ПОДХОД К ОПРЕДЕЛЕНИЮ СОСТОЯНИЯ СЕРТИФИКАТА ЧЕРЕЗ СЕТЬ С ИСПОЛЬЗОВАНИЕМ СМАРТ-КОНТРАКТОВ

Маркевич Н.А.[1], Лившиц И.И.[2]

Целью *исследования является повышение доступности службы OCSP.*

Метод *исследования: в работе использованы методы математической логики, методы сканирования хостов, методы построения распределённых систем.*

Результат. *В работе проведено исследование доступности трёхсот трёх OCSP-ответчиков, которые хранят состояния SSL-сертификатов более полумиллиона самых популярных веб-сайтов. Полученные данные позволили выявить проблему, заключающуюся в том, что OCSP-ответчики не всегда в состоянии обеспечивать требуемой от них 100% доступности. В качестве противопоставления централизованной системе в работе была исследована доступность узлов блокчейн-платформы Ethereum. Показано, что доступность децентрализованной системы выше по причине большего количества и распространенности узлов.*

Предложен подход к построению децентрализованной службы проверки состояния сертификата – представлена модель децентрализованного OCSP-ответчика и на ее основе описаны функции смарт-контракта и порядок взаимодействия участников системы. Предложенный подход позволяет увеличить доступность службы OCSP на 30% и распределить нагрузку между OSCP-ответчиками. Решение может быть применено для построения высокодоступных систем открытых ключей масштабов Интернета, а также для корпоративных систем управления ключами.

Ключевые слова: *информационная безопасность, Интернет, удостоверяющий центр, OCSP, HTTPS, SSL, Ethereum, блокчейн, увеличение доступности.*

Список сокращений:
ИБ – информационная безопасность
ИТ – информационные технологии
ПО – программное обеспечение
УЦ – удостоверяющий центр
CRL – Certificate Revocation List
OCSP – Online Certificate Status Protocol

DOI:10.21681/2311-3456-2020-2-32-41

Введение

Сегодня по сети Интернет циркулирует огромное количество конфиденциальной информации разного рода – это может быть коммерческая тайна, платежные данные пользователей онлайн-магазинов, персональные данные граждан, обращающихся за госуслугами. Для обеспечения безопасного взаимодействия используется расширение протокола HTTP – HTTPS. HTTPS позволяет аутентифицировать веб-сервер и зашифровать канал связи.

Взаимодействие между веб-сервером и клиентом строится с использованием сертификатов открытого ключа, или, по-другому, SSL-сертификатов. Веб-сервер обращается за получением SSL-сертификата в УЦ, доверие к которому устанавливают разработчики браузеров и операционных систем [1].

УЦ ответственны за поддержку информации о статусе сертификата и доведении её до пользователей. Это обеспечивается двумя механизмами. Первый – CRL (списки отозванных сертификатов) [2]. В этом случае УЦ выпускает раз в день, неделю или месяц (в зависимости от регламента) список сертификатов, которые были отозваны. Такой метод является устаревшим – списки могут достигать больших размеров – в исследовании Yabing Liu и др. были обнаружены списки размером в 76 Мб [3], а их обновление пользователю необходимо проводить вручную. Второй механизм – OCSP (протокол определения состояния сертификата через сеть) [4]. Он позволяет установить состояние сертификата в режиме реального времени прямо перед установлением защищенного соединения. В этом случае клиент обращается к OCSP-ответчику, адрес которого указан в сертификате веб-сервера и опрашивает его о статусе сертификата – валиден ли он или отозван.

Начиная с января 2013 года, согласно требованиям CA/Browser Forum Baseline Requirements, УЦ должны поддерживать OCSP службы, при этом они должны предоставлять ответ клиентам в течение 10 секунд 24 часа/7 дней в неделю. Таким образом, OCSP-ответчики

1 Маркевич Никита Алексеевич, аспирант, Университет ИТМО, г. Санкт-Петербург, Россия. E-mail: markevich.nikita1@gmail.com

2 Лившиц Илья Иосифович, доктор технических наук, доцент, Университет ИТМО,г. Санкт-Петербург, Россия. E-mail: livshitz.il@yandex.ru

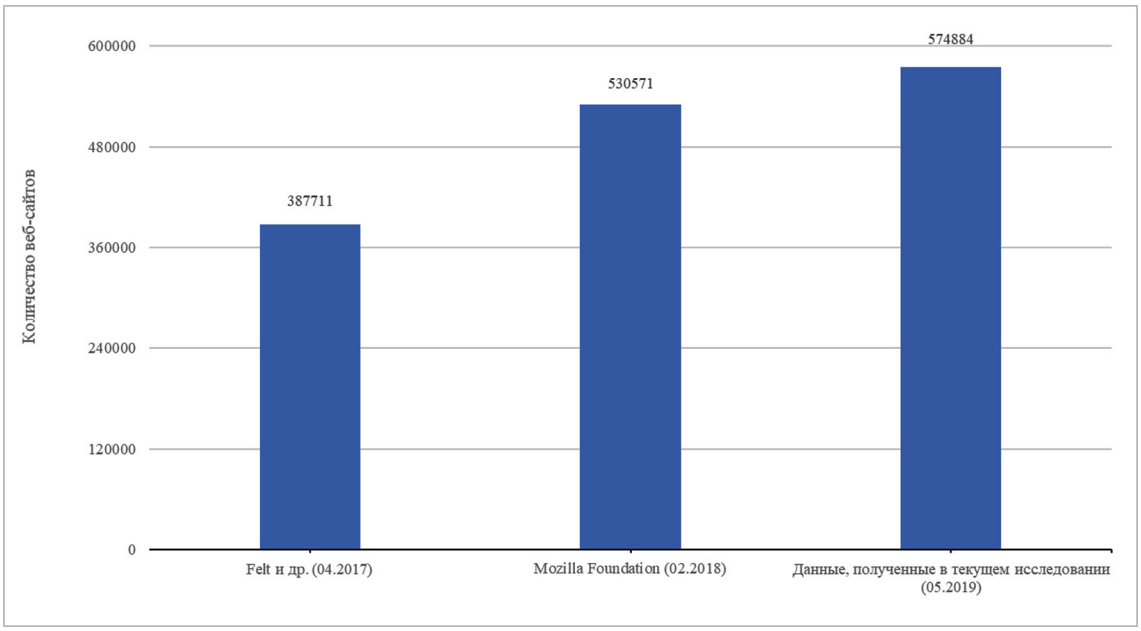

Рисунок 1. Сравнение полученных данных поддержки веб-сайтами HTTPS

представляют собой высокодоступные публичные сервера. В работе Serrano и др. [5] было проанализировано более 1300 инцидентов, в которые были вовлечены УЦ. В 39 случаях инциденты были связаны с несоответствием проблемами в механизмах проверки состояния сертификата или с несоответствием этих механизмов требованиям CA/Browser Forum.

