції значення R на табличне значення $T[u_j]$, яке адресується полем коду u_j таблиці Q: $R=R\cdot T[u_j] \bmod M$.

Викладене вище може бути ілюстровано прикладом таблиці Q для коду E експоненти E = 3031 =101 111 010 111_2 за умови виділення груп G_1={101} та G_2= {11}, яка представлена в таблиці 1.

Таблиця 1. Приклад таблиці Q для E =3031, m=2, n=12, G_1={101}, G_2= {11}.

Номер j біту E	Теговий біт b_j	Адреса u_j
12	0	-
11	0	-
10	1	1
9	0	-
8	1	2
7	0	-
6	0	-
5	1	1
4	0	-
3	1	0
2	0	-
1	1	2

Відповідно, запропонована процедура обчислення модулярної експоненти з використанням двох таблиць T і Q в формалізованому вигляді зводиться до виконання наступної послідовності дій:

1. Значення змінної поточного результату R встановлюється в одиницю: R=1; значення індексу j таблиці Q встановлюється рівним n: $j=n$.
2. Здійснюється модулярне піднесення до квадрату поточного значення R: $R = R^2 \bmod M$.

3. Якщо теговий біт b_j, який зберігається в $Q[j]$ дорівнює одиниці, тобто b_j=1, то полем u_j табличного значення $Q[j]$ адресується таблиця T, з якої зчитується код $T[u_j]$, що множиться по модулю M на поточний результат R: $R = R \cdot T[u_j] \bmod M$.
4. Значення індексу j зменшується на одиницю: $j=j-1$. Якщо $j > 0$, здійснюється повернення на повторне виконання п.2.

Робота запропонованої процедури може бути ілюстрована прикладом обчислення модулярної експоненти $2026^{3031} \bmod 3953 = 1255$. В цьому прикладі A=2025, модуль M=3953, n=12, E=3031. Якщо для передобчислень обрано групи G_1={101} та G_2= {11}, то порядок обчислення експоненти визначається таблицею Q, заповнення якої представлено в таблиці 1. Тоді, динаміка покрокових змін поточного результату R в ході виконання запропонованої процедури модулярного експоненціювання представлено в таблиці 2.

Отримане в останньому рядку таблиці 2 значення R=1255 свідчить про те, що обчислення виконані правильно. В процесі обчислення виконано n=12 операцій модулярного піднесення до квадрату та 5 операцій модулярного множення. При здійсненні модулярного експоненціювання за класичним алгоритмом здійснюється 9 операцій модулярного множення при 12-ти операціях модулярного піднесення до квадрату. Тобто в рамках наведеного прикладу видно, що запропонована організація дозволяє не тільки захистити найбільш секретну компонент операції, але й прискорити її виконання.

Таблиця 2. Динаміка покрокових змін поточного результату R при обчисленні $2026^{3031} \bmod 3953$ з передобчисленням груп G_1={101} та G_2= {11}

Номер j біту E	Теговий біт b	Зміна значення R	
		Піднесення до квадрату	Множення на табл..значення
12	0	$1^2 \bmod M = 1$	-
11	0	$1^2 \bmod M = 1$	-
10	1	$1^2 \bmod M = 1$	$1 \cdot T[1] \bmod M = 1433$
9	0	$1433^2 \bmod M = 1882$	-
8	1	$1882^2 \bmod M = 36$	$36 \cdot 1215 \bmod M = 257$
7	0	$257^2 \bmod M = 2801$	-
6	0	$2801^2 \bmod M = 2849$	-
5	1	$2849^2 \bmod M = 1292$	$1292 \cdot 1433 \bmod M = 1432$
4	0	$1432^2 \bmod M = 2970$	-
3	1	$2970^2 \bmod M = 1757$	$1757 \cdot 2026 \bmod M = 1982$
2	0	$1982^2 \bmod M = 2995$	-
1	1	$2995^2 \bmod M = 668$	$668 \cdot 1215 \bmod M = 1255$

З точки зору організації захисту від відновлення коду експоненти відслідковуванням по діаграмі споживання потужності операцій модулярного множення запропонований метод на порядки утруднює реалізацію такої атаки.

