Мельников Андрей Кимович – Закрытое акционерное общество «ИнформИнвестГрупп»; e-mail: ak@iigroup.ru; 115432, г. Москва, Проспект Андропова, 18, Бизнес-Центр Nagatino i-Land, корп. 7 “Ломоносов”, эт. 3; тел.: 84957871109; с.н.с.; к.т.н.

Levin Ilya Izrailevich – Southern Federal University; e-mail: iilevin@sfedu.ru; 15, Petrovskaya str., ap. 143, Taganrog, 347928, Russia; phone: +78634612111; head of the department of Intellectual and Multiprocessor Systems; dr. of eng. sc.; professor.

Dordopulo Alexey Igorevich – “Supercomputers and Neurocomputers Research Center” Co. Ltd; e-mail: dordopulo@superevm.ru; 44, 9th lane, Taganrog, 347902, Russia; phone:: +78634477407; head of the Division of Mathematic and Algorithmic Support; cand. of eng. sc.

Pisarenko Ivan Vadimovich – e-mail: pisarenko@superevm.ru; 8, Nizhnyaya Liniya str., Taganrog, 347942, Russia; phone: +78634477407; junior researcher.

Melnikov Andrey Kimovich – “InformInvestGroup” C. C.; e-mail: ak@iigroup.ru; 18, Andropov Av., Nagatino i-Land Business Centre, building 7 “Lomonosov”, floor 3, Moscow, 115432, Russia; phone: +74957871109; senior researcher; cand. of eng. sc.

УДК 004.422 DOI 10.23683/2311-3103-2018-5-48-56

Л.К. Бабенко, И.А. Писарев

ЭЛЕКТРОННОЕ ГОЛОСОВАНИЕ С ПРИМЕНЕНИЕМ МНОЖЕСТВЕННОГО БРОСАНИЯ БЮЛЛЕТЕНЕЙ*

Применение электронного голосования постепенно вытесняет традиционное. Однако вопрос создания честной и надежной системы электронного голосования все еще открыт. Создание надежной системы электронного голосования, удовлетворяющей всем требованиям безопасности, является сложной задачей. В работе представлена система электронного голосования на основе принципа слепых посредников с применением множественного бросания бюллетеней. Слепые посредники позволяют исключить связь голоса пользователя с какими-либо его аутентификационными данными. Принцип множественного бросания бюллетеней основан на использовании обманных бюллетеней и позволяет верифицировать корректность бюллетеня без применения доказательства с нулевым разглашением, а также обеспечивает выполнение требований к системам электронного голосования. Приведена архитектура системы, описаны компоненты, задействованные в процессе проведения голосовании, и их взаимодействие между собой. Описан процесс голосования, состоящий из нескольких этапов: подготовка, голосование, подсчет результатов. Приведены криптографические протоколы, которые используются при передаче данных между компонентами системы. Описаны принципы заполнения бюллетеней в зависимости от количества кандидатов и типа голосования. Описаны типы голосования один из многих и несколько из многих. Обосновано соблюдение требований безопасности системой. Описана процедура обманного голосования, которая применяется для защиты реального голоса избирателя в случае возможного принуждения к определенному выбору кандидата. Описаны основные атаки, которые возможно провести с помощью вредоносного программного обеспечения на клиентском приложении, обоснованы механизмы защиты от той или иной атаки, а также дальнейшие пути доработки системы для противодействия атакам.

Электронное голосование; криптографическая защита; криптографические протоколы; шифрование.

* Работа поддержана грантом РФФИ № 18-07-01347 А.

L.K. Babenko, I.A. Pisarev

ELECTRONIC VOTING USING MULTIPLE CAST BALLOTS

The use of electronic voting is gradually replacing the traditional one. However, the issue of creating an honest and reliable electronic voting system is still open. Creating a reliable e-voting system that meets all security requirements is a difficult task. The paper presents an electronic voting system based on the principle of blind mediators using multiple ballots. Blind mediators allow you to exclude the user's voice from communicating with any of its authentication data. The principle of multiple ballot casting is based on the use of fraudulent bulletins and allows verifying the correctness of the ballot paper without the use of evidence with zero disclosure, as well as ensuring compliance with the requirements for electronic voting systems. The architecture of the system is described, the components involved in the voting process, and their interaction with each other are described. The voting process is described, consisting of several stages: preparation, voting, counting of results. Cryptographic protocols are used, which are used for data transfer between system components. The principles of filling out the ballots are described, depending on the number of candidates and the type of voting. Types of voting are described as one of many and several of many. The compliance with the security requirements of the system is justified. The procedure for fraudulent voting is described, which is used to protect the real voice of the voter in case of possible compulsion to a certain candidate's choice. It describes the main attacks that can be carried out with the help of malicious software on the client application, the mechanisms of protection against this or that attack are justified, as well as further ways of finalizing the system to counter attacks.

