

ИНСТРУКЦИЯ ПО УСТАНОВКЕ И ЭКСПЛУАТАЦИИ

СОЗ «ТАЙГА»

Версия документа: 1.0 от 23.01.2026

Код: НМА-001-INSTRUCT

г. Москва, 2026 г.

СОДЕРЖАНИЕ

1. Термины и сокращения.....	4
2. Краткое описание ПО и его назначение	6
2.1. Основные функции.....	6
2.1.1. Создание записок.....	6
2.1.2. Доступ к запискам.....	6
2.1.3. Редактирование и управление.....	6
2.1.4. Особенности	6
2.1.5. Назначение	7
2.1.6. Целевая аудитория документа	7
3. Системные требования.....	8
3.1. Операционная система.....	8
3.2. Язык программирования.....	8
3.3. Зависимости.....	8
3.3.1. База данных.....	8
3.3.2. Дополнительные инструменты.....	8
3.3.3. Оборудование.....	9
4. Установка и настройка.....	10
4.1. Конфигурационные файлы	10
4.1.1. Основной конфиг: config.json.....	10
4.1.2. Параметры по умолчанию	10
4.1.3. Переменные окружения.....	11
4.2. Настройка PostgreSQL	11
4.2.1. Создание базы данных и пользователя	11
4.2.2. Проверка подключения.....	11
5. Запуск и использование	13
5.1. Основные команды запуска.....	13
5.1.1. Запуск сервера (без миграций).....	13
5.1.2. Запуск с флагами	13
5.2. Демонстрационные сценарии	13
5.2.1. Тестовый сценарий: Создание и чтение записки.....	13
Откройте главную страницу	13
Введите содержимое записки.....	14

Настройте записку	15
Создайте записку	16
Дополнительные действия.....	17
6. Специфические особенности.....	18
6.1. Миграции базы данных	18
6.2. Шифрование и ключи	18
6.3. Разграничение доступа	18
7. Системное логирование.....	19
7.1. Уровни логирования.....	19
7.2. Вывод логов.....	19
8. Устранение неполадок: типичные ошибки и их решение	20
8.1. Ошибки при запуске сервера	20
8.1.1. Ошибка: "Port is already in use" (порт занят).....	20
8.2. Ошибки при подключении к PostgreSQL	20
8.2.1. Ошибка: "could not connect to server" (не подключается к БД).....	20
8.3. Ошибки в браузере	21
8.3.1. Ошибка: "WASM modules not loaded" (не загружены WASM модули).....	21
8.4. Ошибки при работе с CAPTCHA	21
8.4.1. Ошибка: "captcha validation failed" (CAPTCHA не проходит)	21
8.5. Ошибки при работе с LDAP	22
8.5.1. Ошибка: "LDAP connection failed" (не подключается к LDAP)	22
8.6. Ошибки при работе с кэшем	22
8.6.1. Ошибка: "cache initialization failed" (кэш не инициализирован)	22
8.7. Ошибки в логировании.....	23
8.7.1. Ошибка: "logs not output" (логи не выводятся)	23
9. Поддержка	24
9.1. Контакты.....	24

1. ТЕРМИНЫ И СОКРАЩЕНИЯ

СОЗ «ТАЙГА» — Сервис обмена записками «Тайга», защищённое решение для одноразовой передачи конфиденциальных сообщений с использованием криптографических методов.

Оконечное (сквозное) шифрование (end-to-end encryption, E2EE) — окончное шифрование; метод защиты данных, при котором сообщения шифруются на устройстве отправителя и расшифровываются только на устройстве получателя. Все операции происходят в браузере без передачи незашифрованных данных на сервер.

ГОСТ 34.10-2018 — отечественный стандарт, определяющий алгоритмы формирования и проверки электронной цифровой подписи (ЭЦП).

ГОСТ 34.11-2018 — отечественный стандарт хэширования данных, также известный как «Стрибог».

ГОСТ 34.12-2018 — отечественный стандарт для симметричного блочного алгоритма шифрования "Кузнечик".

ЭЦП (Электронная цифровая подпись) — механизм подписания данных с использованием криптографических ключей, обеспечивающий аутентичность и неотказуемость информации.

Записка — одноразовая зашифрованная информация, созданная через сервис Тайга. Может содержать текст, файлы и быть защищена различными методами шифрования.