Її можливість може бути практично виключена шляхом застосування поліморфізму, тобто зміни порядку обчислень модулярної експоненти за

рахунок використання різних наборів груп.

Цілком очевидно, що, в силу незмінності коду експоненти E, для обчислення A^E mod M можуть бути використані передобчислення різних груп. Наприклад, в рамках розглянутого вище прикладу, тобто при розрядності n=12 і значенні коду експоненти E = 3031 =1011 1 1 0 10 11 1_2 може бути використані одна група (m=1): G_1={1011}. Тоді таблиця T передобчислень містить лише два значення: $T[0]=A=2026$ та $T[1]=A^{11} \text{mod} M=2026^{11}$ mod 3953 = 2240. Відповідно, таблиця Q для такої організації обчислення модулярної експоненти має вигляд, представлений в таблиці 3.

Таблиця 3. Приклад таблиці Q для E =3031, m=1, n=12, G_1={1011}.

Номер j біту E	Теговий біт b_j	Адреса u_j
12	0	-
11	0	-
10	0	-
9	1	1
8	1	0
7	1	0
6	0	-
5	0	-
4	0	-
3	0	-
2	1	1
1	1	0

Таблиця 4. Динаміка покрокових змін поточного результату R при обчисленні 2026^{3031} mod 3953 з передобчисленням групи G_1={1011}.

Номер j біту E	Теговий біт b	Зміна значення R	
		Піднесення до квадрату	Множення на табл..значення
12	0	1^2 mod M =1	-
11	0	1^2 mod M =1	-
10	0	1^2 mod M =1	-
9	1	1^2 mod M =1	1·2240 mod M = 2240
8	1	2240^2 mod M = 1243	1243·2026 mod M = 257
7	1	257^2 mod M = 2801	2801·2026 mod M = 2271
6	0	2271^2 mod M = 2729	-
5	0	2729^2 mod M = 3942	1292·1433 mod M = 1432
4	0	3942^2 mod M = 121	-
3	0	121^2 mod M = 2782	-
2	1	2782^2 mod M = 3503	3503·2240 mod M = 15
1	1	15^2 mod M = 225	225·2240 mod M = 1255

Динаміка змін поточного результату при обчислення модулярної експоненти 2026^{3031} mod 3953 = 1255 запропонованим методом з передобчисленням однієї групи G_1={1011}. відображається таблицею 4. Порівняльний аналіз даних таблиць 2 і 4 свідчить про те, що при порядок обчислень зазнав суттєвих змін при тому. що в кінцевому підсумку отримано ідентичний результат. Тобто в рамках прикладу

показано функціонування запропонованого методу поліморфної реалізації фундаментальної для криптографічних застосувань операції модулярного експоненціювання.

Оцінка ефективності

Запропонований метод обчислення модулярної експоненти дозволяє ефективність захисту від атак на секретні ключі аналізом динаміки споживання потужності термінальними платформами IoT при реалізації на них алгоритмів на базі криптографії з відкритим ключем. Підвищення ефективності відбувається за рахунок як підвищення рівня захищеності, так і за рахунок прискорення реалізації фундаментальної обчислювальної операції цього виду криптографії – модулярного експоненціювання, що виконується над числами, довжина яких на порядки перевищує розрядність процесора. З 2024 вимогами NIST [10] встановлено мінімальну довжину чисел для використання в криптографії з відкритим ключем рівну n = 2048.