Electronic voting; cryptographic security; cryptographic protocols; encryption.

Введение. Безопасность систем электронного голосования зависит от соблюдения требований безопасности [1]. Существует большое количество систем и схем электронного голосования [2–16], в которых обосновываются соблюдение тех или иных требований. Однако на данный момент не существует системы, которая полностью и без каких-либо допущений удовлетворяла всем требованиям безопасности. В каждой системе присутствует ряд ограничений и допущений, принимая во внимание которые можно говорить о соблюдения этих требований. Главной задачей является соблюсти разумный уровень допущений и ограничений, при которых достигается соблюдение свойств безопасности. Кроме того, немаловажной задачей является снижение негативного влияния на систему в целом клиентских компонентов, в которых присутствуют вредоносное аппаратное или программное обеспечение.

Подготовка. Компонентами системы являются клиентское приложение, сервер аутентификации и сервер голосования. Текущая версия системы является модифицированной версией системы, ранее разработанной авторами работы [17], и привносит ряд ключевых моментов. В данной работе рассматривается голосование типа один из многих. На этапе подготовки генерируются случайные идентификаторы ID равные количеству голосующих. Генерируется пара открытый\закрытый ключ pk, sk для симметричного шифра. Каждый ID подписывается секретным ключом и на выходе получаем подпись Sign(ID). Секретный ключ разделяется по схеме разделения секрета на части, которые отдаются уполномоченным людям. Вся безопасность крепится на том, что если хотябы 1 из уполномоченных людей честный, то выборы пройдут честно. Список ID хранится на сервере голосования и не публикуется в открытом виде. Список подписей Sign(ID) публикуется сразу.

Голосование. Множественное бросание бюллетеней. Принцип множественного бросания бюллетеней заключается в создании L количества бюллетеней, где L – количество кандидатов. Сервер присылает идентификатор голосования ID аутентифицированному пользователю. Генерируются обманные идентификаторы FakeID. Пользователь делает свой выбор, например, в нашей случае он голосует за

Petrov. Создается 4 бюллетеня, в случайном порядке выставляются голоса в них, но так, чтобы не было двух одинаково заполненных бюллетеня. Для бюллетеня с реальным голосом используется идентификатор голосования ID, а для остальных FakeID. После чего все бюллетени шифруются асимметрично публичным ключом и отправляются на BB (Bulletin board – доска объявлений), а голосующий может себе оставить 1 бюллетень с обманным FakeID. Для соблюдения свойства устойчивости к принуждению необходимо убрать какие-либо способы доказать, что пользователь проголосовал определенным образом. Поэтому оставить себе бюллетень с реальным голосом нельзя. В тоже время бюллетень с обманным FakeID позволит проверить, что все бюллетени пользователя дошли с вероятностью в данном случае 1/4. Поскольку сервер не знает, какой именно из бюллетеней является реальным, ему придется опубликовать без подмены все. В противном случае, если хотя бы 1 пользователь обнаружит, что бюллетень не дошел до BB, это покажет, что сервер нарушил правила, подменив бюллетень или не опубликовав их. В клиенте используется проверка того, что голос дошел до BB вне зависимости от того хотел ли пользователь проверить свой голос или нет.

