Батуев Кирилл Александрович

студент

Кротова Елена Львовна

канд. физ.-мат. наук, доцент

ФГБОУ ВПО «Пермский национальный

исследовательский политехнический университет»

г. Пермь, Пермский край

ТЕОРЕТИЧЕСКИЕ ПРОТОКОЛЫ ЭЛЕКТРОННОГО ГОЛОСОВАНИЯ

***Аннотация**: данная статья посвящена основным протоколам электронного голосования. В работе рассматриваются алгоритмы работы протоколов, их достоинства и недостатки.*

***Ключевые слова**: электронное голосование, протоколы электронного голосования, простой протокол голосования, протокол Нурми-Салома-Сантина, протокол двух агентств, протокол Фудзиока-Окамото-Охта, протокол Не-Su.*

Протоколы электронного голосования – это протоколы обмена данными для безопасного голосования через электронные технические средства.

Многие страны уже внедряют данный тип голосования и для уверенности в целостности, конфиденциальности и доступности выборов используют протоколы с доказанной защищенностью, которые должны реализовывать обязательные требования в том, что только голосующий может знать свой выбор, что проголосовать можно один раз и только участником, допущенным к выборам и решение проголосовавшего не может быть кем-либо изменено.

Рассмотрим алгоритм простого протокола электронного голосования:

Шаг 1. Агентство, проводящее электронное голосование (далее *A*) выкладывает списки возможных участников выборов.

Шаг 2. Участник, допущенный к выборам (далее *B*) сообщает о своем намеренье участвовать в голосовании.

Шаг 3. *A* выкладывает списки зарегистрированных *B*.

Шаг 4. *A* создает закрытый ($К_{А_{зак}}$) и открытый ($К_{А_{отк}}$) ключ и выкладывает в общий доступ $К_{А_{отк}}$, чтобы любой мог зашифровать сообщение, но расшифровать мог только *A*.

Шаг 5. *B* создает свои ключи $К_{В_{зак}}$ и $К_{В_{отк}}$ и выкладывает в общий доступ $К_{В_{отк}}$, чтобы любой мог проверить его электронный избирательный бюллетень (далее *C*), но подписать мог только он сам.

Шаг 6. *B* формирует сообщение *C*, где выражает свой выбор, подписывает $К_{В_{зак}}$, шифрует $К_{А_{отк}}$ и отправляет *A*.

Шаг 7. *A* собирает *C*, расшифровывает с помощью $К_{В_{отк}}$ и публикует подсчитанные результаты.

Довольно простой протокол, помогает защититься от подделки голосов и внешнего вмешательства, но *B* должен доверять *A*, чья работа никем не контролируется.

Далее рассмотрим алгоритм протокола Нурми-Салома-Сантина или другими словами протокола двух агентств:

Шаг 1. Валидатор (далее *V*) отправляет секретные опознавательные метки (далее *M*) всем *B* до голосования.

Шаг 2. *V* отправляет *A* весь набор *M*, но без информации о том, кому они принадлежат.

Шаг 3. *B* создает свои ключи $К_{В_{зак}}$, $К_{В_{отк}}$ и выкладывает в общий доступ $К_{В_{отк}}$, а также создает секретный ключ ($К_{В_{сек}}$), который нужен, чтобы никто не узнал содержимое бюллетеня до нужного момента.

Шаг 4. *B* формирует сообщение *C*, где выражает свой выбор, подписывает $К_{В_{зак}}$, прикладывает к нему полученную *M* и шифрует $К_{В_{сек}}$.

Шаг 5. К зашифрованному тексту *B* прикладывает *M* и отправляет *A*.

Шаг 6. *A* получает зашифрованный текст, по *M* определяет, что он пришел от *B*, но не знает от кого именно и как *B* проголосовал, после публикует его.

Шаг 7. Опубликованный зашифрованный текст служит информацией, чтобы *B* отправил $К_{В_{сек}}$.

Шаг 8. *A* собирает ключи, расшифровывает текст, подсчитывает голоса и присоединяет к опубликованному зашифрованному тексту *C* без *M*.

На 6 шаге *A* не сможет отрицать, что не получал сообщения от *B*. Благодаря публикации зашифрованного текста и бюллетеня, каждый *B* может проверить, что его голос был учтен должным образом. Данный протокол имеет минусы, если *A* вступит в тайный сговор с *V*, то он сможет манипулировать голосованием, специально не принимая сообщения от некоторых *B*. Так же присутствует проблема «мертвых душ», если *V* специально внесет несуществующих *B*, то *A* сможет фальсифицировать бюллетени от них.

Следующим рассмотрим алгоритм протокола Фудзиока-Окамото-Охта. Частично решает проблему сговора двух агентств. Работа протокола заключается в заранее выбранном способе маскирующего шифрования – это особый вид шифрования, который позволяет убедиться, что документ подлинный и был подписан авторизированным пользователем, но не дает информации о содержащихся данных. Алгоритм выглядит следующим образом:

Шаг 1. *V* утверждает список *B*.

Шаг 2. *B* создает свои ключи $К_{В_{зак}}$, $К_{В_{отк}}$, $К_{В_{сек}}$ и выкладывает в общий доступ $К_{В_{отк}}$.

Шаг 3. *B* формирует сообщение *C*, где выражает свой выбор, шифрует его $К_{В_{сек}}$, маскирует, подписывает $К_{В_{зак}}$ и отправляет *V*.

Шаг 4. *V* создает свои ключи $К_{V_{зак}}$, $К_{V_{отк}}$ и выкладывает в общий доступ $К_{V_{отк}}$.

