УДК 681.3

ЦИФРОВАЯ ПОДПИСЬ ВЕБ-СЕРВИСА ПРОТОКОЛА TLS НА ОСНОВЕ ЭЛЛИПТИЧЕСКИХ КРИВЫХ И ОБЛАЧНОЙ ПЛАТФОРМЫ HEROKU

Н.С. ФИЛИППОВ, С.Б. САЛОМАТИН

Белорусский государственный университет информатики и радиоэлектроники, Республика Беларусь

Поступила в редакцию 02 ноября 2019

Аннотация. Рассмотрены криптографические алгоритмы цифровой подписи на эллиптических кривых в контексте «Интернет вещей», а также расширение возможностей стандартного протокола защиты транспортного уровня TLS на основе устройства *raspberry pi* и облачного протокола *Heroku.*

Ключевые слова: криптографическая защита информации. эллиптическая кривая, протокол TLS, интернет вещей, веб-сервис.

Введение

Сети «Интернет вещей» (IoT) используют симметричные и ассиметричные криптографические алгоритмы для защиты информации [1–3]. В качестве сенсоров часто используются устройства типа *raspberry pi*, имеющие малое потребление энергии, ограниченную пропускную способность и вычислительную мощность.

Криптографические алгоритмы на эллиптических кривых протокола TLS реализуются достаточно сложным образом, что приводит к значительным накладным расходам по отношению к времени выполнения, энергопотреблению и затрудняет их использование в устройствах *raspberry pi,* большая часть ресурсов которых занята прикладным программным обеспечением, оставляя очень ограниченное пространство для протоколов безопасности.

Одним из путей решения задачи по расширению возможностей протокола TLS на эллиптических кривых в контексте сетей IoT – *raspberry pi* связан с привлечением облачных технологий и, в частности, платформы *Heroku.*

Веб-сервис цифровой подписи TLS IoT на базе алгоритма ECDSA

В работе рассматривается реализация веб-сервиса цифровой подписи на базе алгоритма *ECDSA* (*Elliptic Curve Digital Signature Algorithm*) с использованием языка программирования *Python* и облачной платформы *Heroku*.

В качестве устройства сети «Интернет вещей» рассматривается *raspberry pi zero w* (рис. 1) со следующими техническими характеристиками [1]:

– однокристальная система *SoC Broadcom BCM2835*;
– процессор 32-битный 1-ядерный *ARMv6Z ARM1176JZF-S* с тактовой частотой 1 ГГц;
– графический 2-ядерный сопроцессор *Video Core IV Multimedia*;
– ОЗУ 512 Мб *LPDDR2 SDRAM*.
– WIFI 802.11n + Bluetooth 4.1 Low Energy (BLE).

Рис. 1. *Raspberry pi zero w*

В общем случае «Интернет вещей» состоит из беспроводных сенсорных узлов, которые собирают информацию и отправляют ее на ближайшую базовую станцию, называемую «шлюзом», которая, в свою очередь, связывается с общим или частным облаком [2].

Криптосистемы с открытым ключом работают медленнее, чем симметричные криптосистемы. Одна из наиболее распространенных асимметричных криптосистем – *RSA*.

Известно, что у данных алгоритмов имеются ряд недостатков, такие как большие размеры ключей, а также большие затраты времени для генерации ключей и цифровой подписи. Для использования криптографии в устройствах, память и вычислительная мощность которых сильно ограничены, необходимы более эффективные алгоритмы [3]. В качестве такого алгоритма в работе используется *ECDSA*.

Алгоритм криптографической защиты *ECDSA* TLS на основе эллиптических кривых и устройства *raspberry pi*

Эллиптическая кривая – это множество точек, описываемое уравнением

$$y^2 = x^3 + ax + b \bmod p.$$

Данные кривые обладают следующими характеристиками:

1. Простое p, задающее размер конечного поля;
2. Коэффициенты a и b уравнения эллиптической кривой;
3. Базовая точка G, генерирующая подгруппу.
4. Порядок n подгруппы.
5. Кофактор h подгруппы.