Ссылка на записку — уникальный URL с ключом расшифрования, позволяющий получить доступ к конкретной записке. После прочтения записка автоматически удаляется.

Ключ расшифрования — секретный элемент, необходимый для декодирования и расшифрования зашифрованного контента. Может быть дополнительным образом защищен паролем или сертификатом.

API (Application Programming Interface) — интерфейс прикладного программирования; набор определений и протоколов для построения и интеграции программных приложений.

AES (Advanced Encryption Standard) — симметричный алгоритм шифрования данных, принятый в качестве стандарта правительством США.

CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart) — тест, предназначенный для проверки того, что пользователь является человеком, а не ботом.

UI (User Interface) — пользовательский интерфейс; совокупность компонентов системы, взаимодействующих с человеком через браузерный веб-интерфейс.

Markdown — облегчённая разметка текста, позволяющая форматировать содержимое записки (заголовки, списки, выделение текста и т.д.).

PoW (Proof of Work) — доказательство через решение небольшой задачи.

WASM (WebAssembly) — низкоуровневый формат кода, используемый для ускорения работы в браузере через специализированные модули.

Rutoken — сертифицированное криптографическое средство защиты информации, позволяющее использовать аппаратные ключи для расшифровки записок.

FIDO (Fast Identity Online) — стандарт аутентификации без паролей через криптографические ключи, доступный в системе Тайга для дополнительной защиты ссылок.

LDAP/AD — протоколы аутентификации и авторизации, интегрируемые с системой Тайга для корпоративного использования.

POSTGRESQL — объектно-реляционная система управления базами данных, используемая в Тайга для хранения зашифрованных записок и пользовательских данных.

REDIS — нереляционная (NoSQL) система управления базами данных с оперативной памятью, применяемая как кэш для ускорения запросов и хранения CAPTCHA.

XOR (Exclusive OR) — логическая операция, используемая для побитового шифрования данных. Также применяется как дополнительный способ защиты ссылок на записки.

2. КРАТКОЕ ОПИСАНИЕ ПО И ЕГО НАЗНАЧЕНИЕ

Сервис обмена записками «Тайга» (СОЗ «Тайга») - это веб-приложение для безопасной передачи одноразовых зашифрованных записок с использованием *оконечного шифрования*. Система обеспечивает конфиденциальность данных через интерактивный (браузерный) интерфейс на стороне клиента. Все операции по созданию, передаче и чтению записок происходят без отправки незашифрованных данных на сервер.

2.1. Основные функции

2.1.1. Создание записок

- Пользовательский интерфейс позволяет вводить текст, прикреплять файлы и использовать Markdown разметку.
- Выбор метода шифрования: XOR, AES, ГОСТ 34.12-2018 (Кузнечик).
- Настройка времени жизни записки (автоматическое удаление после определённого периода).

2.1.2. Доступ к запискам

- Получение уникальной ссылки с ключом для расшифрования зашифрованного контента.
- Дополнительная защита ключа в ссылке через пароль, сертификаты (Rutoken, FIDO) или мнемонические фразы.

2.1.3. Редактирование и управление

- Возможность редактирования записки при наличии специального ключа.
- Управление сертификатами: импорт/экспорт, хранение в локальном хранилище (localStorage), доверенные сертификаты

2.1.4. Особенности

- **Оконечное шифрование:** Все данные шифруются в браузере перед отправкой, обеспечивая максимальную безопасность. Сервер является хранилищем шифротекстов.
- **Одноразовое использование:** записки могут быть прочитаны только один раз и автоматически удаляются после этого.
- **Многоязычный интерфейс:** Поддержка русского и английского языков для удобства пользователей.

2.1.5. Назначение

Приложение предназначено для безопасного обмена информацией между пользователями, где важно обеспечить защиту данных от несанкционированного доступа.

Может использоваться:

- В личных целях: хранение и передача личных записок.
- В корпоративной среде: безопасная коммуникация по юридическим или техническим документам, передача чувствительных данных: пароли, токены, базы данных.

2.1.6. Целевая аудитория документа

- **Конечные пользователи:** для создания и получения зашифрованных сообщений.
- **Администраторы/разработчики:** для настройки системы, интеграции с LDAP/AD и управления провайдерами аутентификации.