При этом, сама по себе доступность OCSP-ответчиков еще не была в полной мере изучена ни российскими, ни зарубежными исследователями. На это есть причины. Пользователи обычно не задумываются, а исследователи мало обращают внимания на то, как происходит проверка статуса сертификата веб-сервера. В случае, если OCSP-ответчик не ответил клиенту за приемлемое время, браузер или иное прикладное ПО принимает статус сертификата как валидный. Такой режим работы носит название soft-fail. При этом, переключение в режим hard-fail, при котором в случае невозможности проверки статуса сертификата посредством OCSP-служб происходит его отклонение, приводит к недоступности веб-сервисов [6].

Для повышения безопасности инфраструктуры открытых ключей Интернета вводятся все новые механизмы защиты. Так, подход Certification Transparency [7] предлагает использование публичных лог-серверов для записи всех выданных сертификатов. Такой способ позволяет обнаружить злоупотребления со стороны УЦ или его компрометацию, однако они не предназначены для снижения нагрузки на OCSP-ответчики. Лог-серверы разворачиваются на текущих мощностях УЦ и их количество невелико – 41 сервер[3].

В последнее время находят свое место решения на базе распределенных децентрализованных систем. Исследования, проведенные зарубежными учеными, направлены на полный отказ от текущей парадигмы инфраструктуры открытых ключей и замену УЦ на полностью децентрализованные механизмы, в которых пользователь сам генерирует себе сертификат и публикует его в базе данных – блокчейне [8, 9]. В работе Yao S. и др. [9] предложеный механизм сохранения состояния сертификатов в блокчейне предполагает использования OCSP-серверов УЦ, которые, как будет показано ниже, не всегда могут предоставить информацию о статусе сертификата конечному пользователю.

Проделанная работа

В работе было проведено исследование доступности OCSP-ответчиков на выборке 1 миллиона самых популярных веб-сайтов (по версии компании Alexa). Было отдано предпочтение рейтингу Alexa поскольку он содержит только веб-сайты в отличии от рейтинга Cisco Umbrella [10].

Первым этапом исследования доступности OCSP-ответчиков была проверка наличия HTTPS у опрашиваемых сайтов. Каждому сайту из списка посылался HEAD-запрос по протоколу HTTP и в случае доступности, следующий HEAD-запрос посылался уже по протоколу HTTPS. Удалось установить, что HTTPS поддерживает 574884 сайта. Прирост по сравнению с данными, полученными исследователями в 2017 году [11], составил 187173 веб-сайта и 44313 веб-сайт по сравнению со сканированием, проведенным организацией Mozilla Foundation в феврале 2018 года[4] (рис. 1).

На следующем этапе после установления HTTPS соединения из сертификата веб-сервера извлекалась

3 Согласно данным https://www.gstatic.com/ct/log_list/log_list.json

4 Analysis of the Alexa Top 1M Sites: https://blog.mozilla.org/security/2018/02/28/analysis-alexa-top-1m-sites-2/.

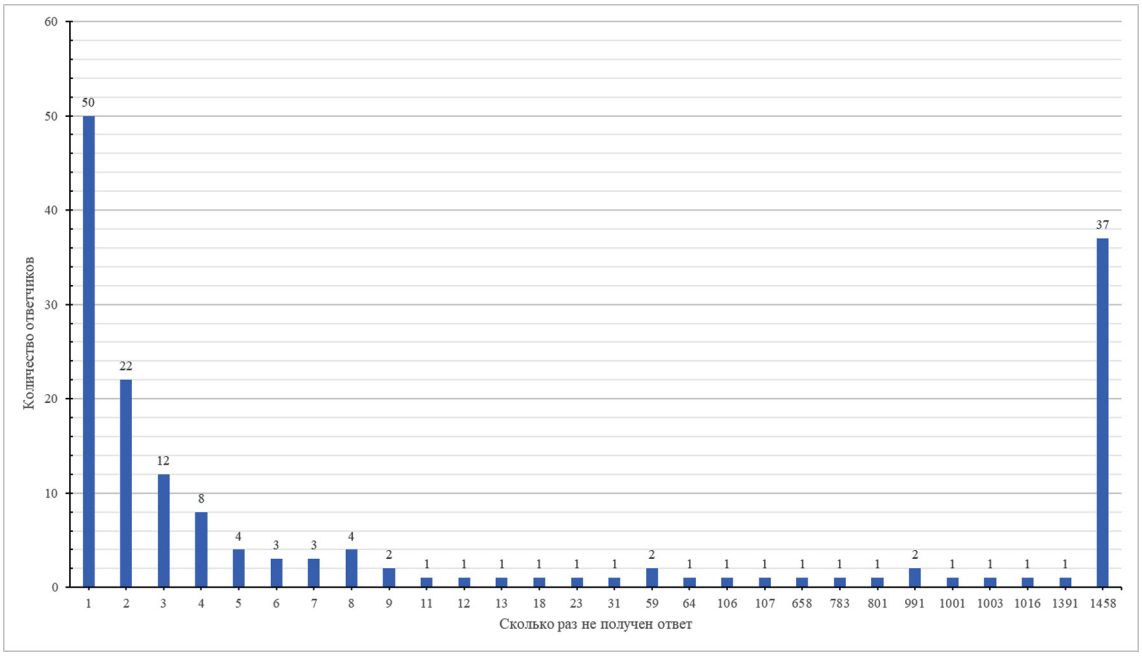

Рисунок 2. Соотношение между количеством OCSP-ответчиков и тем, сколько раз они были недоступны

OCSP URI запись. Из 574884 сертификатов было извлечено 303 уникальных URI. По каждому адресу посылался GET/HEAD-запрос с целью определения доступности ответчика. Всего было отправлено 1458 запросов к каждому сервису в течение 5 дней.