Підвищення рівня захищеності реалізується за рахунок поліморфної організації обчислень фундаментальної операції криптографії з відкритим ключем – модулярного експоненціювання. Цілком очевидно, що при застосуванні передобчислених значень для обраних груп розрядів постійного коду експоненти E пряма реалізація технології SPA для незаконної реконструкції коду експоненти E не може бути використана. Це зумовлено тим, що операції модулярного множення не співвідносяться зі значеннями бітів коду експоненти E. Разом з тим, застосування статистичних методів обробки результатів динаміки споживання потужності теоретично дає можливості суттєвого зменшення обсягу перебору при підборі можливих кодів груп [11]. Для виключення такої можливості запропоновано поліморфна реалізація модулярного експоненціювання. При цьому, зміна груп розрядів коду експоненти здійснюється при кожному обчисленні, що виключає можливість застосування статистичних методів DPA.

Разом з тим, передобчислення для груп, що повторюються в коді експоненти E, дозволяє прискорити процес обчислення за рахунок зменшення числа модулярних множень. Ефект прискорення залежить від кількості □ одиниць в групі та числа N повторень групи в коді експоненти. При цьому, залежність між довжиною □ групи та N визначається формулою (2). Цілком очевидно, що максимальне значення N досягається при □=11. Це дає тривіальне рішення вибору груп з точки зору прискорення обчислень. Це рішення полягає в використані групи G ={1,1} та □ груп, що містять нуля між крайніми одиницями. Для реальної розрядності коду експоненти n=4096 ефективність такого вибору груп досліджувала експериментально. Досліди показали, що кількість операцій множення, що економляться при використанні передобчислень близька до 900. Тобто, з загальної середньої кількості 6144 операцій модулярного множення при класичній реалізації, в середньому виключаються 900 операцій, що дозволяє прискорити реалізацію криптографію з відкритим ключем на термінальних мікроконтролерах IoT на 17%.

Проте таке рішення не дозволяє генерувати велику кількість варіантів вибору груп. В результаті експериментальних досліджень показано, що для розрядності 4096 оптимальний вибір груп має включати 5-6 груп довжиною 10 біт і групу бітів, для покриття всіх залишкових одиниць коду експоненти. В таблиці 5 наведено приклад такого вибору груп для реального 4096-бітового коду експоненти E.

Таблиця 5. Приклад вибору груп для реального 4096-бітового коду експоненти *E*.

Номер групи	Код групи	Кількість повторів	Кількість множень, що економляться
1	1010111101	12	85
2	1101110011	11	77
3	1010011111	10	70
4	1111011011	9	81
5	1011101101	7	49
6	1000001	6	6
7	100001	4	4
8	100000001	3	2
9	1000000001	1	0

Очевидно, що при наведено в таблиці 5 виборі груп, число операцій модулярного множення, що економляться складає 374, що відповідає коефіцієнту прискорення обчислення модулярної експоненти, рівному 1.07.

Висновки

В результаті досліджень, направлених на підвищення ефективності захисту від атак, що здійснюються аналізом динаміки споживання потужності комп'ютерними платформами IoT при реалізації модулярного експоненціювання – базової операції криптографії з відкритим ключем і направлені на незаконну реконструкцію секретних ключів, запропоновано метод, що забезпечує захист від таких атак.

Метод обчислення модулярної експоненти зі старших розрядів коду ступеня, який відрізняється тим, що обробка його бітів організована групами зі змінною довжиною, часткові експоненти для яких реалізуються в рамках передобчислень, за рахунок чого виключається зв'язок між значеннями бітів ступеня і операціями модулярного множення, які можуть бути розпізнані за динамікою споживання потужності при реалізації модулярного експоненціювання. Відмінність запропонованого методу полягає в поліморфній організації модулярного експоненціювання шляхом вибору різних наборів груп при кожному обчисленні. Це виключає можливість незаконної реконструкції коду експоненти методами статистичного аналізу.

Показано, що запропонований метод, крім підвищення рівня захищеності забезпечує незначне 7-15% прискорення процесу комп'ютерної реалізації базової операції криптографії з відкритим ключем – модулярного експоненціювання.