Примером эллиптической кривой, используемая в современных протоколах защиты информации, таких как TLS является кривая secp256k1 [4], представленная на рис. 2 и имеющая следующие параметры:

1. p=0xfffffffffffffffffffffffffffffffffffffffffffffffffffffffefffffc2f;
2. a=0;
3. b=7;
4. g=(0x79be667ef9dcbbac55a06295ce870b07029bfcdb2dce28d959f2815b16f81798, 0x483ada7726a3c4655da4fbfc0e1108a8fd17b448a68554199c47d08ffb10d4b8);
5. n=0xfffffffffffffffffffffffffffffffebaaedce6af48a03bbfd25e8cd0364141;
6. h=1.

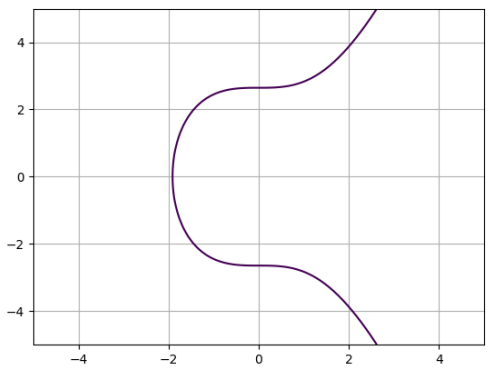

Рис. 2. График эллиптической кривой secp256k1

В качестве эксперимента выберем несколько эллиптических кривых, предоставляемых *OpenSSL*. *OpenSSL* – криптографическая библиотека с открытым исходным кодом, широко известна из-за расширения *SSL*/*TLS*, используемого в веб-протоколе *HTTPS*. Алгоритм цифровой подписи [5] был реализован на языке *Python*. Общая функция генерации цифровой подписи представлена на рис. 3.

```
   def sign_message(private_key, message):
z = hash_message(message)
r = 0
s = 0
while not r or not s:
    k = random.randrange(1, curve.n)
    x, y = scalar_mult(k, curve.g)
    r = x % curve.n
    s = ((z + r * private_key) * inverse_mod(k, curve.n)) % curve.n
return (r, s)
```

Рис. 3. Функция генерации цифровой подписи

Полученные результаты времени, необходимого *raspberry pi* для генерации цифровой подписи, представлены в табл. 1.

Таблица 1. **Результаты измерений времени генерации цифровой подписи на *raspberry pi***

Эллиптическая кривая	Время генерации подписи, с
Secp192k1	0,78
Secp192r1	0,73
Secp224r1	0,92
Secp256k1	1,20
Secp256r1	1,32
Secp384r1	2,91

Алгоритм криптографической защиты на основе эллиптических кривых, устройства *raspberry pi* и платформы *Heroku*

Для увеличения производительности данный алгоритм перенесен на облачную платформу *Heroku* [6] в виде веб-приложения.

Heroku – облачная *PaaS (Platform as a Service)* платформа, поддерживающая ряд языков программирования.

PaaS – модель предоставления облачных вычислений, при которой потребитель получает доступ к использованию информационно-технологических платформ: операционных систем, систем управления базами данных, связующему программному обеспечению, средствам разработки и тестирования, размещенным у облачного провайдера.

Для работы приложения были созданы 2 таблицы в базе данных платформы *Heroku*:

– для хранения информации о каждом устройстве в сети, которому необходим доступ к сервису;

– для хранения используемых эллиптических кривых и их параметров.

Схема базы данных приложения представлена на рис. 4.

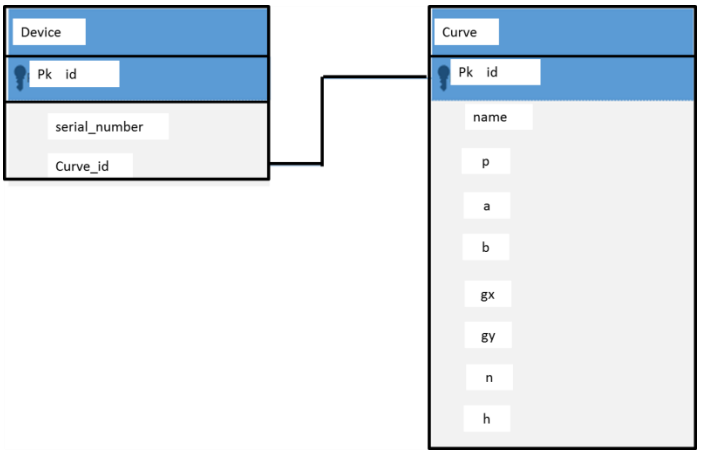

Рис. 4. Схема сущностей в базе данных

Результаты эксперимента для анализа затраченного времени для генерации цифровой на платформе *Heroku* приведены в табл. 2.