На рис. 2 полученные данные проиллюстрированы в виде гистограммы. По вертикальной оси указано количество ответчиков из общего числа, а по горизонтальной оси указано на сколько запросов из 1458 не был получен ответ. При этом 138 ответчиков ответили на все запросы к ним (не указано на гистограмме). 37 серверов были недоступны в течение всего промежутка времени сбора данных. Можно предположить, что эти серверы неправильно сконфигурированы, либо выведены из эксплуатации. Некоторые из сертификатов содержали OCSP URI, которые вели на внутренние сайты организаций (псевдодомен .local), а один из сертификатов содержал ссылку на исполняемый файл (http://gca.nat.gov.tw/cgi-bin/OCSP2/ocsp_server.exe).

Количество обслуживаемых OCSP-ответчиками вебсайтов отражает степенную линию тренда, что отображено на рис. 3. OCSP-ответчики, в том числе крупные, не могут обеспечить требования 100% доступности, испытывают отказы и перебои в работе, как согласно собранным во время исследования данным, так и по заявлениям самих УЦ[5] и пользователей.[6]

Также текущее распределение УЦ и OCSP-ответчиков происходит неравномерно по странам мира. Например, в России ни один из коммерческих и ведомственных (Центральный банк, Министерство обороны, Федеральная налоговая служба и т.д.) УЦ не входит в доверенные списки современных браузеров и операционных систем.

Было определено местоположение 197 ответчиков из 303. Наибольшее количество ответчиков приходится на США (58), на 2-м месте Нидерланды (42), на 3-м – Россия (12). При этом 8 ответчиков на территории России относятся к УЦ, доверие к которым было утрачено – WoSign и StartCom, ввиду многочисленных нарушений требований CAB Forum [12]. На рис. 4 представлена фоновая картограмма распределения OCSP-ответчиков по странам мира.

Как следует из полученных данных, OCSP-ответчики не всегда справляются с нагрузкой. Случается и такое, что IP-адреса блокируются органами государственной власти[7]. Также OCSP-ответчики неравномерно распределены по странам, что увеличивает время на установление защищенного соединения. Ниже предлагается подход к построению децентрализованной системы OCSP-ответчиков на платформе Ethereum, что позволит увеличить доступность службы OCSP.

Технология блокчейн и смарт-контрактов находит успешное применение в областях ИТ. Центральным банком РФ, при поддержке ведущих ИТ-компаний, проводится тестирование пилотных проектов в области распределенных реестров и осуществляются попытки нормативно-законодательного регулирования. [13]

5 OSCP servers down again. URL: https://community.letsencrypt.org/t/seem-that-oscp-servers-down-again/38950/3.
OCSP Partial Service Disruption. URL: https://letsencrypt.status.io/pages/incident/55957a99e800baa4470002da/5a6753733800d404cc4ea7db.
OCSP response delays. URL: https://letsencrypt.status.io/pages/incident/55957a99e800baa4470002da/5a3437ecdd1baf047386b2dd.

6 Let's Encrypt is down. URL: https://news.ycombinator.com/item?id=14374933
Comodo OCSP Outage. URL: https://forums.cpanel.net/threads/comodo-ocsp-outage.604051/.

7 IP-адрес удостоверяющего центра Digicert внесен в реестр запрещенных сайтов. URL: https://habr.com/ru/post/357196/.
Роскомнадзор заблокировал самого себя и некоторые сайты правительства (Comodo). URL: https://habr.com/ru/post/357152/