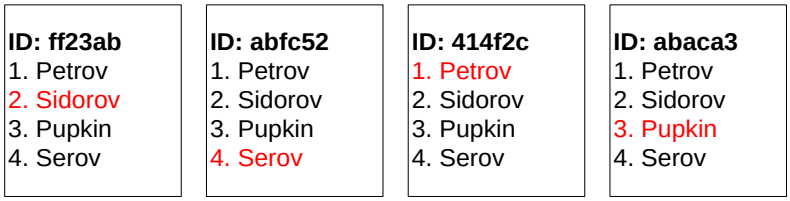

Рис. 1. Пример заполнения бюллетеней для голосования типа 1 из 4

В системе бюллетень представляет собой пару: [ID, Индекс кандидата]. Для примера выше список будет следующим:

```
[ID1,2], [ID2,4], [ID3,1], [ID4,3]
```

Шифруются только идентификаторы и в виде, приведенном ниже, бюллетени отправляются на BB.

```
[Epk(ID1),2], [Epk(ID2),4], [Epk(ID3),1], [Epk(ID4),3]
[EID1,2], [EID2,4], [EID3,1], [EID4,3]
```

Сервер голосования пропускает только бюллетени с равномерным распределением голосов, то есть не должно быть двух одинаково заполненных бюллетеней. Это процесс верификации голоса. В отличие от системы ThreeBallot [18–20], где статично используется 3 бюллетени и другой принцип их заполнения, исключено создание лишних голосов при отсутствии проверки корректности сформированных бюллетеней на стороне клиента. Кроме того в нашем случае проверка голоса осуществляется открыто и сразу, в отличие от применения доказательства с нулевым разглашением в других схемах. На рисунке представлена архитектура системы и направления передаваемых между компонентами данных.

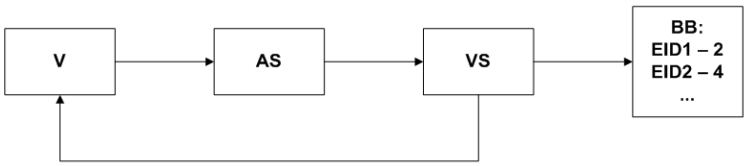

Рис. 2. Архитектура системы

Протокол голосования:

1. *VS->V: E_{vvs}(N)*
2. *V->AS: E_{vas}(AuthData, E_{vvs}(N))*
3. *AS->VS: E_{asvs} (E_{vvs} (N))*
4. *VS->V: E_{vvs} (N, ID, Sign(ID))*
5. *V:*
   5.1. E_{pk} (FakeID1),2: EID1,2
   5.2. E_{pk} (EakeID2),4: EID2,4
   5.3. E_{pk} (ID),1: EID3,1
   5.4. E_{pk} (FakeID3),3: EID4,3
6. *V->VS: E_{vvs} (N, EID1, 2, EID2, 4, EID3, 1, EID4, 3)*
7. *VS->BB:*
   EID1 – 2
   EID2 – 4
   EID3 – 1
   EID4 – 3
8. *VS->V: E_{vvs} (N, Sign(EID1), Sign(EID2), Sign(EID3), Sign(EID4))*
9. *V: EID1 – 2, V check EID1 – 2 on BB and Sign(EID1).*

Перед началом протокола симметричные ключи vas, asvs, vvs генерируются с помощью протокола ECDHE Диффи-Хеллмана на эллиптических кривых с применением эфемерных ключей между сторонами V-AS, AS-VS, V-VS соответственно. В сообщении (1) посылается случайное число для аутентификации. В сообщении (2) с помощью принципа слепых посредников посылается случайное число N зашифрованное на ключе vvs, к этому прикладываются аутентификационные данные пользователя AuthData, все это шифруется на ключе vas и отправляется стороне AS. AS может прочесть только AuthData. Проверяет наличие этих аутентификационных данных в БД и в случае успеха перенаправляет другую часть в виде сообщения (3). VS проверяет значение числа N, и посылает пользователю сообщение (4) с идентификатором для голосования ID и его подписью Sign(ID). На шаге (5) клиент делает свой выбор и клиентское ПО формирует обманные бюллетени вместе с реальной, шифруя каждый идентификатор публичным ключом, после чего посылает пары [EID, голос] серверу голосования в виде сообщения (6). VS посылает бюллетени на BB. VS отправляет обратно подписи EID после чего пользователь выбирает проверочный бюллетень, проверяет его наличие на BB и его подпись Sign(EID).

Подсчет голосов. Процедура подсчета голосов производится открыто. Уполномоченные лица восстанавливают секрет, который является закрытым ключом асимметричного шифрования. Этот ключ публикуется в открытый доступ вместе со всеми ID. Таким образом, на BB имеется:

1. Публичный ключ pk.
2. Секретный ключ sk.
3. Список ID.
4. Список Sign(ID).
5. Список голосов EIDi.