Шаг 5. *V* удостоверяется, что *C* принадлежит *B*, который еще не голосовал, подписывает его $К_{V_{зак}}$ и отправляет *B*.

Шаг 6. *B* удаляет слой маскирующего шифрования и отправляет сообщение *C* к *A*.

Шаг 7. *A* проверяет подписи *B* и *V* и помещает зашифрованный *C* в специальный список, который будет опубликован после голосования.

Шаг 8. После публикации списка, *B* отправляет *A* свой $К_{В_{сек}}$.

Шаг 9. *A* собирает ключи, расшифровывает *C* и подсчитывает голоса, при этом публикует декодирующие ключи вместе с зашифрованными *C*, чтобы *B* смогли самостоятельно проверить результаты голосования.

В 1996 году было предложено модифицировать протокол, который получил название Sensus. Дополнением является то, что после помещения зашифрованного *C* в специальный список *A* отправляет подписанный *C* обратно *B* в качестве квитанции, таким образом можно закончить голосование в течении одного сеанса. Это удобно для конченого пользователя и дает дополнительные гарантии участия в выборах.

Если *A* вступит в тайный сговор с *V*, то уже не сможет опознать *B* до получения ключа. Остается лишь проблема подачи голосов за *B* не пришедших на выборы.

Рассмотри алгоритм протокола He-Su. Данная схема решает проблему тайного сговора *A* и *V*. Как в предыдущих протоколах используется идея слепой подписи, но подписывается открытый ключ *B*, а не его бюллетень. Это позволяет скорректировать свой голос до окончания голосования. Алгоритм:

Шаг 1. *V* утверждает список *B*, создает свои ключи $К_{V_{зак}}$, $К_{V_{отк}}$ и выкладывает в общий доступ $К_{V_{отк}}$.

Шаг 2. *B* создает свои ключи $К_{В_{зак}}$, $К_{В_{отк}}$, генерирует случайное число R вычисляет хэш-функцию h от $К_{В_{отк}}$, маскирует ее R и отправляет *V* полученную функцию, которая выглядит следующим образом: $f = К_{V_{отк}}(R) \cdot h(К_{В_{отк}})$.

Шаг 3. *V* удостоверяется в праве *B* голосовать, подписывает полученное сообщение, т.е. $К_{V_{зак}}\left(К_{V_{отк}}(R) \cdot h\left(К_{В_{отк}}\right)\right) = R \cdot К_{V_{зак}}\left(h\left(К_{В_{отк}}\right)\right)$ и отправляет *B*.

Шаг 4. *B* удаляет слой маскирующего шифрования, проверяет подлинность подписи *V*, т. е. $К_{V_{отк}}\left(К_{V_{зак}}\left(h\left(К_{В_{отк}}\right)\right)\right) = h\left(К_{В_{отк}}\right)$ и отправляет *A* $К_{В_{отк}}$ и подпись *V*, т.е. $К_{V_{зак}}\left(h\left(К_{В_{отк}}\right)\right)$.

Шаг 5. *A* проверяет подлинность подписи *V*, проверяет совпадение хэш-функции от $К_{В_{отк}}$ с той, что хранится в подписи *V*, добавляет $К_{В_{отк}}$ в список авторизованных ключей и сообщает об этом *B*.

Шаг 6. *B* формирует сообщение *C*, где выражает свой выбор, шифрует его созданным $К_{В_{сек}}$ и отправляет *A* набор состоящий из $К_{В_{отк}}$, зашифрованного $К_{В_{сек}}$ сообщение *C* и зашифрованную $К_{В_{зак}}$ хэш-функцию от зашифрованного $К_{В_{сек}}$ сообщения *C*.

Шаг 7. *A* проверяет $К_{В_{отк}}$ со списком, созданным ранее, сравнивает хэш-функцию сообщения *C* зашифрованного $К_{В_{сек}}$ и хэш-функцию, полученную при помощи $К_{В_{зак}}$ и публикует весь набор в открытом списке.

Шаг 8. После публикации списка *B* отправляет *A* новый набор состоящий из $К_{В_{отк}}$, $К_{В_{сек}}$ и зашифрованную $К_{В_{зак}}$ хэш-функцию от $К_{В_{сек}}$.

Шаг 9. *A* проверяет подлинность $К_{В_{сек}}$, сравнивая хэш-функцию от $К_{В_{сек}}$ и хэш-функцию полученную при помощи $К_{В_{зак}}$, если все верно, то расшифровывает полученную ранее *C*, публикует все данные и подсчитывает голоса.

Шаг 10. После голосования *V* публикует утвержденный список *B*, а *A* – список авторизованных ключей.

A и *V* не могут тайно сговориться, потому что публикуют списки, поэтому нельзя внести несуществующих избирателей и проголосовать за не пришедших. Минусами является уязвимость перед DoS-атаками, так как требуется большое количество ресурсов для поддержания работоспособности протокола из-за его сложности.

Таким образом, видно, что существуют различные протоколы электронного голосования и это только часть из них, которые спроектированы для минимизации мошенничества. Остальные протоколы менее известны, так как применяются для достижения дополнительных целей и особенностей голосования. Многие протоколы практически не осуществимы, но влекут за собой теоретический интерес.

Список литературы

1. Протоколы тайного голосования // Википедия [2013–2016] [Электронный ресурс]. – Режим доступа: http://ru.wikipedia.org/?oldid=77066271 (дата обращения: 09.05.2016).