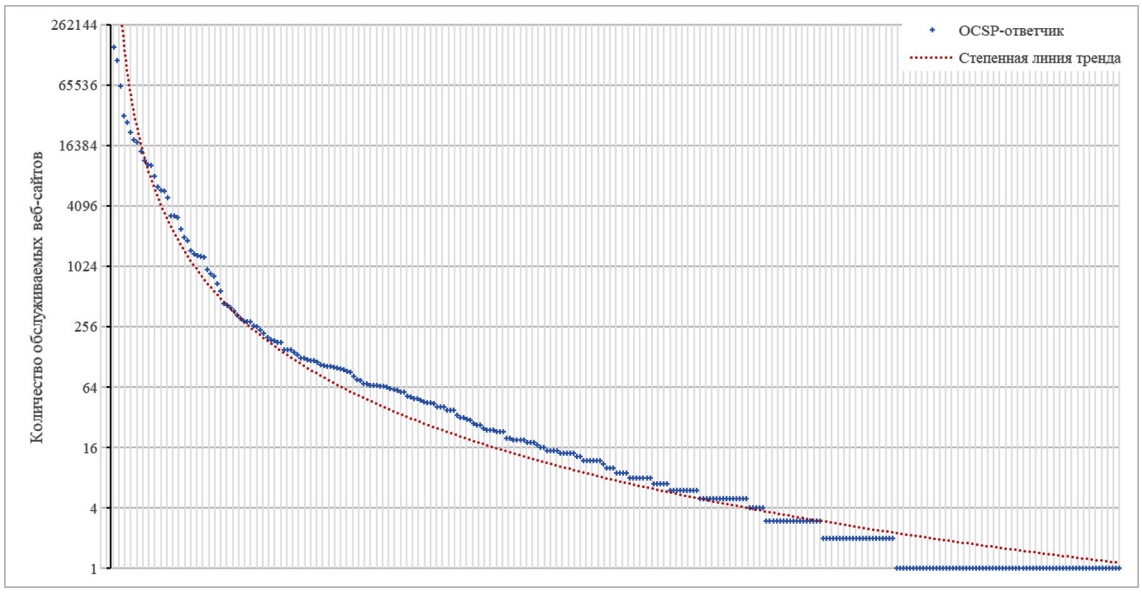

Рисунок 3. Распределение количества обслужиаемых OCSP-ответчиками веб-сайтов

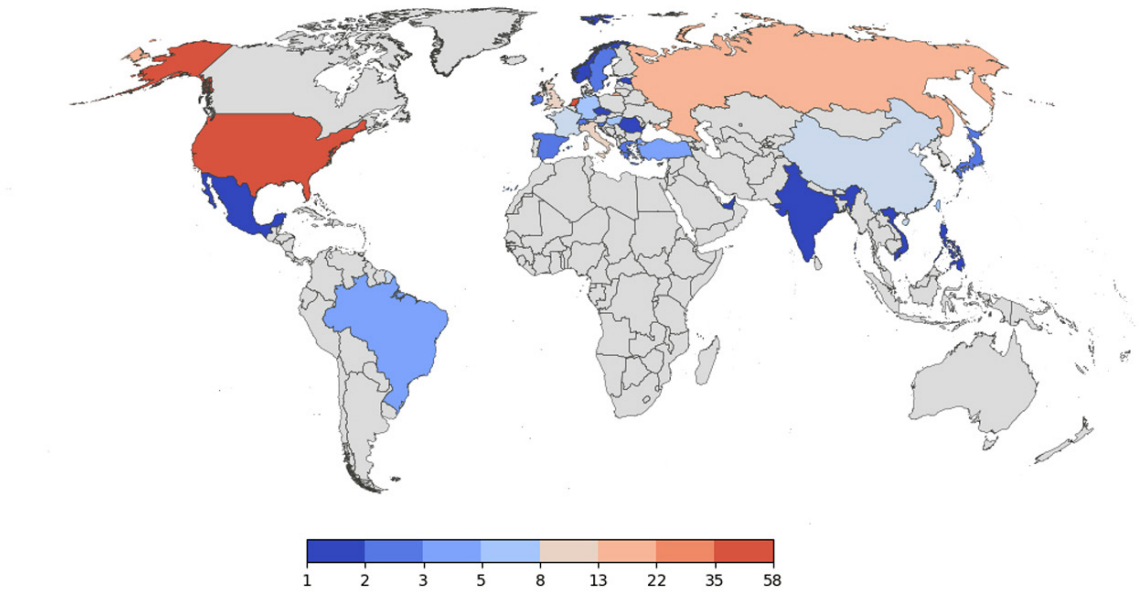

Рисунок 4. Фоновая картограмма распределения OCSP-ответчиков

Аналитическая компания Gartner отдельно выделила в 2018 году «блокчейн для защиты данных» как одну из новых технологий, которая только начинает находить свое применение[8].

Ethereum – это блокчейн-платформа для создания децентрализованных сервисов, работающих на базе смарт-контрактов. Смарт-контракты Ethereum выполняются на всех узлах, которые соединены в единую децентрализованную сеть. Контракты обладают полнотой по Тьюрингу, а значит с их помощью можно реализовать любую логику, в том числе и логику OCSP-ответчиков. Таким образом, каждый узел децентрализованной сети Ethereum мог бы выступать в качестве OCSP-сервера для каждого выпущенного сертификата.

Платформа Ethereum была выбрана также по причине высокой скорости добавления блока, отлаженным механизмам смарт-контрактов и высокой безопасностью сети в целом – разработка и запуск блокчейна «с нуля» приводит к угрозе атаки 51% [14], поскольку количество майнеров в начале жизни блокчейна невелико.

8 Gartner Hype Cycle for Emerging Technologies. URL: https://www.gartner.com/smarterwithgartner/5-trends-emerge-in-gartner-hype-cycle-for-emerging-technologies-2018/

Каждый узел в децентрализованной сети выступает в качестве самостоятельного OCSP-ответчика, который должен знать информацию о каждом сертификате. Таким образом, узел хранит информацию о множестве сертификатов удостоверяющих центров и их состояниях, а также о множестве сертификатов веб-серверов и их состояниях. При этом, сертификат идентифицируется по хешу публичного ключа владельца.

Модель децентрализованного OCSP-ответчика выглядит следующим образом:

$$PK \supset \left(pk_{CA}, pk_{CA}^{rev}, pk_{web}, pk_{web}^{rev}\right), \text{ где}$$

pk_{CA} – множество действительных сертификатов (хеши публичных ключей) УЦ;

pk_{CA}^{rev} – множество отозванных сертификатов УЦ;

pk_{web} – множество действительных сертификатов веб-серверов;

pk_{web}^{rev} – множество отозванных сертификатов веб-серверов.

OCSP-ответчик должен иметь возможность добавлять хешу публичных ключей в блокчейн:

$$f: k \notin PK \mapsto \left(k \in pk_{CA} \vee k \in pk_{web}\right)$$

OCSP-ответчик должен иметь возможность помечать сертификаты k как «отозванные»:

$$f: k \in pk_{CA} \mapsto k \in pk_{CA}^{rev}$$
$$f: k \in pk_{web} \mapsto k \in pk_{web}^{rev}$$

OCSP-ответчик должен возвращать статус s для сертификата k веб-сервера с вышестоящим УЦ l в ответ на запрос:

$$f: \left(k \in pk_{web} \wedge l \in pk_{CA}\right) \mapsto s := 1$$
$$f: (k \in pk_{web}^{rev} \vee l \in pk_{CA}^{rev}) \mapsto s := 0$$
$$f: (k \notin PK \wedge l \notin PK) \mapsto s := 0$$

Определим функции смарт-контракта, опираясь на описанную выше модель. Исходный код, представленный ниже, написан на языке описания смарт-контрактов Solidity.

