Ссылка на статью:
// Ученые записки УлГУ. Сер. Математика и информационные технологии. УлГУ. Электрон. журн. 2017, № 1, с.48-53.

Поступила: 01.05.2017
Окончательный вариант: 05.08.2017

УДК 519.7

Методы ускорения пост-квантового криптографического протокола, основанного на задаче о нахождении гамильтонова цикла в графе

Ростов М. А.[1,*]

[*] cj3687@yandex.ru
[1]УлГУ, Ульяновск, Россия

В работе описана высокоскоростная программная реализация криптографического протокола с нулевым разглашением, основанного на задаче поиска гамильтонова цикла в графе с использованием технологии параллельных вычислений CUDA. Представлены методы работы и использования криптографических алгоритмов, а также алгоритм работы протокола с нулевым разглашением. Представлены результаты проведенных исследований данного протокола, пример работы с протоколом нулевого разглашения, сравнение результатов выполнения криптографического протокола на таких языках программирования, как Java, C, C# и PHP. Предлагаемое решение в виде внедрения технологии параллельных вычислений CUDA направлено на повышение скорости вычисления и сокращение объема используемой памяти при вычислении на GPU.

Ключевые слова: *CUDA, параллельные вычисления, высокоскоростные вычисления, задача о гамильтоновом цикле, криптографические протоколы с нулевым разглашением, пост-квантовая криптография.*

Введение

В настоящее время проблема задействования криптографических методов защиты данных стала очень актуальна. Ввиду увеличения использования вычислительных сетей, в том числе и такой широко используемой сети Internet. По ним пересылаются значительные объемы данных различного характера (в том числе коммерческого, государственного, частного и военного), которые не допускают возможности пользования ей посторонними лицами [1].

Базисным объектом исследования в теории протоколов с использованием криптографических шифров, являются абоненты, находящиеся удаленно друг от друга, но ведущие активное взаимодействие, зачастую, по открытым каналам связи. Решение какой-то конкретной задачи является основной целью взаимодействия таких абонентов. Довольно часто в таких случаях имеется некоторый противник со своими собственными целями. Основным предметом изучения в данной работе является задача о нахождении гамильтонова цикла в графе в криптографическом протоколе с нулевым разглашением. Решение данной задачи относят к классу NP, что позволяет назвать исследуемый протокол постквантовым [3].

Технология CUDA, применяемая к данному протоколу, уже стала довольно распространенным явлением. Ярким примером является проект компании TechniScan по обработке медицинских изображений аппаратов УЗИ Svara. Вычисления на GPU, как на мощнейшем процессоре, идеально подходящем для тяжелых вычислительных приложений, стали внедрять во множество проектов, связанных с обработкой сейсмических данных в индустрии нефте- и газодобычи, для вычислений в сфере бионаук, а также в финансовом моделировании [5].

1. Гамильтонов цикл в криптографических протоколах с нулевым разглашением знания

Рассмотрим пример, придуманный и описанный Мануэлем Блюмом в одной из его работ в 1986 году. В этом протоколе абонент A будет доказывать абоненту B, что он знает гамильтонов цикл в некотором графе G так, что абонент B не получит никаких знаний о самом этом цикле (доказательство с нулевым разглашением). Стоит напомнить, что «нулевое разглашение» означает, что независимо от числа реализаций протокола доказательства абонент B будет располагать точно такими же сведениями о гамильтоновом цикле, какие он бы мог получить, просто изучая представленный ему граф G.

Пусть абонент A знает гамильтонов цикл в графе G из N вершин. Он может это доказывать абоненту B (и всем, кто имеет этот граф) с помощью протокола.