Розроблений метод орієнтовано для забезпечення інформаційної безпеки побудованих на базі технології IoT систем контролю віддаленими об'єктами, до яких потенційно можливим є фізичний доступ сторонніх осіб.

Література

1. Patel B.C. IoT an Overview: Advantage, Disadvantage and Applications / Bhagwari Charan Patel, Ram Dhankar Tripathi, Naveen Goel // International Journal of Computer Applications Technology and Research. Vol. 10.- № 5.- P.119-122.

2. Schneier B. Applied Cryptography. Protocols. Algorithms and Source codes in C. Ed.John Wiley, 1996 - 758 p.

3. Mangard S. Power Analysis Attacks: Revealing the Secrets of Smart Cards / Stefan Mangard, Elisabeth Oswald, Thomas Pop //

Springer-Verlag.- 2007. – P. 338. DOI: 10.1007/978-0-387-38162-6.

4. Randolp M. Power Side-Channel Atttack Analysis: A Review of 20 Years of Study for the / Mark Randolph, William Diehl // Cryptography.- 2020.- Vol.4- № 4. - P.1-33. DOI: 10.3390/ cryptography4020015.

5. Марковський О.П.Метод прискорення модулярного піднесення до квадрату довгих чисел для криптографічних застосувань / О.П. Марковський , Аль-Мраят Гассан Абдель Жаліль // Проблеми управління та інформатизації.- 2024.- № 1 (77).- С.68-79. DOI: 10.18372/2073-4751.77.18659

6. Messerges T.S. Examining smart-cart security under the threat of power analysis attacks / T.S. Messerges, E.A. Dabbish, R.H. Sloan // IEEE Transaction on Computers.- 2002.- Vol.51.- № 5. - P.541-552. DOI: 10.31109/TC.2002.1004593.

7. Марковський О.П. До проблеми захисту операндів модулярного експоненціювання від їх реконструкції аналізом споживання потужності // О.П. Марковський, А.А.Зюзя, Мухаммад Мефлех Алиса Абабне, В.М. Гаразд // Вісник НТУУ. Інформатика, управління та обчислювальна техніка 2007.- Вип.47.- С.22-32.

8. Clavier C. Universal exponentiation algorithm - A first step to toward provable SPA-resistance / C. Clavier, M. Joye // Proceeding of 3-th International Workshop "Cryptographic Hardware and Embedded Systems"(CHES-2001), - 2001. LNCS-2162,-P. 300-308.

9. Русанова О.В. Метод модулярного експоненціювання з захистом від атак аналізом динаміки споживання потужності / О.В. Русанова, О.П. Марковський, В.В. Вовк //. Проблеми управління та інформатизації.- 2024.- № 4 (80).- С.93-103. DOI: 10.18372/2073-4751.80.19774.

10. Markovskyi O. A Secure Cloud Computing Approach for Rapid Implementation of Public Key Cryptography on IoT Terminal Devices / O. Markovskyi, M. Haidukevych, J. Borges, N. Serhiichuk // 14th International Conference on Dependable Systems, Services and Technologies DESSERT-2024, Athens, Greece, -2024,- P. 55-59, DOI: 10.1109/DESSERT65323.2024.11122255.

11. Markovskyi O.P. Method for Power Analysis-Proof Implementation of Modular Exponentiation on IoT Terminal Microcontrollers / O.P. Markovskyi, Jose Borges, Nazar Serhiichuk, N.G. Bardis //14-th International Conference on Dependable system, Service and Technologies DESSERT-2024, Greece, Athens.-2024.-P.248-253. - DOI: 10.1109/DESSERT65323.2024.11122248.

12. Kocher P. Differential Power Analysis / P. Kocher, J.Jaffe, B.Jun // Proceeding of CRYPTO'99.-Springer-Verlag.-1999.- P.388-404.