Определим данные, которые будут храниться в блокчейне:

```
1. mapping (string => bool) caCertStatus;
```

Тип данных `mapping` представляет собой хеш-таблицу. `CaCertStatus` будет хранить в блокчейне состояние сертификата УЦ в виде булева значения. `True` – сертификат действителен, `false` – отозван (например, если закрытый ключ УЦ скомпрометирован). Так, `caCertStatus` может принимать вид:

```
1. caCertStatus {
2.   "cc2bd8f09bb88b5dd20f9b432631b8ca":
      True,
3.   "82cc921c6a5c6707e1d6e6862ba3201a" :
      False
4.     ...
5. }
```

Для хранения состояния сертификата веб-сервера также будет использоваться тип данных `mapping`.

```
1. mapping (string => bool) certStatus;
```

В дополнение к этому необходимо связывать сертификат веб-сервера с соответствующем ему сертификатом УЦ.

```
1. struct statusStruct {
2.   mapping (string => bool)certStatus;
3. }
4. mapping (string => statusStruct)
   caCertStorage;
```

Определим модификатор `onlyOwner`, который позволит ограничить использование функций смарт-контракта только его создателю.

```
1. modifier onlyOwner() {
2.   require(msg.sender == owner);
3.   _;
4. }
```

Определим функцию `createCA`, которая принимает на вход хеш открытого ключа издателя сертификата – УЦ и сохраняет его состояние как «валидный» в блокчейне.

```
1. function createCA(string memory _
   issuerKeyHash) public onlyOwner {
2.   caCertStatus[_issuerKeyHash] =
   true;
3. }
```

Определим функцию `revokeCACert`, которая принимает на вход хеш открытого ключа издателя сертификата и выставляет значение статуса для этого УЦ в `false`.

```
1. function revokeCACert(string memory
   _issuerKeyHash) public onlyOwner {
2.   caCertStatus[_issuerKeyHash] =
   false;
3. }
```

Определим функцию `storeCert`, которая принимает на вход хеш публичного ключа веб-сервера и хеш публичного ключа издателя и сохраняет состояние сертификата веб-сервера как «валидный» в блокчейне

```
1. function storeCert(string memory
   _certHash, string memory _
   issuerKeyHash) public onlyOwner {
2.   statusStruct storage s =
     caCertStorage[_issuerKeyHash];
3.   s.certStatus[_certHash] = true;
4. }
```

Определим функцию `revokeCert`, которая принимает на вход хеш публичного ключа веб-сервера и хеш публичного ключа издателя и помечает состояние сертификата как «невалидный» в блокчейне.

```
1. function revokeCert(string
   memory _certHash, string memory _
   issuerKeyHash) public onlyOwner {
2.   statusStruct storage s =
     caCertStorage[_issuerKeyHash];
3.   s.certStatus[_certHash] = false;
4. }
```

Определим функцию `checkCertStatus`, которая принимает на вход хеш публичного ключа веб-сервера и хеш публичного ключа издателя и возвращает состояние сертификата, при этом она не меняет состояние блокчейна, а значит является бесплатной для участников сети.

```
1. function checkCertStatus(string
   memory _certHash, string memory
   _issuerKeyHash) public view
   returns(bool) {
2.     require(caCertStatus[_
       issuerKeyHash] == true);
3.     return caCertStorage[_
       issuerKeyHash].certStatus[_
       certHash];
4. }
```

Дополнительная директива `require` позволяет убедиться в том, что сертификат УЦ действителен. Так, если сертификат УЦ был отозван, то проверка сертификата веб-сервера завершится с ошибкой.

УЦ должен выполнить последовательно следующие действия для создания OCSP-ответчика.

1. Создать смарт-контракт с указанным выше кодом и анонсировать его в сеть Ethereum.
2. Создать транзакцию с вызовом функции `createCA`, подав на вход хеш своего публичного ключа.
3. При подписании сертификата веб-сервера, добавлять адрес смарт-контракта в сертификат, например, в поле “Authority Information Access”, чтобы клиенты веб-сервера могли проверить статус сертификата, создав вызов `checkCertStatus`.

В общем случае порядок взаимодействия всех участников системы проиллюстрирован на рис. 5.