3. СИСТЕМНЫЕ ТРЕБОВАНИЯ

3.1. Операционная система

- **Linux** (Ubuntu 20.04 LTS или Debian 11):
 - Предпочтительная платформа для развертывания и тестирования.
 - Архитектура: x86_64, arm, arm64, 386.
- **Astra Linux** (1.6+):
 - Предпочтительная отечественная платформа для развертывания.
 - Архитектура: x86_64, arm, arm64, 386.
- **Red OS** (3.0+):
 - Архитектура: x86_64, arm, arm64, 386.
- **Windows Server**:
 - Поддержка возможна, но требует дополнительной конфигурации и проверки совместимости.

3.2. Язык программирования

- **Go** (Golang, версия ≥ 1.23):
 - Основной язык для backend-разработки (porcini/, pkg/).
- **JavaScript** (ES6+):
 - Для frontend компонентов и WASM модулей.

3.3. Зависимости

3.3.1. База данных

- **PostgreSQL** ≥ 13 :
 - Основное хранилище данных (записки, пользователи, сертификаты).
 - Схема taiga обязательна для создания.
 - Требуется прав на создание таблиц и триггеров.

3.3.2. Дополнительные инструменты

- Расширение для браузера **Рутокен-плагин**:
 - Только для расшифрования с помощью токенов Рутокен.

3.3.3. Оборудование

- **Процессор:**
 - Минимум 2 ядра (x86_64, arm, arm64, 386 архитектура).
 - Рекомендуется 4+ для нагрузок и параллельных операций.
- **Память ОЗУ:**
 - Минимум 4 ГБ (для тестирования).
 - Рекомендуется 8 ГБ для продакшна и высокой нагрузки.
- **Дисковое пространство:**
 - Минимум 10 ГБ свободного места.
 - Для установки и логов.
- **База данных:**
 - Минимум 10 ГБ свободного места.
 - Рекомендуемый рассчитывается индивидуально, в зависимости от нужд и количества пользователей.
- **Сеть:**
 - Поддержка TCP/IP для взаимодействия с PostgreSQL, LDAP серверами и API.

Важно!

Для полноценной работы требуется:

- 1. Доступ к PostgreSQL DB (по умолчанию на localhost)*
- 2. Запуск миграций БД через migrator после установки*
- 3. Поддержка WebAssembly в браузере для ускорения шифрования в браузере*

4. УСТАНОВКА И НАСТРОЙКА

4.1. Конфигурационные файлы

4.1.1. Основной конфиг: config.json

Создайте файл *config.json* в корне проекта со следующим содержимым:

```
{
  "db": {
    "host": "localhost",
    "port": 5432,
    "user": "taiga_user",
    "password": "secure_password",
    "dbname": "taiga_db",
    "sslmode": "disable"
  },
  "http": {
    "port": 80
  },
  "options": {
    "with_admin": true
  },
  "captcha": {
    "start_complexity": 6,
    "step_complexity": 3
  },
  "cache": {
    "host": "localhost",
    "port": 6379,
    "username": "",
    "password": "",
    "db_number": 0,
    "ssl_mode": "disable"
  }
}
```

4.1.2. Параметры по умолчанию

Если файл не указан, система использует значения по умолчанию:

Таблица 1

Параметр	Значение по умолчанию
db.host	localhost
db.port	5432
db.user	taiga_user

db.password	secure_password
db.dbname	taiga_db
http.port	8080
options.with_admin	true

4.1.3. Переменные окружения

Для переопределения конфигурации используйте переменные среды:

```
export TAIGA_DB_HOST=localhost
export TAIGA_DB_PORT=5432
export TAIGA_DB_USER=taiga_user
export TAIGA_DB_PASSWORD=secure_password
export TAIGA_HTTP_PORT=8080
export TAIGA_WITH_ADMIN=true
```

Примечание:

Переменные автоматически загружаются через `configs.SetEnv` при запуске. Имена переменных следуют шаблону `TAIGA_[PARAMETER]`.

4.2. Настройка PostgreSQL

4.2.1. Создание базы данных и пользователя

Подключитесь к PostgreSQL

```
psql -U postgres
```

Создайте пользователя и БД

```
CREATE USER taiga_user WITH PASSWORD 'secure_password';
CREATE DATABASE taiga_db OWNER taiga_user;
```

Назначьте права

```
GRANT ALL PRIVILEGES ON DATABASE taiga_db TO taiga_user;
```

4.2.2. Проверка подключения

Тестовое соединение

```
psql -h localhost -p 5432 -U taiga_user -d taiga_db
```

Важно!