13. Bardis N.Organization of the polymorphic implementation of Rijndael on microcontrollers and smart cards / N/Bardis, N. Doukas, O.P. Markovskyi // Conference Military Communication – Milcom-2010. USA-2010.- P.43-51. DOI: 10.1109/ MILCOM.2010.5680249.

14. Wells J.B. A calculus with polymorphic and polyvariant flow types / J.B. Wells, A. Dimock, R. Muller, F.Turback // Journal of Functional Programming.- 2002.- Vol.12.- 3.- P.183-227. DOI: 10.1017/ S095679680 1004245.

15. Markovskyi O. An Accelerate Approach for Public Key Cryptography Implementation on IoT Terminal Platforms / O. Markovskyi, Al-Vrayat Ghassan Abdel Jalil Halil, Nikolaos Doukas, Nikos Bardis // 13-th International Conference on Dependable system, Service and Technologies DESSERT-2023, 13-15 October, Greece, Athens. DOI: 10.1109/DESSERT61349.2023.104165

Марковський О. П., Череватенко О. В., Вовк В. В.

ОРГАНІЗАЦІЯ ПРОТИДІЇ АТАКАМ НА КРИПТОГРАФІЧНІ КЛЮЧІ АНАЛІЗОМ ДИНАМІКИ СПОЖИВАННЯ ПОТУЖНОСТІ ТЕРМІНАЛЬНИМИ ПЛАТФОРМАМИ IOT

У статті розглянуто проблему підвищення захищеності операції модулярного експоненціювання, що є базовою для криптографії з відкритим ключем у системах дистанційного керування на базі технології IoT, від атак, заснованих на аналізі динаміки споживання потужності.

Показано, що класичні алгоритми модулярного експоненціювання є вразливими до SPA, DPA та CPA, оскільки послідовність виконуваних операцій залежить від бітів секретного коду експоненти. Запропоновано метод поліморфного обчислення модулярної експоненти, у якому обробка розрядів коду експоненти здійснюється групами змінної довжини з використанням передобчислень. Застосування різних наборів груп при кожному обчисленні усуває прямий зв'язок між значеннями бітів експоненти та операціями модулярного множення, що ускладнює реалізацію атак аналізом споживання потужності та виключає можливість статистичної реконструкції коду експоненти.

Доведено, що запропонований метод, крім підвищення рівня захищеності, забезпечує прискорення комп'ютерної реалізації модулярного експоненціювання на 7–15%, що є важливим для малопотужних термінальних IoT-платформ.

Ключові слова: модулярне експоненціювання; криптографія з відкритим ключем; аналіз споживання потужності; поліморфне обчислення; передобчислення; IoT.

Markovskyi O. P., Cherevatenko O. V., Vovk V. V.

ORGANIZATION OF COUNTERMEASURES AGAINST ATTACKS ON CRYPTOGRAPHIC KEYS BY ANALYZING THE POWER-CONSUMPTION DYNAMICS OF IoT TERMINAL PLATFORMS

The paper addresses the problem of improving the protection of modular exponentiation, which is the basic operation of public-key cryptography in IoT-based remote control systems, against attacks based on power-consumption analysis. It is shown that classical modular exponentiation algorithms are vulnerable to SPA, DPA and CPA because the sequence of executed operations depends on the bits of the secret exponent.

A method of polymorphic modular exponentiation is proposed, in which the exponent bits are processed in variable-length groups using precomputations. The use of different group sets for each computation eliminates the direct relationship between the exponent bits and modular multiplication operations, which complicates power-analysis attacks and prevents statistical reconstruction of the exponent code.

It has been proven that, in addition to improving the protection level, the proposed method provides a 7–15% acceleration of software modular exponentiation, which is important for low-power IoT terminal platforms.

***Keywords*:** modular exponentiation; public-key cryptography; power analysis; polymorphic computation; precomputation; IoT.

Стаття подана до редакції: 13/03/2026
Стаття прийнята до опублікування:23/03/2026
Стаття опублікована: 30/05/2026