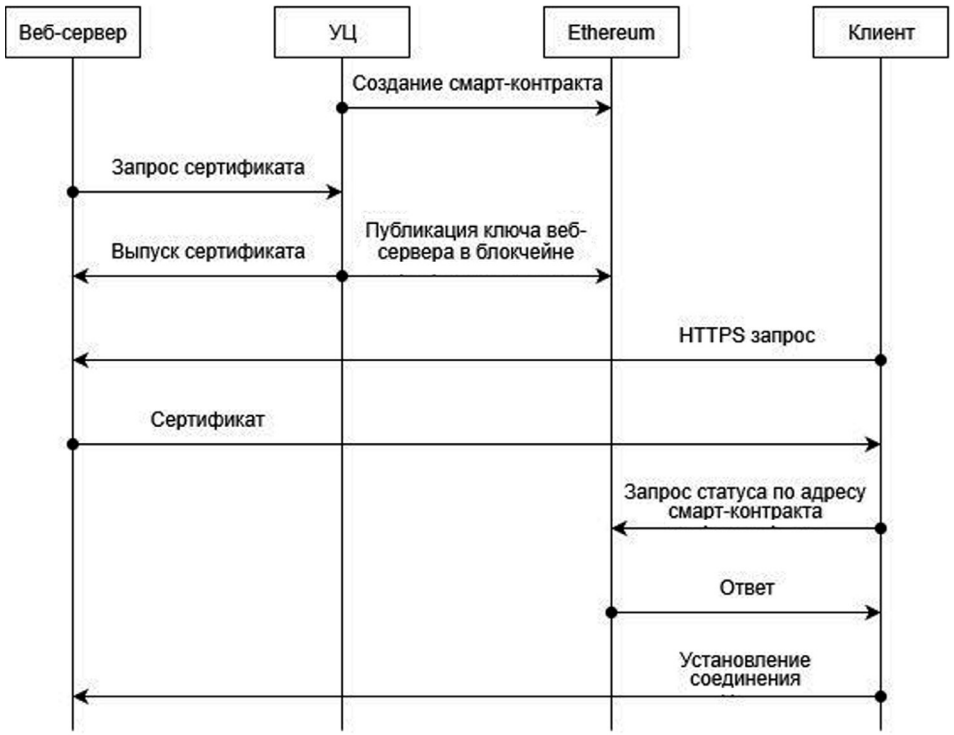

Рисунок 5. Порядок взаимодействие участников системы

DOI:10.21681/2311-3456-2020-2-32-41

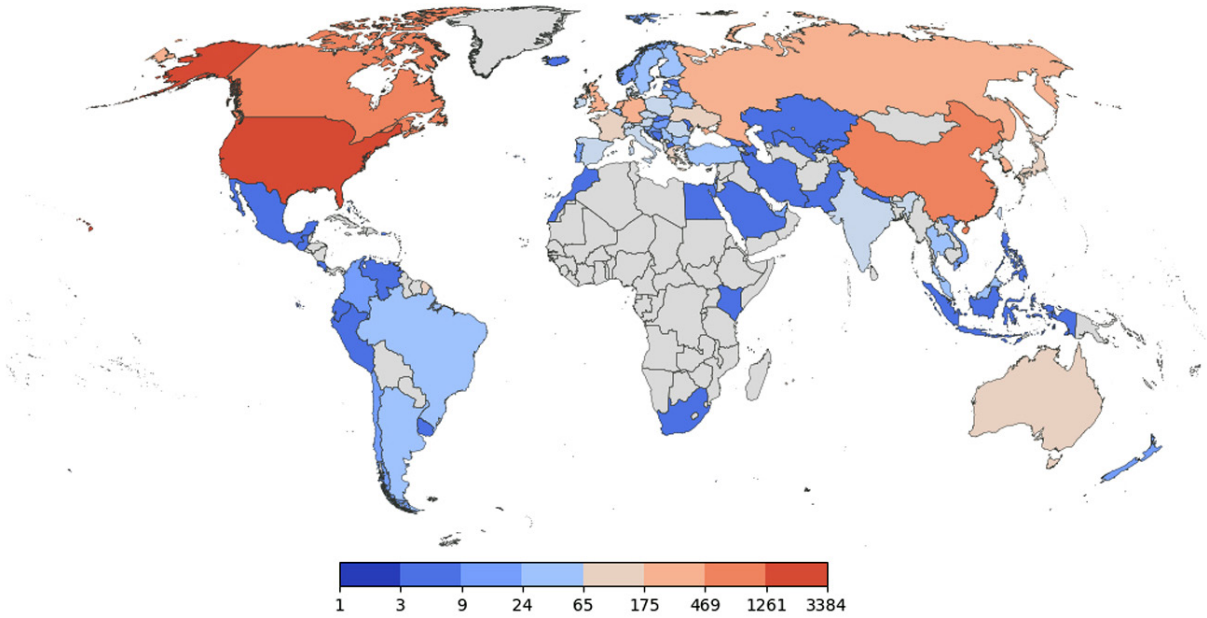

Рисунок 6. Фоновая картограмма распределения Ethereum-узлов

Число узлов в сети Ethereum находится в пределах от 4302 до 15454 [15]. На рис. 6 показано распределение Ethereum-узлов согласно данным, собранным 1 июня 2019 года. Максимальное количество узлов зафиксировано в США – 3384 узла. В России – 257. Как минимум 1 узел присутствует в 84 странах. Общее количество обнаруженных узлов – 7961.

Ввиду большего распространения Ethereum узлов, чем традиционных OCSP-ответчиков, помимо увеличения доступности службы OCSP, из-за увеличения количества узлов в 26 раз, уменьшится время, необходимое на проверку состояния сертификата. Так как каждый узел в сети Ethereum хранит копию всего блокчейна, обращаться для проверки статуса сертификата можно будет к ближайшему узлу.

Было измерено время ответа от OCSP-серверов, которые ответили на запросы в ходе исследования и от Ethereum-узлов, территориально расположенных близко к Санкт-Петербургу – в России, Украине, Беларуси, Финляндии, Норвегии, Эстонии, Германии и Польше.

На рис. 7 и рис. 8 проиллюстрировано сравнение минимальных и максимальных значений времени ответов, полученных на запросы к OCSP-серверам и Ethereum-узлам

На рис. 9 проиллюстрировано сравнение медиан времени ответов, полученных на запросы к OCSP серверам и узлам Ethereum.

Таким образом, внедрение OCSP-служб с использованием смарт-контрактов и блокчейн-хранилища позволит снизить в среднем на 14,25 мс (-30%) время ответа на запрос для пользователей из Европы. Также, создание инфраструктуры открытых ключей на платформе Ethereum позволило бы обеспечить практически постоянной работой майнеров, что позволит привлечь новых участников сети и положительно скажется на конкуренции между ними.