Убедитесь, что PostgreSQL сервер запущен и доступен на указанном хосте/порте.

5. ЗАПУСК И ИСПОЛЬЗОВАНИЕ

5.1. Основные команды запуска

Тайга - это Go-приложение, работающее как серверное приложение, которое обрабатывает все HTTP запросы и управляет бизнес-логикой.

5.1.1. Запуск сервера (без миграций)

```
./taiga --config=config.json
```

Важно!

Обязательно указать файл конфигурации (*config.json*) для подключения к *PostgreSQL*.

5.1.2. Запуск с флагами

Таблица 2

Флаг	Назначение
-version	Показать версию приложения и сборки

5.2. Демонстрационные сценарии

5.2.1. Тестовый сценарий: Создание и чтение записки

После открытия сервиса в браузере вы окажетесь на главной странице приложения **Тайга**. Здесь вы можете создать свою первую записку, воспользовавшись удобным интерфейсом.

Откройте главную страницу

После авторизации вы попадёте на экран создания новой записки. Этот экран содержит все необходимые инструменты для шифрования, добавления файлов и выбора метода защиты данных.

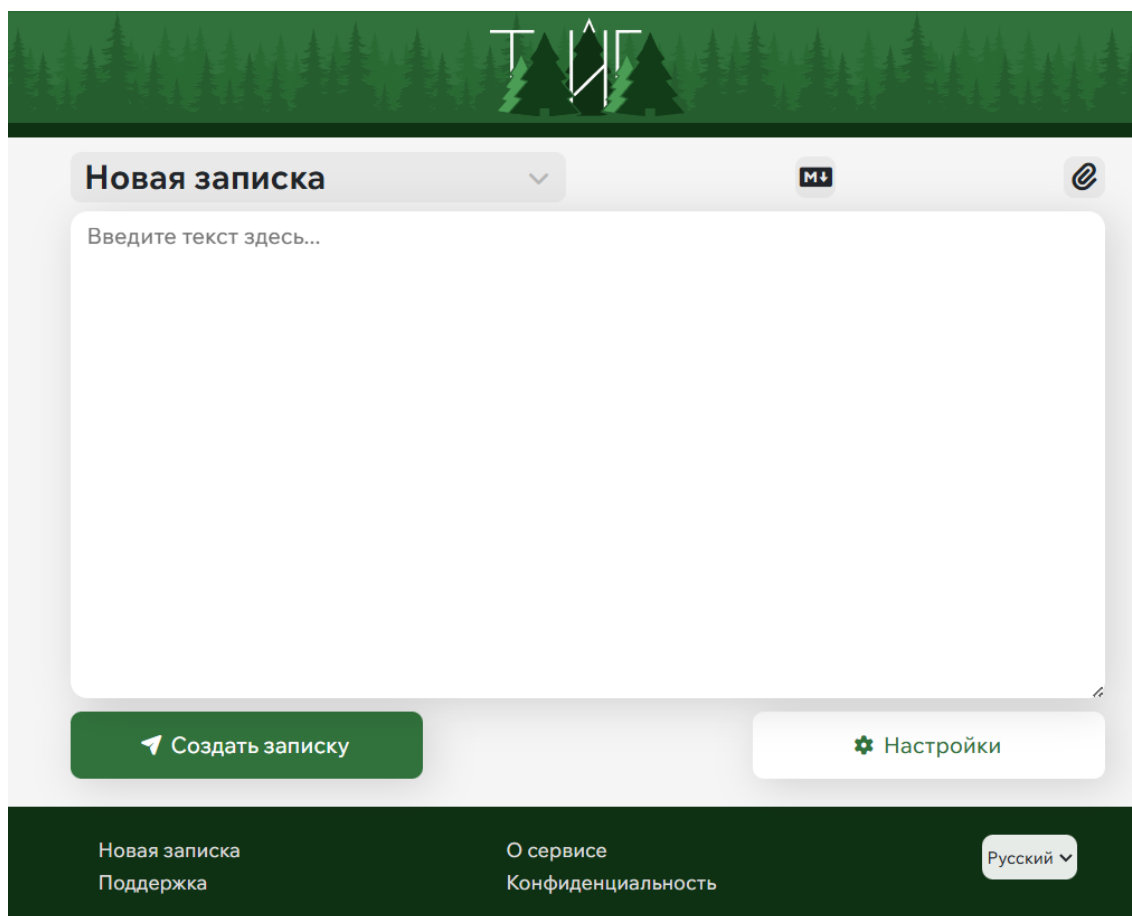


Рисунок 1 — Главная страница

Введите содержимое записки

В центральной части экрана расположен блок **ввода текста**. Здесь можно написать обычный текст или использовать Markdown для форматирования.