Любой пользователь с помощью клиентского приложения по своему желанию может самостоятельно подсчитать результаты голосования, используя вышеописанные данные. Сама же система подсчитывает голоса следующим образом.

1. Проверяется равенство количества ID с количеством подписей Sign(ID) а так же их сопоставление, то есть для каждого ID из списка есть его подпись Sign(ID).

2. Расшифровываются голоса EIDi.
3. В подсчете участвуют голоса только с ID из списка.
4. При наличии двух и более голосов с одинакового ID – все голоса с таким ID не засчитываются.
5. Публикуются финальные результаты.

Обоснование соблюдения требований. Требований к системам электронного голосования большое количество. Основные описаны в [1] и далее будет приведено обоснование соблюдения этих требований нашей системой.

♦ Eligibility (Право на голосование):

Достигается с помощью сервера аутентификации и принципа слепых посредников. Только пользователи, чьи аутентификационные данные присутствуют в БД могут голосовать.

♦ Fairness (Честность, тайна предварительных результатов):

Достигается использованием схемы разделения секрета при распределении секретного ключа асимметричного шифрования. Если хотя бы 1 из уполномоченных лиц является честным – результаты не могут быть узнаны до окончания выборов.

♦ Individual verifiability (Индивидуальная проверяемость):

После голосования голосующий получает 1 бюллетень из нескольких с выбором невыбранного им кандидата. Бюллетень с обманным FakeID позволит проверить, что все бюллетени пользователя дошли с вероятностью в 1/L (где L – количество кандидатов и используется голосование типа один из многих). Поскольку сервер не знает, какой именно из бюллетеней является реальным, ему придется опубликовать без подмены все. В противном случае, если хотя бы 1 пользователь обнаружит, что бюллетень не дошел до ВВ, это покажет, что сервер нарушил правила, подменив бюллетень или не опубликовав их. В клиенте используется проверка того, что голос дошел до ВВ вне зависимости от того хотел ли пользователь проверить свой голос или нет.

♦ Universal verifiability (Универсальная проверяемость):

Поскольку на ВВ после окончания голосования есть вся информация, с помощью которой можно произвести подсчет, то любой неаутентифицированный пользователь может подсчитать результаты голосования и убедиться в их корректности.

♦ Vote-privacy (Тайна голоса):

Благодаря принципу слепых посредников и создания множества обманных бюллетеней невозможно создать связь аутентификационных данных с идентификатором или определенным бюллетенем.

♦ Receipt-freeness, Coercion-resistance (Свобода от остаточных данных, сопротивление принуждению):

У пользователя есть только 1 бюллетень с обманным FakeID. Таким образом голосующий сможет доказать что он точно не голосовал за одного из кандидатов. На систему наложено ограничение на минимальное количество кандидатов – 3. Из-за чего имея информацию об 1 кандидате, за которого пользователь не голосовал нельзя сделать вывод о том, за кого именно он голосовал. Однако, на данный момент в системе не предусмотрено переголосование. Несмотря на то, что в ряде стран переголосование официально поддерживается, по мнению авторов это необходимо только для соблюдения текущих свойств безопасности. В системе имеется возможность обманного голосования, когда голосующий посылает бюллетени без реального ID в случае возможного принуждения. Предположительно пользователь, у которого есть подозрения, что в его сторону возможно принуждение голосовать определенным образом, сообщает организаторам голосования об этом и выбирает некоторый отрезок времени, в котором будет происходить обманное голосование. В течение этого отрезка времени в БД для аутентифика-

ции это будет помечено, и любые голоса, посланные от лица этого пользователя, будут обманными и в подсчете учувствовать не будут. Это так же предотвращает возможность продажи голосов, поскольку нет никаких доказательств, что пользователь проголосовал определенным образом. Более того нет никаких доказательств, что пользователь вовсе голосовал в действительности с помощью реального идентификатора.