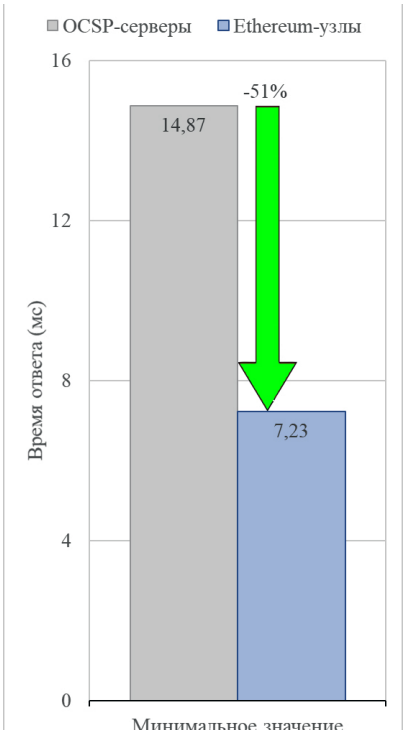

Рисунок 7. Сравнение минимального времени ответа OCSP-серверов и Ethereum-узлов

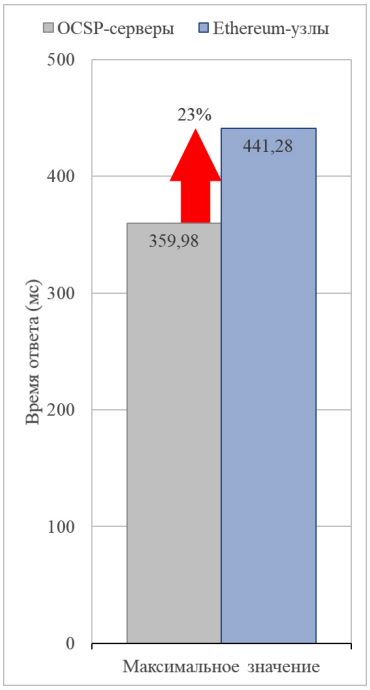

Рисунок 8. Сравнение максимального времени ответа OCSP-серверов и Ethereum-узлов

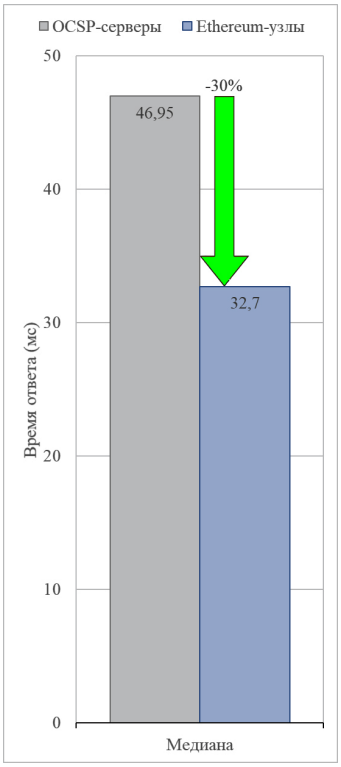

Рисунок 9. Медиана времени ответа OCSP-серверов и Ethereum-узлов

Выводы

Внедрение OCSP-ответчиков на платформе Ethereum с использованием смарт-контрактов позволило бы не только увеличить доступность за счет увеличения числа таких ответчиков в 26 раз и большей территориальной распространённости (84 страны с Ethereum-узлами против 28 стран с традиционными OCSP-ответчиками), что представлено на картограммах, но и распределить нагрузку между ними более равномерно.

Полученные в ходе исследования результаты позволяют подтвердить на практике возможность снижения задержки ответа на 30%.

Рецензент: *Молдовян Александр Андреевич, доктор технических наук, профессор, начальник научно-исследовательского отдела проблем информационной безопасности Федерального государственного бюджетного учреждения науки Санкт-Петербургского института информатики и автоматизации РАН (СПИИРАН), г. Санкт-Петербург, Россия. E-mail: maa1305@yandex.ru.*

Литература

1. Fadai T., Schrittwieser S., Kieseberg P., Mulazzani M. Trust me, I'm a Root CA! Analyzing SSL Root CAs in Modern Browsers and Operating Systems. In Proceedings – 10th International Conference on Availability, Reliability and Security // IEEE, 2015. pp. 174–179. DOI: 10.1109/ARES.2015.93
2. Cooper D., Santesson S., Farrell S., Boeyen S., Housley R., Polk W. Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile. 2008. DOI: 10.17487/RFC5280
3. Liu Y., Tome W., Zhang L., Choffnes D., Levin D., Maggs B., Mislove A., Schulman A., Wilson C. An End-to-End Measurement of Certificate Revocation in the Web's PKI // ACM Press, 2015. pp. 183–196.
4. Santesson S. Myers M., Ankney R., Malpani A., Galperin S., Adams C. X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP. 2013. DOI: 10.17487/RFC6960
5. Serrano N., Hadan H., Camp L.J. A Complete Study of P.K.I. (PKI's Known Incidents) // SSRN Electronic Journal. 2019. DOI: 10.2139/ssrn.3425554

6. Бусыгин А.Г., Коноплев А.С., Зегжда Д.П. Схема и инфраструктура обеспечения защищенности коммуникаций в сети Интернет от атаки «человек посередине», основанной на использовании отозванных сертификатов // Материалы конференции «Информационная безопасность регионов России». Санкт-Петербург, 2017. С. 170-171.
7. Laurie B. Certificate Transparency // Queue. 2014. № 8 (12). pp. 10–19.
8. Fromknecht, C., Velicanu, D., Yakoubov, S. A Decentralized Public Key Infrastructure with Identity Retention. // IACR Cryptology ePrint Archive, 2014. p. 803.
9. Yao S. Chen J., He K., Du R., Zhu T., Chen, X. PBCert: Privacy-Preserving Blockchain-Based Certificate Status Validation Toward Mass Storage Management // IEEE Access. 2019. (v.7). pp. 6117–6128. DOI: 10.1109/ACCESS.2018.2889898
10. Scheitle Q., Hohlfeld O., Gamba J., Jelten J., Zimmermann T., Strowes S., Vallina-Rodriguez N. A Long Way to the Top: Significance, Structure, and Stability of Internet Top Lists. In Proceedings of the Internet Measurement Conference 2018 (IMC '18). ACM, New York, NY, USA, 2018, pp. 478-493. DOI: 10.1145/3278532.3278574
11. Porter Felt A., Barnes R., King A., Palmer C., Bentzel C., Tabriz P. Measuring HTTPS Adoption on the Web // 26th USENIX Security Symposium. Vancouver: USENIX Association, 2017. pp. 1323–1338.
12. Kumar D., Wang Z., Hyder M., Dickinson J., Beck G., Adrian D., Mason J., Durumeric Z., Halderman A., Bailey M. Tracking Certificate Misissuance in the Wild // 2018 IEEE Symposium on Security and Privacy (SP). IEEE, 2018. pp. 785–798. DOI: 10.1109/SP.2018.00015
13. Нурмухаметов Р.К., Степанов П.Д., Новикова Т.Р. Технология блокчейн и ее применение в торговом финансировании // Финансовая аналитика: проблемы и решения. 2018. № 2(344). С. 179-190. DOI: 10.24891/fa.11.2.179
14. Ye C. Li G., Cai H., Gu Y., Fukuda A. Analysis of Security in Blockchain: Case Study in 51%-Attack Detecting IEEE, 2018. pp. 15–24. DOI: 10.1109/DSA.2018.00015
15. Kim, S., Ma Z., Murali S., Mason J., Miller A., Bailey M. Measuring Ethereum Network Peers. // IMC '18 Proceedings of the Internet Measurement Conference. pp. 91–104. DOI: 10.1145/3278532.3278542

THE ONLINE CERTIFICATE STATUS PROTOCOL WITH SMART-CONTRACT APPROACH USAGE

N. Markevich[9], I. Livshitz[10]

Abstract.