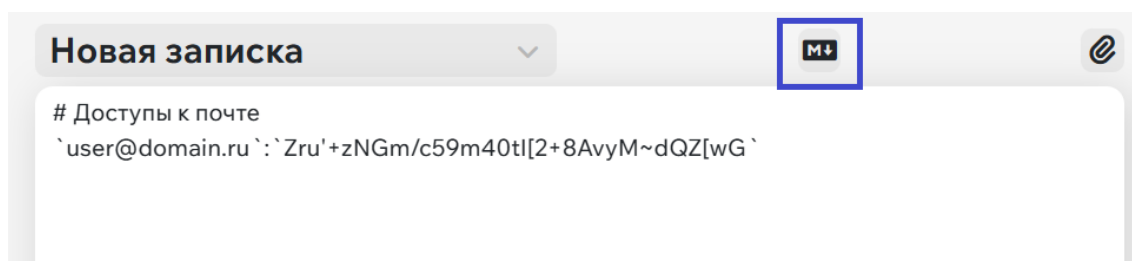


Рисунок 2 — Кнопка для включения Markdown разметки

При включенном Markdown можно включить предпросмотр нажав на иконку.

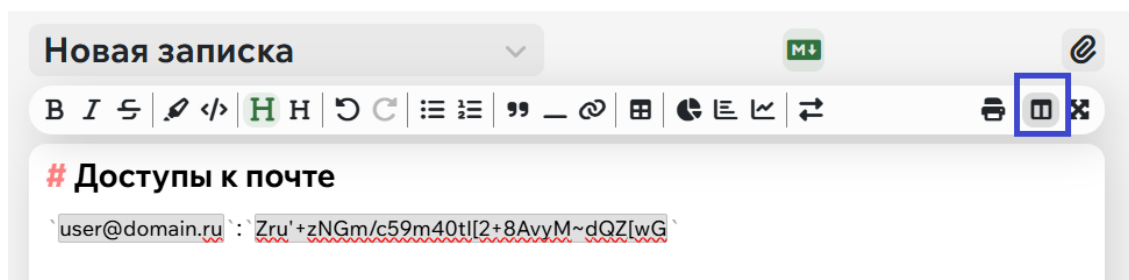


Рисунок 3 — Кнопка для включения предпросмотра Markdown разметки

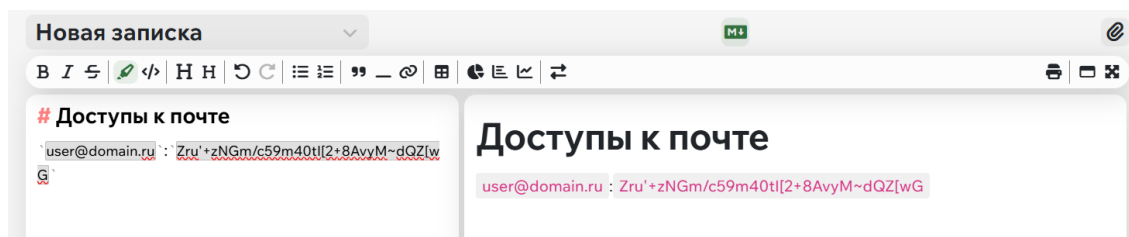


Рисунок 4 — Включенный предпросмотр

Настройте записку

Для большинства сценариев достаточно стандартных настроек, но вы можете настроить записку под себя, для этого откройте блок «**Настройки**»



Рисунок 5 — Кнопка для открытия настроек

В открывшемся поле:

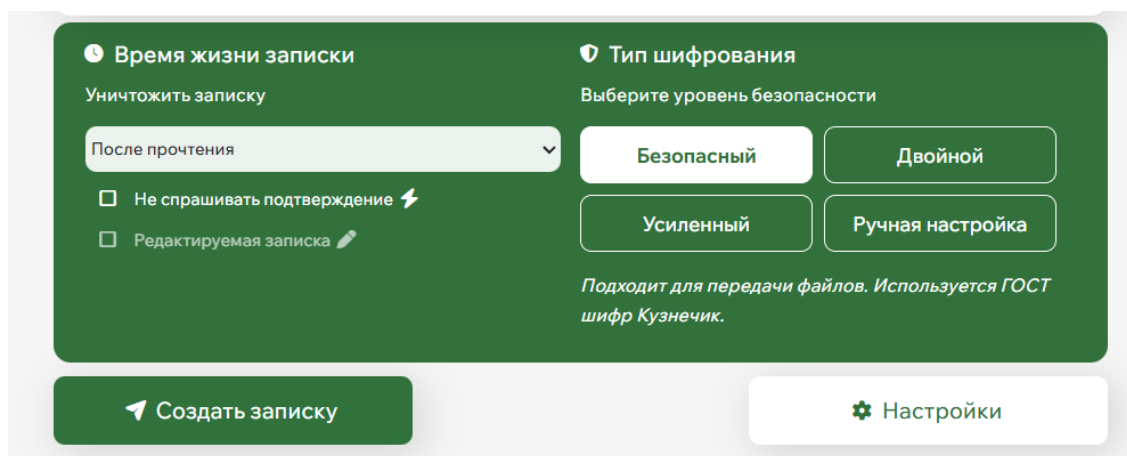


Рисунок 6 — Меню настроек записки

Время жизни записки. Выберите, как долго ваша записка будет доступна:

- **После прочтения** (по умолчанию). Удалится сразу после прочтения, одноразовая.
- **15 минут, 1 час, 7 дней** и т.д.

Метод шифрования. Здесь можно выбрать уровень защиты:

- **Простое:** автоматический выбор режима шифра Кузнечик. (рекомендуется по умолчанию).
- **Двойное / Максимальное:** более высокий уровень защиты. Сначала сообщение шифруется шифром Кузнечик, затем AES
- **Усиленный:** записка шифруется с помощью XOR, ключ больше или равен длине записки. Режим подходит только для небольших текстов.
- **Ручная настройка:** позволяет выбрать конкретные параметры шифрования.

Создайте записку

Нажмите кнопку «Создать записку».

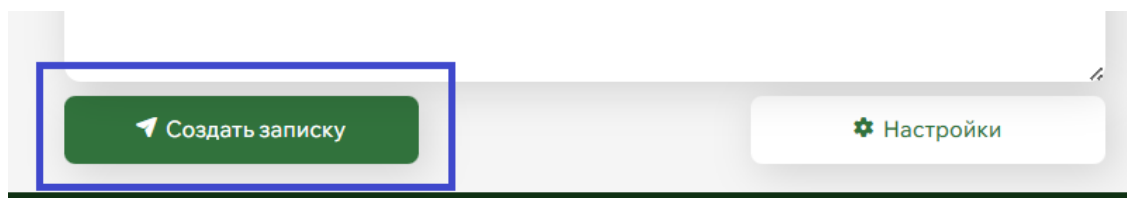


Рисунок 7 — Кнопка для создания записки

После успешного создания появится ссылка на вашу записку.