Протокол доказательства состоит из следующих шагов [1]:

1. Абонент A случайно выбирает перестановку $\sigma \in S_n$ и применяет ее к номерам вершин графа G, получив при этом $H = \sigma(G)$. Понятно, что графы G и H изоморфны. Зная гамильтонов цикл графа G, абонент A знает гамильтонов цикл и в графе H. Граф H передается проверяющему B.
2. Абонент B, получив граф H, случайным образом выбирает $a \in \{0,1\}$ и передает a абоненту A.
3. Если $a = 0$, то абонент A предоставляет абоненту B перестановку σ (тем самым доказывая, что он знает изоморфизм графов G и H). Если $a = 1$, то абонент A предоставляет проверяющему гамильтонов цикл графа H.

4. Проверяющий B проверяет, что в случае $a = 0$ предъявленная перестановка σ действительно переводит граф G в граф H, а в случае $a = 1$ проверяет гамильтонов цикл графа H.

Весь протокол повторяется m раз.

Абонент B задает два вопроса абоненту A по следующей причине. Если бы он задавал только первый вопрос, то абонент A, не зная в действительности гамильтонова цикла в графе G, мог бы предоставить абоненту B совсем другой граф с таким же количеством вершин и искусственно заложенным в него гамильтоновым циклом. Поэтому иногда абонент B просит абонента A доказать изоморфизм графов G и H. При этом важно, что абонент A заранее не знает, какой из двух вопросов задает абонент B [2].

2. Методы повышения производительности исследуемого протокола

В ходе реализации криптографического протокола с нулевым разглашением, основанного на задаче поиска гамильтонова цикла в графе на различных языках программирования, были проведены скоростные тесты, основанные на изменении размера количества вершин графа (N). Как видно из диаграммы (Рис. 1), где различными цветами представлены графики зависимости времени (в миллисекундах) от количества вершин графа для различных языков программирования. Тесты проводились на ПК со следующими характеристиками: ОС Windows 10, GPU GeForce GTX 550 TI 2 Gb, ОЗУ 8 Gb, CPU Intel Core i5 1,7 GHz.

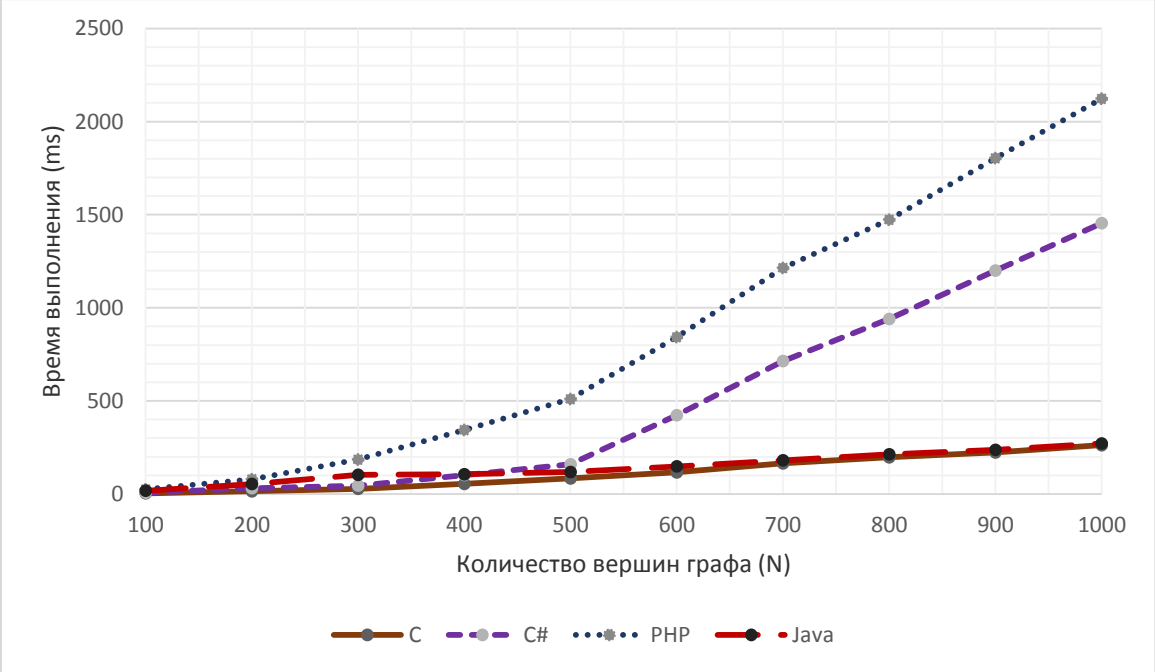

Рис. 1. Графики скоростных тестов выполнения протокола на языках Java, C, C#, PHP