The study aims *to increase OCSP service availability.*

In this paper methods *of mathematical logic, scanning network methods and distributed system design methods used.*

As a result, *we examined the availability of more than three hundred OCSP responders that store status of SSL-certificates for more than half a million of the most popular websites. The data obtained allowed us to identify the problem that OCSP responders are not always able to provide the required 100% availability. In contrast to the centralized system, the availability of the Ethereum blockchain platform nodes was investigated. This showed that decentralized system availability is higher due to the larger number and prevalence of Ethereum-nodes.*

An approach to building a decentralized certificate status verification service is proposed - a decentralized OCSP responder model is presented and based on it, the functions of a smart contract and the interaction procedure of the system participants are described. The proposed approach allows to increase the availability of the OCSP service by 30% and to distribute the load between OSCP responders. The solution can be used to build highly available public key systems on an Internet scale, as well as for corporate key management systems.

Keywords: *information security, Internet, Certification Authority, OCSP, HTTPS, SSL, Ethereum, blockchain, availability*

References

1. Fadai T., Schrittwieser S., Kieseberg P., Mulazzani M. Trust me, I'm a Root CA! Analyzing SSL Root CAs in Modern Browsers and Operating Systems. In Proceedings - 10th International Conference on Availability, Reliability and Security. IEEE, 2015. pp. 174–179. DOI: 10.1109/ARES.2015.93

9 Nikita Markevich, Graduate student, ITMO University, Saint Petersburg, Russia. E-mail: markevich.nikita1@gmail.com

10 Ilya Livshitz, Dr.Sc., ITMO University, Saint Petersburg, Russia. E-mail: livshitz.il@yandex.ru

2. Cooper D., Santesson S., Farrell S., Boeyen S., Housley R., Polk W. Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile. 2008. DOI: 10.17487/RFC5280
3. Liu Y., Tome W., Zhang L., Choffnes D., Levin D., Maggs B., Mislove A., Schulman A., Wilson C. An End-to-End Measurement of Certificate Revocation in the Web's PKI // ACM Press, 2015. pp. 183–196.
4. Santesson S. Myers M., Ankney R., Malpani A., Galperin S., Adams C. X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP. 2013. DOI: 10.17487/RFC6960
5. Serrano N., Hadan H., Camp L.J. A Complete Study of P.K.I. (PKI's Known Incidents) // SSRN Electronic Journal. 2019. DOI: 10.2139/ssrn.3425554
6. Busygin A.G., Konoplev A.S., Zegzhda D.P. Skhema i infrastruktura obespecheniya zashchishchennosti kommunikacij v seti Internet ot ataki «chelovek poseredine», osnovannoj na ispol'zovanii otozvannyh sertifikatov // Informacionnaya bezopasnost' regionov Rossii. Sankt-Peterburg, 2017. pp. 170-171
7. Laurie B. Certificate Transparency // Queue. 2014. № 8 (12). pp. 10–19.
8. Fromknecht, C., Velicanu, D., Yakoubov, S. A Decentralized Public Key Infrastructure with Identity Retention. // IACR Cryptology ePrint Archive, 2014. p. 803.
9. Yao S. Chen J., He K., Du R., Zhu T., Chen, X. PBCert: Privacy-Preserving Blockchain-Based Certificate Status Validation Toward Mass Storage Management // IEEE Access. 2019. (7). pp. 6117–6128.
10. Scheitle Q., Hohlfeld O., Gamba J., Jelten J., Zimmermann T., Strowes S., Vallina-Rodriguez N. A Long Way to the Top: Significance, Structure, and Stability of Internet Top Lists. In Proceedings of the Internet Measurement Conference 2018 (IMC '18). ACM, New York, NY, USA, 2018, pp. 478-493. DOI: 10.1145/3278532.3278574
11. Porter Felt A., Barnes R., King A., Palmer C., Bentzel C., Tabriz P. Measuring HTTPS Adoption on the Web // 26th USENIX Security Symposium. Vancouver: USENIX Association, 2017. pp. 1323–1338.
12. Kumar D., Wang Z., Hyder M., Dickinson J., Beck G., Adrian D., Mason J., Durumeric Z., Halderman A., Bailey M. Tracking Certificate Misissuance in the Wild // 2018 IEEE Symposium on Security and Privacy (SP). IEEE, 2018. pp. 785–798. DOI: 10.1109/SP.2018.00015
13. Nurmuhametov R.K., Stepanov P.D., Novikova T.R. TEKHNOLOGIYA BLOKCHEJN I EE PRIMENENIE V TORGOVOM FINANSIROVANII // Finansovaya analitika: problemy i resheniya. 2018. №2 (344). DOI: 10.24891/fa.11.2.179
14. Ye C. Li G., Cai H., Gu Y., Fukuda A. Analysis of Security in Blockchain: Case Study in 51%-Attack Detecting IEEE, 2018. pp. 15–24. DOI: 10.1109/DSA.2018.00015
15. Kim, S., Ma Z., Murali S., Mason J., Miller A., Bailey M. Measuring Ethereum Network Peers. // IMC '18 Proceedings of the Internet Measurement Conference. pp. 91–104. DOI: 10.1145/3278532.3278542