Протокол электронного анонимного голосования

Минусы классического голосования

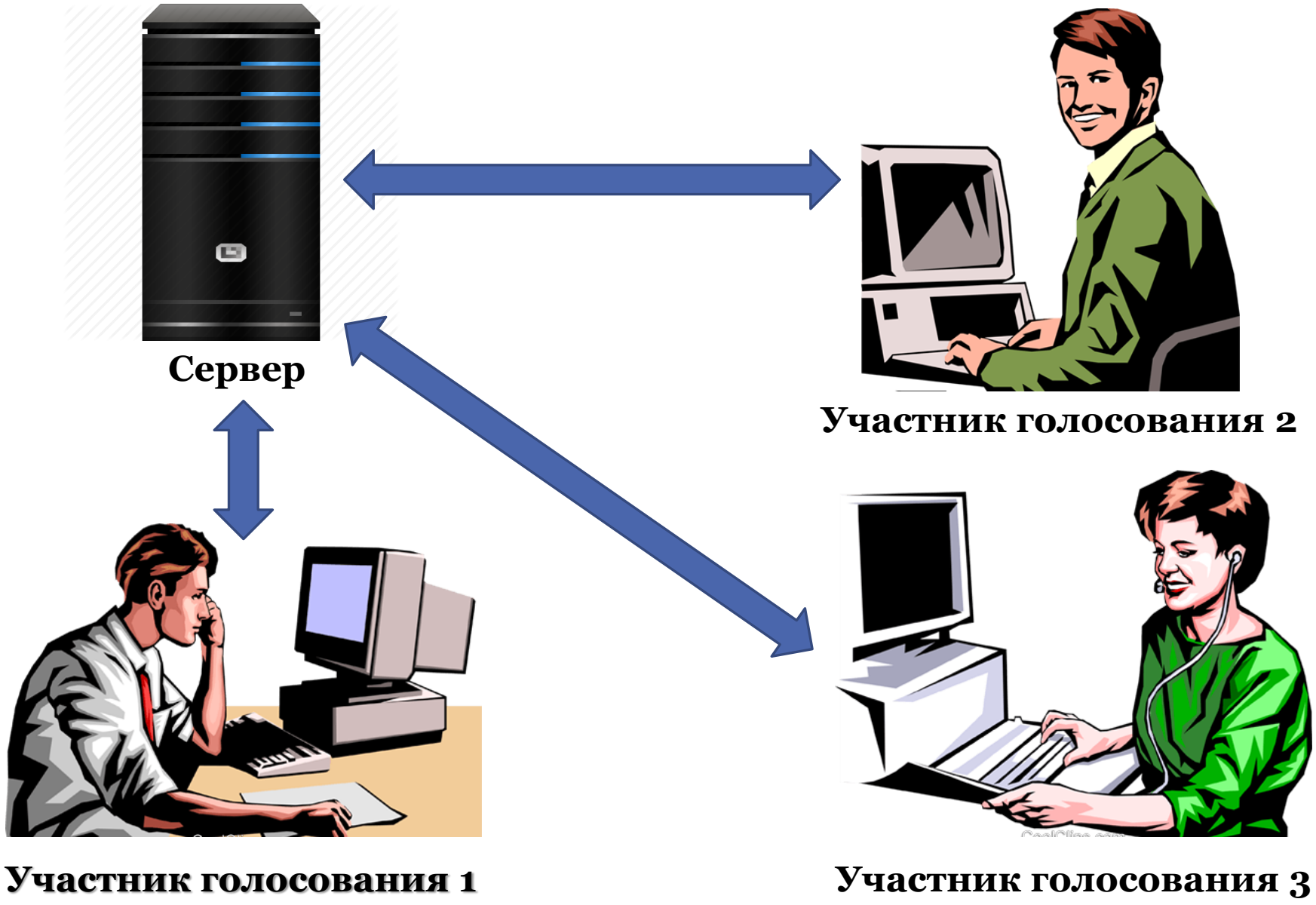
- Необходимость личного присутствия
- Человеческий фактор

Достоинства дистанционного голосования

- Возможность участия из любой точки мира
- Надёжность, обоснованная криптографическими протоколами

Обязательные требования к системам тайного голосования:

- Анонимность голосующих
- Легитимность голосования
- Уникальность каждого бюллетеня
- Решение голосующего не может быть изменено



Необязательные требования

- Каждый участник может проверить, правильно ли зачтён его голос
- Каждый участник может передумать и изменить свой выбор в течение определённого периода времени
- Система должна быть защищена от продажи голосов избирателями
- В случае, если голос зачтён неправильно, каждый участник может сообщить об этом системе

Необязательные требования

- Невозможно отследить, откуда дистанционно проголосовал избиратель
- Можно узнать, кто принимал участие в голосовании, а кто — нет
- Поддержание системы не должно требовать много ресурсов
- Система должна быть отказоустойчива в случае технических неисправностей (потеря электропитания), непреднамеренных (потеря избирателем ключа) и злоумышленных (намеренная выдача себя за другого избирателя, DoS/DDoS) атак.