Для рассмотрения скоростных тестов на начальном этапе обратимся к Рис. 2. Язык C, ввиду того, что является низкоуровневым, а также благодаря оптимизации при работе с разделенной памятью, показывает очень хороший результат, в сравнении с остальными языками. Стоит отметить, что язык C#, в частности сборщик мусора .NET платформы эффективно очищает память на начальных этапах, но при достижении показателя N = 500, скорость выполнения криптографического алгоритма заметно увеличивается. Работа сборщика мусора, встроенного в язык Java, выполняется при любом значении количества вершин графа, что позволяет производить расчёты с довольно высокой скоростью.

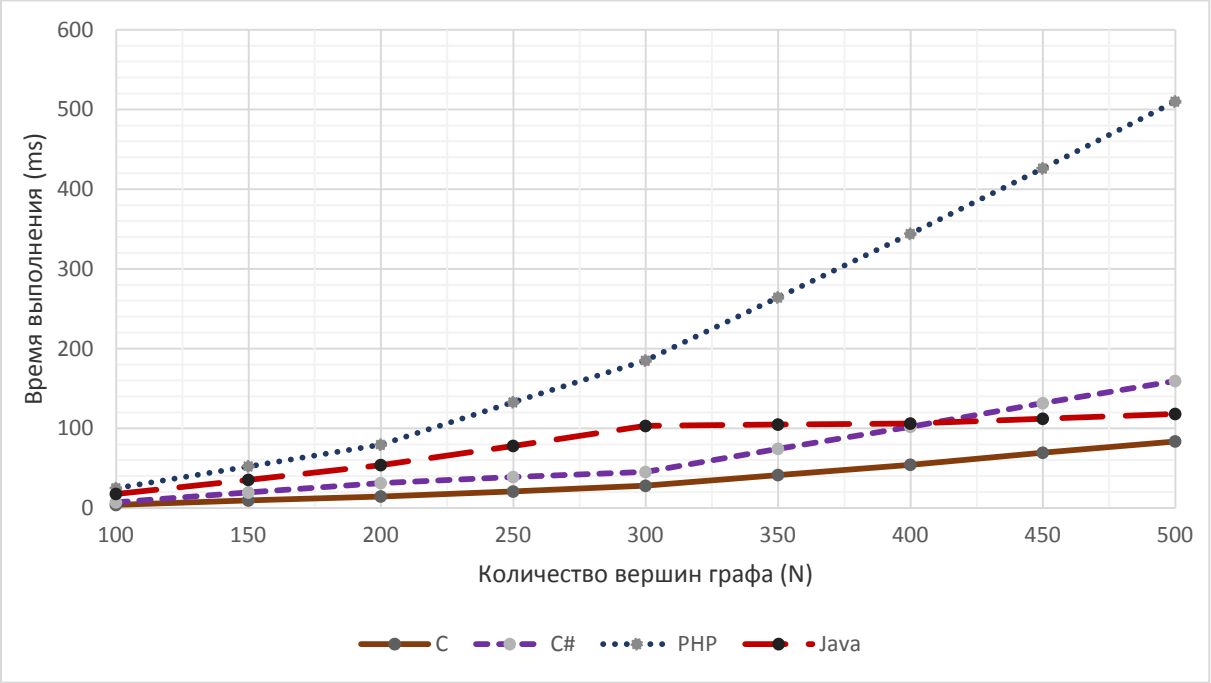

Рис. 2. Графики скоростных тестов на языках Java, C, C#, PHP

После внедрения в реализацию программного решения задачи о нахождении гамильтонова цикла в криптографическом протоколе с нулевым разглашением высокоскоростной технологии параллельных вычислений CUDA, основанном на языке C, провели скоростные тесты, представленные на Рис. 3. На диаграмме видно, с каким большим отрывом показана технология CUDA.