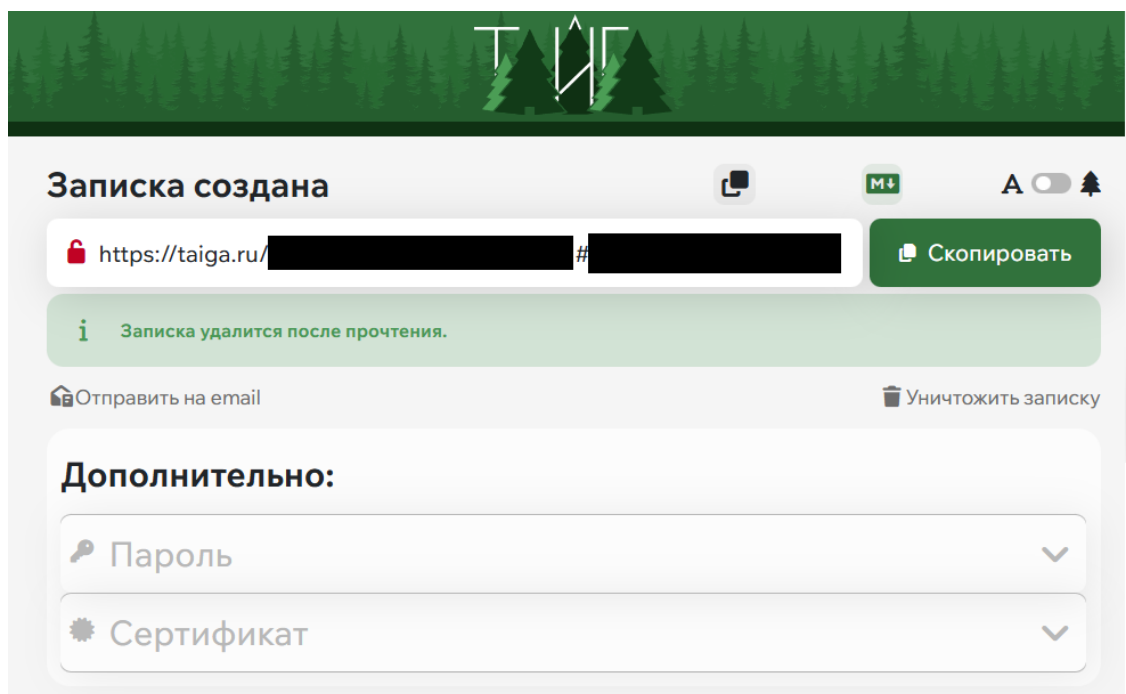


Рисунок 8 — Страница со ссылкой на записку

Важно:

Отправьте эту ссылку получателю, она не будет отображаться снова.

Дополнительные действия

После создания записки доступны дополнительные действия:

- **Дублировать записку** – возвращает на странице редактирования, с ранее заполненной запиской. Позволяет поправить текст и создать ещё одну записку.
- **Копировать ссылку** - чтобы передать её другому человеку.
- **Удалить сейчас** - немедленно удалить запись (необратимо).
- **Зашифровать дополнительно**. Если вы хотите дополнительно зашифровать ключ в ссылке – используйте:
 - Пароль
 - Импортированные или доверенные сертификаты

6. СПЕЦИФИЧЕСКИЕ ОСОБЕННОСТИ

6.1. Миграции базы данных

Миграции автоматически запускаются при запуске новой версии программы и применяются к PostgreSQL схеме taiga. Убедитесь, что БД готова к подключению.

6.2. Шифрование и ключи

Тайга использует библиотеку WellspringJS, которая в свою очередь реализует следующие стандарты:

- **Кузнечик (ГОСТ 34.12-2018)** - основное шифрование
- **Стрибог (ГОСТ 34.11-2018)** - для генерации ключей
- **ГОСТ 34.10-2018** - для цифровых подписей

Примечание:

Шифрование в браузере реализовано через WellspringJS, который интегрируется и работает только во фронтенде. Сервер не хранит и не обрабатывает открытый текст, а также не хранит ключей.

6.3. Разграничение доступа

Система поддерживает несколько режимов дополнительной защиты ссылок:

Таблица 3

Режим	Описание
password	Ключ в ссылке дополнительно шифруется с помощью исключяющего или ключа и PBKDF2 с помощью хэш-функции Стрибог (ГОСТ 34.11-2018)
certificate	Ключ в ссылке дополнительно шифруется с помощью эфемерного закрытого ключа и открытого ключа из сертификата пользователя, вырабатыва с помощью функции КЕГ (Р 1323565.1.020-2018) общий

ключ для шифрования ссылки.

Важно!

Режимы могут быть включены через UI при создании записки.

7. СИСТЕМНОЕ ЛОГИРОВАНИЕ

Логи можно записывать в файл через команду:

```
./taiga -config=config.json > /var/log/taiga.log
```

7.1. Уровни логирования

Таблица 4

Уровень	Описание
debug	Для разработки и диагностики внутренних операций.
info	Информационные сообщения о нормальной работе системы.
warn	Предупреждения, не требующие немедленного вмешательства.
error	Ошибки обработки запросов или задач.
fatal	Критические ошибки, приводящие к завершению приложения.

Важно!

По умолчанию логи выводятся в консоль.

7.2. Вывод логов

По умолчанию все логи пишутся в stdout/stderr. Поддерживается настройка вывода:

- На файловые пути (/var/log/app.log)
- Через пользовательские синки (custom sinks)

8. УСТРАНЕНИЕ НЕПОЛАДОК: ТИПИЧНЫЕ ОШИБКИ И ИХ РЕШЕНИЕ

8.1. Ошибки при запуске сервера

8.1.1. Ошибка: "Port is already in use" (порт занят)

Причина:

Порт, указанный в конфигурационном файле (config.json), уже используется другим процессом.