Таблица 2. **Результаты измерений времени генерации цифровой подписи с использованием веб-сервиса**

Эллиптическая кривая	Время генерации подписи, с
secp192k1	0,16
secp192r1	0,15
secp224r1	0,34
secp256k1	0,56
secp256r1	0,58
secp384r1	1,05

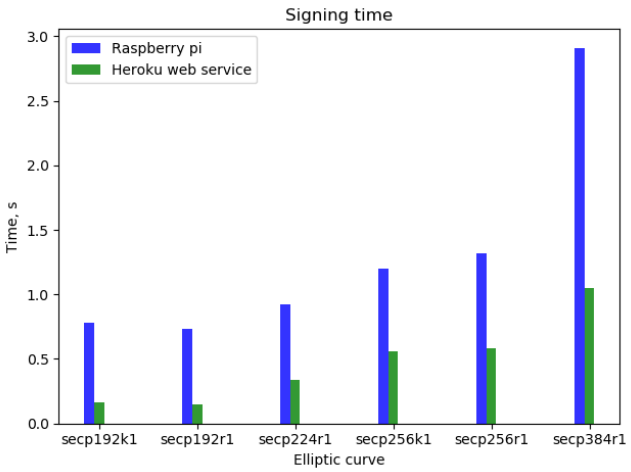

Рис. 5. Сравнение времени генерации цифровой подписи на *raspberry pi* и веб-сервисе

Заключение

Веб-приложение на базе платформы *Heroku*, позволило освободить часть ресурсов *raspberry pi* за счет переноса алгоритма генерации цифровой подписи на облачный веб-сервис, тем самым увеличив производительность устройства. Благодаря встроенной базе данных, данное приложение позволяет гибко изменять используемую эллиптическую кривую и ее параметры для каждого устройства в сети, что не позволяет стандартная реализация протокола TLS [7].

DIGITAL SIGNATURE OF THE TLS PROTOCOL WEB SERVICE BASED ON THE ELLIPTIC CURVES AND HEROKU CLOUD PLATFORM

N.S. FILIPPOV, S.B. SALOMATIN

Abstract. The cryptographic algorithms of digital signature on elliptic curves in the context of the Internet of Things were considered, as well as the expansion of the capabilities of the standard TLS transport layer protection protocol based on the *raspberry pi* device and *Heroku* cloud protocol.

Keywords: cryptographic information protection, elliptic curve, TLS protocol, Internet of things, web service IoT.

Список литературы

1. Why Is IoT? [Electronic resource]. URL: https://www.oracle.com/internet-of-things/what-is-iot.html (date of access: 02.11.2019).
2. Raspberry Pi Zero W [Electronic resource]. URL: https://www.raspberrypi.org/products/raspberry-pi-zero-w/ (date of access: 02.11.2019).
3. Nils Gura, [et. al.] // Workshop on Cryptographic Hardware and Embedded Systems, LNCS 3156. 2004. P. 119–132.
4. SEC 2: Recommended Elliptic Curve Domain Parameters. Daniel R. L. Brown. – NIST, 2010.
5. Johnson D., Menezes A. // International Journal of Information Security. 2001. Vol. 1. P. 36–63.
6. Learn about building, deploying, and managing your apps on Heroku [Electronic resource]. URL: https://devcenter.heroku.com/ (date of access: 02.11.2019).
7. Ключи, шифры, сообщения: как работает TLS [Электронный ресурс]. URL: https://tls.dxdt.ru/tls.html#crypto-params (дата доступа: 02.11.2019).