- Vulnerable software/hardware resistance (Сопротивление вредоносному программному или аппаратному обеспечению):

При наличии вредоносного ПО на стороне клиента можно лишь снизить негативное воздействие на систему в целом. Имеются следующие атаки и их последствия:

1. Подмена голоса – частично защищено. Может быть обнаружена с вероятностью 1/L после опубликования данных для подсчета голосов, в случае если проверочный бюллетень будет содержать реальный голос, что противоречит установке системы. Приведет к потере голоса пользователя.
2. Голосование за всех кандидатов – защищено. Обнаруживается на стадии подсчета голосов, поскольку в этом случае за нескольких кандидатов будет несколько бюллетеней с одинаковым ID.
3. Вброс после легальной аутентификации – защищено. Со стороны одного пользователя невозможен, поскольку ему отправляется 1 идентификатор ID для одного голоса.
4. На стороне клиента все бюллетени посылаются с обманных FakeID. Возможны два варианта:
   4.1. Клиентское ПО посылает все бюллетени с обманных FakeID тем самым просто убирая голос пользователя – частично защищено, но приведет к срыву выборов. При подсчете голосов количество проголосовавших пользователей на компоненте сервера аутентификации не совпадет с количеством реальных голосов.
   4.2. С помощью собранных реальных ID клиентское ПО при следующем легальном голосовании может встроить один или несколько лишних голосов, однако только за разных кандидатов. То есть единоразово сделать вброс множества голосов за одного и того же кандидата невозможно. Лучший вариант это смотреть на голос пользователя в текущем легальном голосовании, и если он не совпадает с желаемым голосом злоумышленника – ставить реальный ID напротив еще одного кандидата. Тем самым в результате подсчета количество проголосовавших совпадет с количеством голосов, однако эффективность такого вброса не особо велика. Противодействовать данной атаке можно только после этапа подсчета голосов и повторной проверки уже расшифрованных идентификаторов на ВВ. В таком случае пользователь обнаружит бюллетень с реальным голосом после проверки, что противоречит установке системы.

Голосование типа K из L. Представленная в предыдущих пунктах система позволяет проводить голосование типа 1 из L. В таком случае достаточно создать L – 1 обманных бюллетеней по требованиям описанного алгоритма. Однако в случае голосования типа K из L, где обычно $0 < K < L$, возникает вопрос о необходимом и достаточном количестве обманных бюллетеней и их наполнения. На рис. 3 представлен пример заполнения бюллетеней для голосования типа k из 5.

Как видно из рисунка, количество обманных бюллетеней создается равным L-2. В нашем случае 5 кандидатов, в итоге один бюллетень будет реальным, а 3 остальных – обманные. Наполнение бюллетеней принимается следующим образом:

1. Пользователь делает свой выбор путем выбора кандидатов. В нашем случае для примера возьмем, что пользователь выбрал кандидатов под номером 2 и 4.

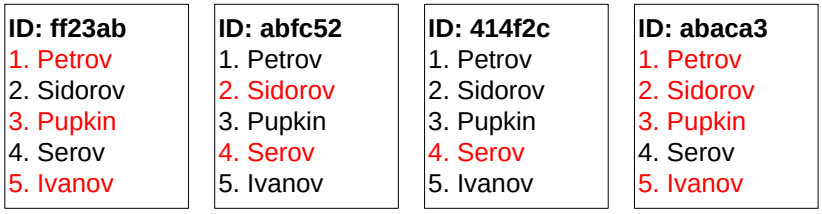

Рис. 3. Пример заполнения бюллетеней для голосования типа k из 5

2. Формируется реальный бюллетень с реальным идентификатором присланным сервером (бюллетень с ID: abfc52).

3. Далее создаются обманные бюллетени таким образом, чтобы в результате в наборе этих бюллетеней количество кандидатов начиналось с 1 и заканчивалось 4. Кандидаты в обманных бюллетенях выбираются случайным образом, однако соблюдается зеркальность голосов. В данном примере бюллетень с выбором 1, 3, 5 зеркальная бюллетеню с выбором 2,4. Это необходимо для того, чтобы в случае подмены голосов это можно было обнаружить. В случае если количество кандидатов четное, то добавляется еще один обманный бюллетень для сохранения свойства зеркальности как на рис. 4.

