# 📖 Полное руководство по TeleBackup v1.1

## **🤖 Настройка Telegram**

### 1. Создание приватного канала

1. **Откройте Telegram** → Нажмите на три линии → "Создать канал"
2. **Название канала**: My Backups (или любое другое)
3. **Тип канала**: ✅ Приватный (только по ссылке)
4. **Настройки** → Управление каналом → Администраторы

### 2. Создание бота

1. **Найдите @BotFather** в поиске Telegram
2. **Отправьте команды**:

/newbot

MyBackupBot # имя бота

MyBackupStorageBot # username бота (должен заканчиваться на Bot)

1. **Скопируйте токен** вида: 1234567890:ABCDEFGHIJKLMNOPQRSTUVWXYZ
2. **Зайдите в бота и нажмите /start**

### 3. Настройка прав бота

1. **В канале** → Управление каналом → Администраторы
2. **Добавить администратора** → Введите username бота (@MyBackupStorageBot)
3. **Права**: ✅ Отправка сообщений, ❌ Все остальное

Если у вас в телеграме настроено автоматическое удаление сообщений по таймеру, то отключите или настройте нужный период удаления для этого канала.

**Итог**: для настройки работы программы нужны:

1. токен
2. channel\_id (если вы используете канал публичный, то нужно просто скопировать имя канала и получить запись вроде **@avsoft\_update**, а если канал приватный, то для получения channel\_id используйте спец утилиту, которая идет в комплекте с программой, в итоге получите channel\_id типа **-1002751931152**

### ****Если токен потерян:****

* Напишите @BotFather команду /mybots
* Выберите нужного бота
* Выберите API Token → Revoke current token
* @BotFather выдаст **новый токен**

## 💡 Рекомендация:

Если не помните токен - лучше **создайте нового бота** специально для backup'ов:

1. /newbot в @BotFather
2. Имя: BackupBot
3. Username: MyBackupStorageBot
4. Используйте свежий токен

Это будет чище и безопаснее! 🔐

### 1. ****Отозвать старый токен:****

* Напишите @BotFather команду /mybots
* Выберите вашего бота
* Выберите API Token → Revoke current token
* Получите **новый токен**

### 2. ****Создать нового бота**** (рекомендую):

bash

/newbot

Bot name: BackupBot

Username: MyBackupStorageBot

## 💡 Почему новый бот лучше:

* **Чистая история** - только для backup'ов
* **Безопаснее** - нет риска, что старый токен куда-то утек
* **Проще управлять** - один бот = одна задача

‘

## ⚠️ Важные моменты:

### ****При смене токена:****

* Старый токен **перестает работать** сразу
* Новый токен **начинает работать** сразу
* Все нужно **обновить** в config.yaml

## **📦 Установка и настройка программы TeleBackup**

## Программа не требует установки, распакуйте в удобную папку. Лучше это сделать в папке вашего профиля Windows, чтобы другие пользователи компьютера не имели доступа к вашим настройкам программы.

### 2. Настройка config.yaml

## ⚙️ Конфигурационный файл

### 📁 Секция backup

backup:

sources: *# Что архивировать*

- "C:/Users/Ivan/Documents" *# Папка*

- "D:/projects/config.json" *# Файл*

- "E:/photos" *# Папка*

exclude: *# Что исключить*

- "\*.tmp" *# Временные файлы*

- "\*.log" *# Лог-файлы*

- "cache/" *# Папки кэша*

- "node\_modules/" *# Библиотеки Node.js*

compression: deflated *# deflated или stored*

smart\_backup: true *# Умное отслеживание изменений*

min\_change\_percent: 1 *# не используется*

### 🔐 Секция security

security:

encryption: aes256 *# Шифрование: aes256 или none*

password: "P@ssw0rd!2024#Backup" *# Пароль для архива*

*# Альтернативно: пароль в файле (безопаснее)*

*# password\_file: "C:/Users/Ivan/.backup\_password"*

### 📡 Секция telegram

telegram:

bot\_token: "1234567890:ABCDEFGHIJKLMNOPQRSTUVWXYZ" *# Токен от @BotFather*

channel\_id: "@my\_backup\_storage" *# ID вашего канала*

max\_file\_size: "48 MB" *# Максимальный размер файла для Telegram*

split\_chunks: true *# Разбивать большие файлы на части*

### 💾 Секция storage

storage:

local\_copy: true *# Сохранять локальную копию*

keep\_local\_days: 7 *# Хранить локальные копии (дней)*

backup\_dir: "./backups" *# Папка для бэкапов*

# **📋 Подробная справка по config.yaml для TeleBackup**

## 📁 Структура файла config.yaml

backup:

compression: deflated

exclude:

- '\*.tmp'

- '\*.log'

- Thumbs.db

min\_change\_percent: 1

smart\_backup: true

sources:

- C:\Users\User\Documents\Important

- C:\Users\User\Pictures

- D:\Work\Projects

logging:

file: backup.log

level: INFO

security:

encryption: aes256

password: 'my\_secure\_password\_123'

storage:

backup\_dir: ./backups

keep\_local\_days: 7

local\_copy: true

telegram:

bot\_token: 1234567890:ABCDEFGHIJKLMNOPQRSTUVWXYZ

channel\_id: -1001234567890

max\_file\_size: 48 MB

split\_chunks: true

advertise\_in\_telegram: true

## 🔧 Секция **backup** - Настройки резервного копирования

### compression - Тип сжатия

* **Значения**: deflated | stored
* **По умолчанию**: deflated
* **Описание**: Алгоритм сжатия ZIP архива
  + deflated - сжатие с уменьшением размера
  + stored - без сжатия (быстрее, но больше размер)

### exclude - Исключаемые шаблоны

* **Тип**: Список строк
* **По умолчанию**: ['\*.tmp', '\*.log', 'Thumbs.db']
* **Описание**: Паттерны файлов для исключения из бэкапа
* **Примеры**:
  + \*.tmp - все временные файлы
  + \*.log - все лог-файлы
  + temp/ - папка temp
  + \*.jpg - все JPEG изображения

### min\_change\_percent - Минимальный процент изменений

* **Тип**: Целое число (0-100)
* **По умолчанию**: 1
* **Описание**: Минимальный процент измененных файлов для создания бэкапа. Эта настройка не работает. Программа ее игнорирует.

### smart\_backup - Умное резервное копирование

* **Тип**: Boolean (true/false)
* **По умолчанию**: true
* **Описание**: Включение инкрементального бэкапа
  + true - бэкап только измененных файлов
  + false - полный бэкап всегда

### sources - Источники для бэкапа

* **Тип**: Список путей
* **Обязательно**: Да
* **Описание**: Файлы и папки для резервного копирования
* **Примеры**:
  + C:\Users\User\Documents - папка Documents
  + D:\Important\file.txt - конкретный файл
  + /home/user/photos - Linux путь

## 📝 Секция **logging** - Настройки логирования

### file - Файл лога

* **Тип**: Строка (путь к файлу)
* **По умолчанию**: backup.log
* **Описание**: Путь к файлу для записи логов

### level - Уровень логирования

* **Значения**: DEBUG | INFO | WARNING | ERROR | CRITICAL
* **По умолчанию**: INFO
* **Описание**: Детализация логирования
  + DEBUG - максимальная детализация
  + INFO - обычная информация
  + ERROR - только ошибки

## 🔐 Секция **security** - Настройки безопасности

### encryption - Тип шифрования

* **Значения**: aes256 | none
* **По умолчанию**: aes256
* **Описание**: Метод шифрования архивов
  + aes256 - AES-256 шифрование (рекомендуется)
  + none - без шифрования (быстрее)

### password - Пароль шифрования

* **Тип**: Строка
* **Обязательно**: Только если encryption: aes256
* **Описание**: Пароль для шифрования архивов
* **Рекомендации**: Минимум 12 символов, буквы+цифры+спецсимволы

## 💾 Секция **storage** - Настройки хранения

### backup\_dir - Директория бэкапов

* **Тип**: Строка (путь)
* **По умолчанию**: ./backups
* **Описание**: Папка для хранения локальных копий архивов

### keep\_local\_days - Хранение локальных копий

* **Тип**: Целое число
* **По умолчанию**: 7
* **Описание**: Сколько дней хранить локальные бэкапы