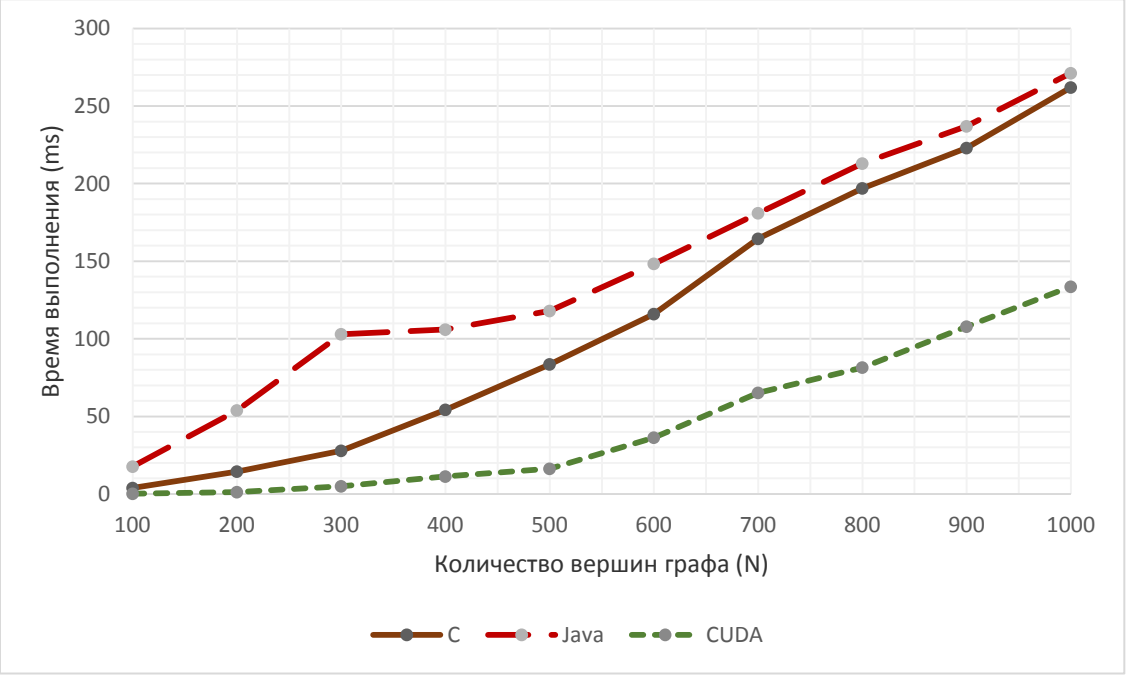

Рис. 3. Графики скоростных тестов выполнения протокола на языках Java, C, CUDA

Для рассмотрения скоростных тестов на начальном этапе обратимся к Рис. 4. Благодаря распараллеливанию процессов вычисления на GPU, технология CUDA имеет высочайшую скорость выполнения, в сравнении со всеми представленными языками программирования [4]. В ходе проведенных тестов было вычислено, что технология CUDA позволяет, в среднем, работает в 50 раз быстрее, чем самый медленный язык из представленных (PHP) и в 13 раз быстрее, чем язык C.