Рис. 4. Пример заполнения бюллетеней для голосования типа k из 4 с добавлением дополнительного обманного бюллетеня

Такой вид создания обманных бюллетеней позволяет не увеличивать их количество по сравнению с типом голосования 1 из L, и с другой стороны позволит сохранить заложенную в принцип множественного бросания бюллетеней возможность проверки голоса пользователем с сохранением требований Receipt-freeness, Coercion-resistance, поскольку не будет доказательства, каким именно образом голосовал пользователь. Однако у данного подхода существует недостаток – зависимость от распределения голосов. Это не позволяет в большей степени удовлетворить требование тайны предварительных результатов. В случае если за определенных кандидатов будут очень часто голосовать и тем самым разрыв между предположительным победителем на данный момент большой по сравнению с остальными кандидатами, можно будет примерно высчитать текущего победителя выборов. Как правило, это будут повторяющиеся пары зеркальных бюллетеней. В дальнейшем планируется модификация такого подхода для устранения данного недостатка.

Заключение. В работе была представлена система электронного голосования на основе слепых посредников с применением множественного бросания бюллетеней. Приведена архитектура системы, описаны компоненты, задействованные в процессе проведения голосования. Описан процесс голосования, состоящий из нескольких этапов: подготовка, голосование, подсчет результатов. Показаны принципы наполнения бюллетеней при голосованиях типа один из многих и несколько из многих. Обосновано соблюдение требований безопасности системой. Описана процедура обманного голосования, которая применяется для защиты реального голоса избирателя в случае возможного принуждения к определенному выбору кандидата. Описаны основные атаки, которые возможно провести с помощью вредоносного программного обеспечения на клиентском приложении. Показана защищенность предлагаемой системы от приведенных атак и так же дальнейшие пути развития противодействия атакам.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Delaune S., Kremer S., & Ryan, M.* Verifying privacy-type properties of electronic voting protocols, *Journal of Computer Security*, 2009, Vol. 17 (4), pp. 435-487.
2. *Dossogne J., Lafitte F.* Blinded additively homomorphic encryption schemes for self-tallying voting, *Journal of Information Security and Applications*, 2015.
3. *David L Chaum*. Untraceable electronic mail, return addresses, and digital pseudonyms, *Communications of the ACM*, 1981, 24 (2): 84-90.
4. *Izabachene M.* A Homomorphic LWE Based E-voting Scheme, *P ost-Quantum Cryptography: 7th International Workshop, PQCrypto 2016, Fukuoka, Japan, February 24-26, 2016*.
5. *Hirt M., Sako K.* Efficient receipt-free voting based on homomorphic encryption, *International Conference on the Theory and Applications of Cryptographic Techniques*. Springer Berlin Heidelberg, 2000, pp. 539-556.
6. *Rivest L. R. et al.* Lecture notes 15: Voting, homomorphic encryption, 2002.
7. *Ben Adida.* Mixnets in Electronic Voting, Cambridge University, 2005.
8. Electronic elections: fear of falsification of the results. Kazakhstan today, 2004.
9. *Lipen V.Y., Voronetsky M.A., Lipen D.V*. technology and results of testing electronic voting systems. United Institute of Informatics Problems NASB, 2002.
10. *Ali S. T., Murray J.* An Overview of End-to-End Verifiable Voting Systems, *arXiv preprint arXiv: 1605.08554, 2016*.
11. *Smart M., Ritter E.* True trustworthy elections: remote electronic voting using trusted computing, *International Conference on Autonomic and Trusted Computing*. Springer Berlin Heidelberg, 2011, pp. 187-202.
12. *Bruck S., Jefferson D., Rivest R.L.* A modular voting architecture ("frog voting"), *Toward strustworthy elections*. Springer Berlin Heidelberg, 2010.
13. *Jonker H., Mauw S., Pang J.* Privacy and verifiability in voting systems: Methods, developments and trends // Computer Science Review, 2013.
14. *Shubhangi S. Shinde, Sonali Shukla, Prof. Chitre D.K*. Secure E-voting Using Homomorphic Technology, *International Journal of Emerging Technology and Advanced Engineering*, 2013.
15. *Neumann S., Volkamer M.* Civitas and the real world: problems and solutions from a practical point of view, *Availability, Reliability and Security (ARES), 2012 Seventh International Conference on*. IEEE, 2012, pp. 180-185.
16. *Yi X., Okamoto E.* Practical remote end-to-end voting scheme, *International Conference on Electronic Government and the Information Systems Perspective*. Springer Berlin Heidelberg, 2011, pp. 386-400.
17. *Babenko L., Pisarev I., Makarevich O*. A model of a secure electronic voting system based on blind intermediaries using Russian cryptographic algorithms, *In Proceedings of the 10th International Conference on Security of Information and Networks (SIN '17). ACM, New York, NY, USA, 2017*, pp. 45-50.