Простой протокол тайного цифрового голосования

A — агентство, проводящее электронное голосование (англ. *agency*),

E — избиратель, легитимный участник голосования (англ. *elector*),

B — цифровой бюллетень (англ. *bulletin*).

Шаг 1. A выкладывает списки возможных избирателей.

Шаг 2. Пользователи, в числе которых и **E**, сообщают о желании участвовать в голосовании.

Шаг 3. A выкладывает списки легитимных избирателей.

Шаг 4. A создаёт открытый и закрытый ключ a_{public} и a_{private} и выкладывает в общий доступ a_{public}.
Кто угодно может зашифровать сообщение при помощи a_{public}, но расшифровать его сможет только **A**.

Шаг 5. E

- создаёт собственные публичный и приватный ключи ЭЦП e_{public} и e_{private}, затем публикует открытый ключ. Кто угодно может проверить документ **E**, но подписать его — только сам избиратель. Этот шаг пропускается, если **A** уже знает электронные подписи избирателей (например, они были сгенерированы при регистрации в системе).
- формирует сообщение **B**, где тем или иным способом выражает свою волю
- подписывает сообщение личным закрытым ключом e_{private}
- шифрует сообщение открытым ключом a_{public}
- отправляет шифрованное сообщение **A**

Шаг 6. A

- собирает сообщения
- расшифровывает их при помощи лежащего в открытом доступе e_{public}
- подсчитывает их и публикует результаты

Достоинства простого протокола голосования

- Простота
- Защита от внешнего вмешательства, подделки голосов и дискредитации легитимных избирателей.

Недостатки простого протокола голосования

- Голосующим приходится абсолютно доверять А
- Е может предоставить злоумышленнику-покупателю голосов доказательство, как он проголосовал, но не может проверить, что А правильно учёл или даже получил его бюллетень
- Можно отследить, откуда пришел бюллетень

Протокол двух агентств (Нурми — Саломаа — Сантина)

A — агентство, проводящее электронное голосование (англ. *agency*),

E — избиратель, легитимный участник голосования (англ. *elector*),

B — цифровой бюллетень (англ. *bulletin*).

V — регистратор (англ. *validator*), в обязанности которого входит подготовка списков, а также допуск или недопуск участника до голосования

Шаг 1. V

- создаёт набор опознавательных меток t_i и утверждает список возможных избирателей
- отправляет по защищённому каналу по одной метке каждому голосующему
- отправляет **A** весь набор меток без информации о том, какая метка кому принадлежит

Шаг 2. E

- генерирует e_{public}, e_{private} (для цифровой подписи) и e_{secret} (для того, чтобы ни **A**, ни посторонний злоумышленник не мог до нужного времени узнать содержимое бюллетеня)
- e_{public} публикуется
- формирует сообщение **B** с выбранным решением
- подписывает его e_{private}
- прикладывает к нему полученный t_i
- шифрует при помощи e_{secret}
- снова прикладывает к шифротексту t_i
- отправляет шифротекст $\left\{t_i, \mathbf{encrypt}\left(e_{\text{secret}}, \{t_i, \mathbf{sign}(e_{\text{private}}, B)\}\right)\right\}$ на рассмотрение в **A**

Шаг 3. A

- получает шифротекст. По внешнему тегу оно определяет, что сообщение пришло от легитимного пользователя, но не может определить, ни от какого, ни как он проголосовал.
- выкладывает в открытый доступ полученную пару тег-шифр

Шаг 4. Опубликованный файл служит сигналом **E** отправить секретный ключ e_{secret}

Шаг 5. A

- собирает ключи
- расшифровывает сообщения
- производит подсчёт голосов
- присоединяет к опубликованному шифротексту бюллетень без опознавательного тега, на чём голосование заканчивается.

Достоинства протокола двух агентств

- Благодаря выкладыванию в общий доступ полученного файла на шаге 3, A не может впоследствии отрицать получение сообщения от E. При помощи пары шифр — бюллетень каждый избиратель может проверить, правильно ли был учтён его голос, что устраняет проблему с недостатком контроля над A.