Решение:

1. Проверьте использование порта:

```
lsof -i :8080
```

2. Остановите конфликтующий процесс или измените порт в config.json:

```
{
  "http": {
    "port": 8081
  }
}
```

Примечание:

Сервер выдает ошибку `listen tcp :8080: bind: address already in use`. Порт по умолчанию — 8080, но может быть изменён через переменную окружения `PORT` или в конфиге.

8.2. Ошибки при подключении к PostgreSQL

8.2.1. Ошибка: "could not connect to server" (не подключается к БД)

Причина:

Неверные параметры соединения с БД, либо сервер PostgreSQL не запущен.

Решение:

1. Проверьте конфигурационный файл:

```
{
  "db": {
    "host": "localhost",
    "port": 5432,
    "user": "taiga_user",
  }
}
```

```
"password": "secure_password",  
"dbname": "taiga_db"  
}  
}
```

2. Убедитесь, что PostgreSQL работает:

```
systemctl status postgresql
```

3. Проверьте доступность БД:

```
psql -U taiga_user -d taiga_db
```

Примечание:

Ошибка может быть обернута через `errors.Wrap()` в `storage/pg/connection.go`. Убедитесь, что пользователь имеет права на базу данных.

8.3. Ошибки в браузере

8.3.1. Ошибка: "WASM modules not loaded" (не загружены WASM модули)

Причина:

WebAssembly модули для шифрования (`128.wasm`, `64.wasm`) не доступны в браузере.

Решение:

1. Обновите браузер до современного

Примечание:

Ускоренное оконечное шифрование и Markdown разметка зависят от WASM. Если модули не загружены, пользователь не сможет использовать полный функционал сервиса.

8.4. Ошибки при работе с CAPTCHA

8.4.1. Ошибка: "captcha validation failed" (CAPTCHA не проходит)

Причина:

Неверная конфигурация CAPTCHA или отсутствие библиотеки для вычисления хешей.

Решение:

1. Проверьте параметры CAPTCHA в `config.json`:

```
{  
  "captcha": {  
    "start_complexity": 6,  
  }  
}
```

```
"step_complexity": 3
}
}
```

2. Убедитесь, что время не менялось. Обновите страницу и повторите попытку отправки записки.

8.5. Ошибки при работе с LDAP

8.5.1. Ошибка: "LDAP connection failed" (не подключается к LDAP)

Причина:

LDAP-сервер недоступен или конфигурация неверна.

Решение:

1. Проверьте LDAP URL в config.json:

```
{
  "ldap": {
    "url": "ldaps://example.com",
    "username": "admin",
    "password": "secret"
  }
}
```

2. Убедитесь, что сервер доступен:

```
ldapsearch -H ldaps://example.com -b "" -s base "(objectclass=*)"
```

Примечание:

LDAP интеграция опциональна. Если не используется, можно отключить через `options.with_ldap=false`.

8.6. Ошибки при работе с кэшем

8.6.1. Ошибка: "cache initialization failed" (кэш не инициализирован)

Причина:

Конфигурационные параметры кэша некорректны.

Решение:

1. Проверьте config.json:

```
{
  "cache": {
```

```
"host": "localhost",  
"port": 6379,  
"db_number": 0  
}  
}
```

2. Убедитесь, что Redis доступен:

```
redis-cli ping
```

Примечание:

Кэш используется для ускорения запросов и сохранения CAPTCHA. Если не работает, производительность приложения снизится.

8.7. Ошибки в логировании

8.7.1. Ошибка: "logs not output" (логи не выводятся)

Причина:

Неверный уровень логирования или отсутствие файла для сохранения.

Решение:

1. Запустите с детальным логингом:

```
./taiga --config=config.json
```

2. Проверьте файл logs/taiga.log на наличие данных.

Примечание:

Уровни: info, debug. По умолчанию - info.

9. ПОДДЕРЖКА

9.1. Контакты

E-mail: support@taiga.ru

Телеграм: https://t.me/taiga_help_chat

По ним можно:

- Получить техническую помощь
- Отправить баг-репорты
- Связаться с командой разработчиков