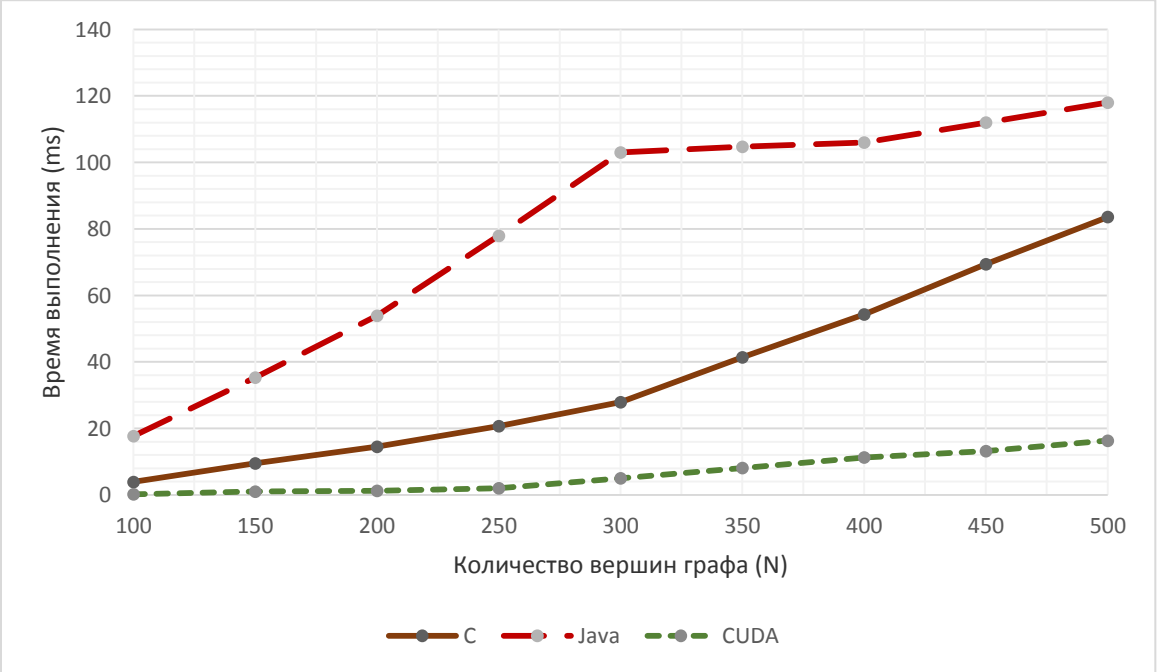

Рис. 4. Графики скоростных тестов на языках Java, C, CUDA

Заключение

Разработанная высокоскоростная программная реализация криптографического протокола с нулевым разглашением позволяет оптимизировать производительность данного алгоритма, а также сократить затраты на используемую память для вычислений. Теоретическое обоснование криптостойкости протокола с нулевым разглашением подтверждает безопасность использования исследуемого криптографического алгоритма в сетях Internet и Intranet. Внедрение высокопроизводительной технологии параллельных вычислений CUDA ускоряет процесс работы криптопротокола до 50 раз. В рамках постквантовой криптографии это говорит об отличных результатах, с учетом того, что сама технология базируется на языке С, то есть не требует огромного количества трудозатрат на ее изучение на начальном этапе.

Список литературы

1. Рябко Б. Я. *Криптографические методы защиты информации:* Учебное пособие для вузов / А. Н. Фионов, Б. Я. Рябко — Москва: Горячая линия — Телеком, 2005 — 229 с.
2. Шнайер Б. *Прикладная криптография. Протоколы, алгоритмы, исходные тексты на языке Си* / Б. Шнайер — Москва: Триумф, 2002 — 816 с.
3. Bernstein D. *Post-Quantum Cryptography* / Bernstein Daniel J., Buchmann Johannes, Dahmen Erik, 2008 — 245 с.
4. Боресков А. В. *Основы работы с технологией CUDA* / А. В Боресков, А. А. Харламов — Москва : ДМК Пресс, 2010 — 232 с.
5. Sanders J. *CUDA by Example: An Introduction to General-Purpose GPU Programming* / J. Sanders, E. Kandrot — Illinois: Addison-Wesley Professional, 2010 — 312 с.