Недостатки протокола двух агентств

- Если A и V сговорятся, A может манипулировать голосованием.
- Если агентству известно, кто скрывается под каким опознавательным тегом, оно может специально не принимать сообщения от некоторых избирателей.
- Проблема «мёртвых душ». Если V внесёт в список заведомо несуществующих избирателей, то A сможет фальсифицировать бюллетени от них.

Протокол Фудзиоки — Окамото — Оты

A — агентство, проводящее электронное голосование (англ. *agency*),

E — избиратель, легитимный участник голосования (англ. *elector*),

B — цифровой бюллетень (англ. *bulletin*).

V — регистратор (англ. *validator*), в обязанности которого входит подготовка списков, а также

допуск или недопуск участника до голосования

Шаг 1. V утверждает списки легитимных избирателей

Шаг 2. E

- создаёт e_{public}, e_{private} (для цифровой подписи) и e_{secret} (для того, чтобы ни **A**, ни посторонний злоумышленник не мог до нужного времени узнать содержимое бюллетеня)
- подготавливает сообщение **B** с выбранным решением
- шифрует его e_{secret}
- накладывает слой ослепляющего шифрования
- подписывает его e_{private}
- отправляет **V** $\text{blind}\left(\text{sign}(e_{\text{private}}, \text{encrypt}(e_{\text{secret}}, B))\right)$

Шаг 3. V

- создаёт v_{public} и v_{private}, публичный ключ выкладывается в общий доступ
- удостоверяется, что бюллетень действительный и принадлежит легитимному и не голосовавшему избирателю
- подписывает его v_{private}
- возвращает его **E**

Шаг 4. E снимает с бюллетени слой маскирующего шифрования (в силу коммутативности остаётся $\text{sign}\left(v_{\text{private}}, \text{sign}\left(e_{\text{private}}, \text{encrypt}(e_{\text{secret}}, B)\right)\right)$) и отправляет её **A**

Шаг 5. A

- проверяет подписи **E** и **V**
- помещает всё ещё зашифрованную e_{secret} бюллетень в специальный список, который будет опубликован после того как все избиратели проголосуют или по истечении заранее оговорённого срока

Шаг 6. После того как список появляется в открытом доступе, **E** высылает **A** e_{secret}

Шаг 7. A

- расшифровывает сообщение
- подсчитывает результаты

Достоинства протокола Фудзиоки — Окамото — Оты

- Если агентствам удастся сговориться, А не сможет опознать избирателей до того, как получит ключ.
- Хотя всё ещё есть возможность не принимать сообщения, отпадает возможность игнорировать сообщения конкретно от «неугодных» избирателей.

Недостатки протокола Фудзиоки — Окамото — Оты

- Проблема подачи голосов за избирателей, не пришедших на выборы.
- Чтобы позволить избирателю переголосовать, в том числе и из-за технической ошибки, необходим дополнительный модуль.

Протокол Sensus
(модификация Фудзиоки — Окамото — Оты)

Отличие в шагах 5—6:

- После того, как A получило зашифрованное сообщение от E, оно добавляет его в публикуемый список, вдобавок отправляет подписанный бюллетень обратно избирателю в качестве квитанции.
- Таким образом E не нужно ждать, пока проголосуют все остальные, и он может закончить голосование за один сеанс.
- Это не только удобно для конечного пользователя, но ещё и предоставляет дополнительного доказательство, что E участвовал в выборах.
- В Sensus регламентированы дополнительные вспомогательные модули, упрощающие и автоматизирующие ход голосования.

Протокол Хэ — Су

A — агентство, проводящее электронное голосование (англ. *agency*),

E — избиратель, легитимный участник голосования (англ. *elector*),

B — цифровой бюллетень (англ. *bulletin*).

V — регистратор (англ. *validator*), в обязанности которого входит подготовка списков, а также допуск или недопуск участника до голосования

Шаг 1. V

- утверждает списки легитимных избирателей
- создаёт v_{public} и v_{private} (используются для асимметричного шифрования)
- v_{public} выкладывается в свободный доступ

Шаг 2. E

- создаёт e_{public} и e_{private} (используются для подписей)
- вычисляет хеш-функцию от публичного ключа: $\mathrm{h}(e_{\text{public}})$
- накладывает слой маскирующего шифрования на $\mathrm{h}(e_{\text{public}})$. Так как шифруется только хеш от ключа, а не длинное сообщение, можно выбрать какой-нибудь простой способ. Например, **E** может сгенерировать случайное число x и вычислить
  $f = \mathbf{encrypt}(v_{\text{public}}, x)\,\mathrm{h}(e_{\text{public}})$
- отправляет f **V**