18. *Rivest R.L., Smith W.D.* Three voting protocols: ThreeBallot, VAV, and Twin, *USENIX/ACCURATE Electronic Voting Technology (EVT 2007), 2007*.
19. *Santin A.O., Costa R.G., Maziero C.A.* A three-ballot-based secure electronic voting system, *IEEE Security & Privacy*, 2008, Vol. 6, No. 3, pp. 14-21.
20. *Henry K.J., Stinson D.R., Sui J.* The effectiveness of receipt-based attacks on threeballot, IEEE *Transactions on Information Forensics and Security*, 2009, Vol. 4, No. 4, pp. 699.

Статью рекомендовал к опубликованию д.т.н., профессор Я.Е. Ромм.

Бабенко Людмила Климентьевна – Южный федеральный университет; e-mail: blk@tsure.ru; г. Таганрог, 18-ый переулок, 43; тел.: 89054530191; кафедра безопасности информационных технологий; д.т.н.; профессор.

Писарев Илья Александрович – e-mail: ilua.pisar@gmail.com; г. Таганрог, ул. Котлостроительная 7, кв. 35; тел.: 89885350837; кафедра безопасности информационных технологий; студент.

Babenko Ludmila Klimentevna – Southern Federal University; e-mail: blk@tsure.ru; 18th Lane, 43, Taganrog, Russia; phone: +79054530191; the department of information technology security; dr. of eng. sc.; professor.

Pisarev Ilya Aleksandrovich – e-mail: ilua.pisar@gmail.com; 7, Kotlostroitelnaia street, Apt. 35, Taganrog, Russia; phone: +79885350837; the department of information technology security; student.

УДК 621.396.624 DOI 10.23683/2311-3103-2018-5-56-68

К.Е. Румянцев, Е.А. Рудинский

ОЦЕНКА ПАРАМЕТРОВ ДВУХЭТАПНОГО АЛГОРИТМА ОДНОФОТОННОЙ СИНХРОНИЗАЦИИ АВТОКОМПЕНСАЦИОННОЙ СИСТЕМЫ КВАНТОВОГО РАСПРЕДЕЛЕНИЯ КЛЮЧА

Решение проблемы обеспечения секретности при распределении ключа базируется на принципах квантовой криптографии и предполагает кодирование квантового состояния фотона. Анализ стратегий съёма информации посредством атак в квантовых системах показывает, что реализация многофотонного режима в процессе синхронизации потенциально упрощает злоумышленнику организацию несанкционированного доступа к информации. Последнее определяет актуальность разработки и исследования методов и алгоритмов синхронизации систем квантового распределения ключа (КРК) с автоматической компенсацией поляризационных искажений, обеспечивающих повышенную защищённость процесса от несанкционированного доступа. Исследована защита процесса синхронизации автокомпенсационных системы КРК от несанкционированного доступа. Предложенный алгоритм синхронизации позволяет использовать однофотонные лавинные фотодиоды (ОЛФД) со значительной временной задержкой между моментами приёма фотона и восстановления ОЛФД. В данной работе описан графоаналитический метод оценки параметров однофотонной синхронизации системы распределения квантовых ключей с автоматической компенсацией. В работе формируется и описывается диаграмма состояний и переходов для случайного поиска импульса фотона с учетом вероятности ложных срабатываний. На основе результатов анализа определены выражения для вычисления математического ожидания, дисперсии и среднего количества шагов, затраченных на процесс синхронизации. Определён интервал для среднего числа шагов, используемых в процессе синхронизации. Среднее время синхронизации согласно предложенному алгоритму оценивается с учетом вероятности ложной тревоги. Расчеты показали, что ошибка в определении среднего времени синхронизации согласно представленным выражениям составляет не более 1,7 %. Таким образом, графоаналитический метод оценки параметров однофотонной синхронизации системы автокомпенсационной системы КРК позволяет определить статистические характеристики и среднее время синхронизации с низким уровнем ошибки.

Квантовое распределение ключа; синхронизация; защита; волоконно-оптическая линия; двухэтапный алгоритм; однофотонная регистрация.