Шаг 3. V

- проверяет легитимность избирателя
- дешифрует f:
  $g = \mathbf{decrypt}\left(v_{\text{private}}, \mathbf{encrypt}(v_{\text{public}}, x)\,\mathrm{h}(e_{\text{public}})\right) = x\,\mathbf{decrypt}\left(v_{\text{private}}, \mathrm{h}(e_{\text{public}})\right)$.
  Часть $e^{\text{signed}}_{\text{public}} = \mathbf{decrypt}\left(v_{\text{private}}, \mathrm{h}(e_{\text{public}})\right)$ считается подписанным ключом
- отправляет g **E**

Шаг 4. E

- снимает слой ослепляющего шифрования (умножает на обратный элемент x) и получает подписанный ключ $e_{\text{public}}^{\text{signed}}$
- проверяет подлинность подписи регистратора: выполняется ли

$$\text{encrypt}\left(v_{\text{public}}, \text{decrypt}\left(v_{\text{private}}, \text{h}(e_{\text{public}})\right)\right) = \text{h}(e_{\text{public}})$$

- отправляет **A** пару $\left\{e_{\text{public}}, e_{\text{public}}^{\text{signed}}\right\}$

Шаг 5. A

- как и **E** проверяет подлинность подписи регистратора
- проверяет, совпадает ли хеш-функция от e_{public} в паре с той, что хранится в $e_{\text{public}}^{\text{signed}}$
- добавляет e_{public} в список авторизированных ключей и сообщает об этом **E**

Шаг 6. E

- создаёт e_{secret} (используется для шифровки бюллетеней, чтобы ни **A** ни внешний злоумышленник до нужного времени не мог узнать содержимое бюллетеня)
- подготавливает сообщение **B** с выбранным решением
- отправляет **A** набор

$$\left\{e_{\text{public}}, \text{encrypt}\left(e_{\text{secret}}, B\right), \text{sign}\left(e_{\text{private}}, \text{h}(\text{encrypt}(e_{\text{secret}}, B))\right)\right\}$$

Шаг 7. A

- проверяет авторизованность ключа
- проверяет подлинность сообщения сравнивая хеш зашифрованного сообщения и хеш, полученный при помощи $e_{private}$
- публикует тройку в открытом списке

Шаг 8. Появление тройки в открытом списке сигнализирует **E** отправить **A** новый набор:
$\{e_{\text{public}}, e_{\text{secret}}, \text{sign}\,(e_{\text{private}}, \text{h}(e_{\text{secret}}))\}$

Шаг 9. A

- проверяет подлинность сообщения, сравнивая хеши
- расшифровывает ранее полученную бюллетень
- публикует все данные
- подсчитывает результат

Шаг 10. После голосования **V** публикует список всех зарегистрировавшихся избирателей, а **A** — список всех авторизованных ключей.

Достоинства протокола Хэ – Су

- A и V не могут жульничать, так как теперь публикуются все списки: возможных избирателей, зарегистрировавшихся и авторизированных ключей.
- Нельзя внести несуществующих избирателей, голосовать за существующих, но не пришедших. При этом во время составления этих списков ни избирательное агентство, ни регистратор дополнительной информации не получают.
- У избирателей есть возможность изменить голос.

Недостатки протокола Хэ – Су

- Его сравнительная сложность.
- Так как для поддержания протокола необходимо большое количество ресурсов, он уязвим перед DoS-атаками.

Описание алгоритма голосования:

Пусть в системе голосований имеется сервер, который выдаёт бюллетени.

1. В начале каждого голосования сервер генерирует общие данные согласно RSA:

Выбираются случайные большие числа P **(1024 бит)** и Q **(1024 бит)**, на их основе создаются числа $N = PQ$ и $\phi(N) = (P-1)(Q-1)$.

Затем выбирается случайное число $0 < D < \phi(N)$, взаимнопростое с $\phi(N)$. Далее вычисляется $C = D^{-1} mod\ \phi(N)$.

Секретные параметры: P, Q, C

Открытые параметры: N, D

2. Пользователь собирается проголосовать, тогда клиентская сторона системы:

- Формирует некоторое число $\boldsymbol{n = rnd|v}$ **(1024 бит)**,

где $\boldsymbol{v}$ - закодированный результат голосования + служебная информация (512 бит).

- Формирует некоторое случайное число $\boldsymbol{r}$, взаимно простое с $\boldsymbol{N}$.
- Вычисляет криптографическую хэш-функцию от числа **n**.

$$\boldsymbol{h = SHA3(n),\ h < N}$$

- Находит число $\boldsymbol{\overline{h} = hr^D mod\ N}$, затем оно отправляется по защищенному верифицированному каналу на сервер.

3. Сервер помечает, что выдал бюллетень пользователю (это необходимо, чтобы участник не мог проголосовать дважды), вычисляет $\bar{s} = \bar{h}^{c} \boldsymbol{mod}\ \boldsymbol{N}$ и отсылает $\bar{s}$ обратно клиенту.

4. Клиентская сторона вычисляет подпись своего бюллетеня по формуле

$$s = \bar{s} r^{-1} \boldsymbol{mod}\ \boldsymbol{N}$$

где r^{-1} – это инверсия числа r по модулю N.

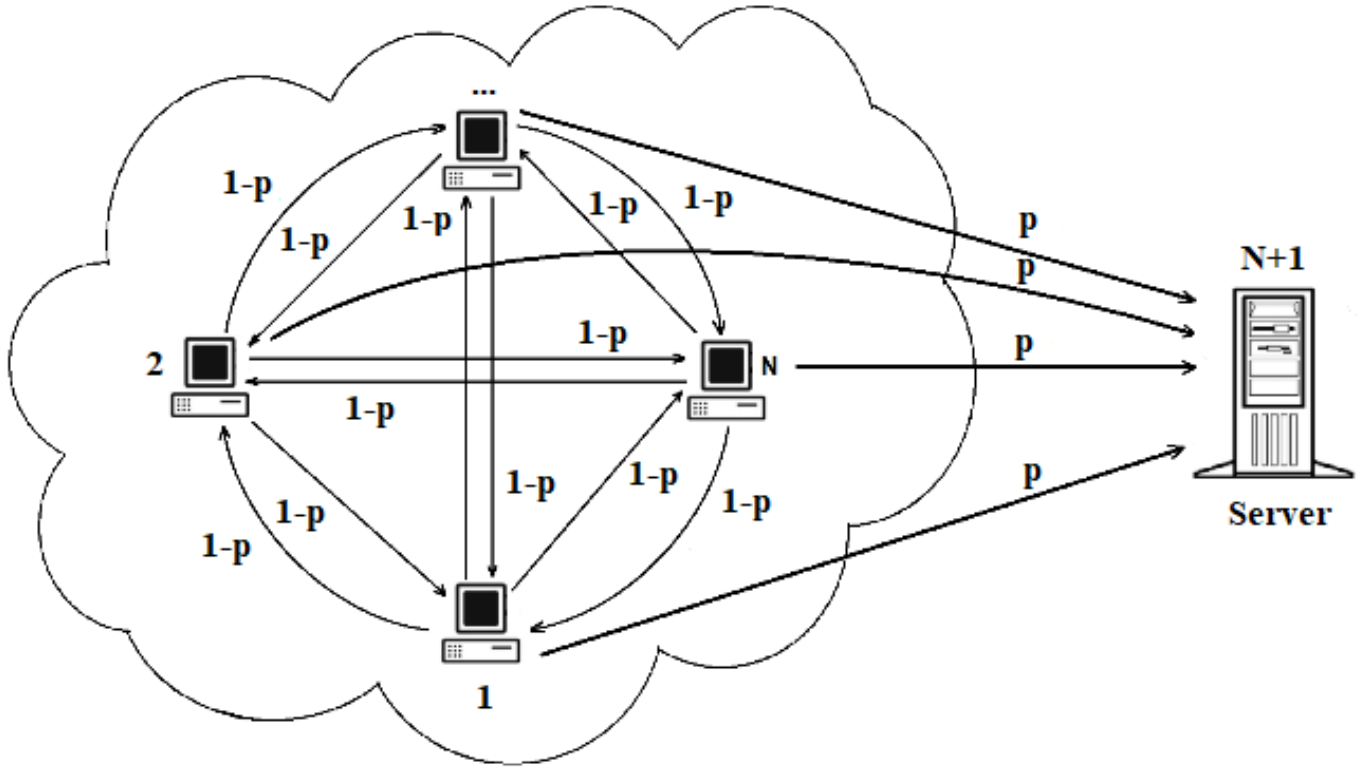
5. Подписанный бюллетень **<n, s>** отправляется на сервер голосования по **анонимному каналу связи**.

Сервер проверяет корректность бюллетеня с помощью равенства

$$SHA3(n) = s^D mod\ N\ ,$$

и, в случае корректности, учитывает голос и заносит информацию о бюллетене в открытую базу данных.

Анонимный канал связи



Сообщение гарантированно будет передано на сервер, вероятность отправки сообщения за n или меньше шагов:

$$P(l \leq n) = \sum_{i=0}^{n} p(1-p)^i$$