### local\_copy - Сохранять локальную копию

* **Тип**: Boolean (true/false)
* **По умолчанию**: true
* **Описание**: Сохранять ли архив локально после отправки в Telegram
  + true - сохранять локально
  + false - удалять после отправки

## 📱 Секция **telegram** - Настройки Telegram

### bot\_token - Токен бота

* **Тип**: Строка
* **Обязательно**: Да
* **Описание**: Токен вашего Telegram бота
* **Формат**: 1234567890:ABCDEFGHIJKLMNOPQRSTUVWXYZ
* **Как получить**: через @BotFather

### channel\_id - ID канала

* **Тип**: Число или строка
* **Обязательно**: Да
* **Описание**: ID Telegram канала/чата для отправки
* **Формат**: "@my\_backup\_storage" (для публичных каналов, в кавычках) или -1001234567890 (отрицательное для приватных каналов, без кавычек)

### max\_file\_size - Максимальный размер файла

* **Тип**: Строка
* **По умолчанию**: 48 MB
* **Описание**: Максимальный размер для отправки в Telegram. Речь идет о размере файла архива для отправки в Телеграм. Не выключайте следующую настройку («разделение на части»), если надо заливать большие архивы в Телеграм. Т.е. логика этих двух настроек такая: если архив бекапа больше 48 МБ, то он разделяется на части по 48 МБ и заливается несколькими файлами. Это для делается из-за ограничений Телеграма.
* **Доступные единицы**: B, KB, MB, GB
* **Лимит Telegram**: 2GB (но лучше до 1.5GB), но лимит для ботов – 50 МБ. Можете проверить, сделать больше, вдруг уже что-то поменялось.

### split\_chunks - Разделение на части

* **Тип**: Boolean (true/false)
* **По умолчанию**: true
* **Описание**: Автоматически разделять большие файлы. Эта настройка завязана на предыдущую, если там установлено 48 МВ, то архив бекапа будет разделяться на файлы по 48 МВ и отправляться по частям.
  + true - разделять файлы больше max\_file\_size
  + false - отправлять как есть (может не работать для больших файлов)

### advertise\_in\_telegram - Информация в сообщениях

* **Тип**: Boolean (true/false)
* **По умолчанию**: true
* **Описание**: Добавлять ли информацию о программе в сообщения
  + true - добавлять подпись
  + false - только техническая информация

## ⚠️ Возможные проблемы и решения

### Проблема: "Bot token not found"

**Решение**: Получите токен через @BotFather и добавьте в конфиг

### Проблема: "Channel ID not found"

**Решение**: Используйте вкладку "Chat ID" в GUI для получения ID

### Проблема: "No valid sources found"

**Решение**: Проверьте пути в sources, они должны существовать

### Проблема: Файлы не отправляются в Telegram

**Решение**: Убедитесь что:

1. Бот добавлен в канал как администратор
2. Бот имеет права на отправку файлов
3. Размер файла не превышает лимиты Telegram (если вручную меняли в конфиге)

## 🔄 Приоритеты настроек

1. **GUI форма** → **config.yaml** → **Значения по умолчанию**
2. При изменении в GUI и сохранении - config.yaml обновляется
3. При запуске программы - читается config.yaml

## 📊 Рекомендации по настройке

В архиве дистрибутива есть две дополнительные утилиты:

* chat\_id.exe – получение Chat ID вашего приватного канала для   
  Telegram бота;
* get\_chat\_id\_gui.exe – утилита GUI для получения Chat ID и настройки   
  config.yaml через графический интерфейс.

Эти утилиты необязательные для работы утилиты. Если знаете другие способы получения Chat ID, то можете их удалить.

## 🤖 Автоматизация

### Windows: Планировщик заданий

1. Откройте Планировщик заданий
2. Создайте задание → Триггеры → Ежедневно в 2:00
3. Действие: Запуск программы → TeleBackup.exe
4. Аргументы: --full
5. Рабочая папка: C:\path\to\TeleBackup\

**Рабочую папку нужно указать обязательно!**

### Подробно о создании Задания Windows

В Поиске Windows введите «Планировщик задач» и запустите его.

В меню Планировщика задач выберите «Создать простую задачу».

Введите имя простой задачи и нажмите «Далее»

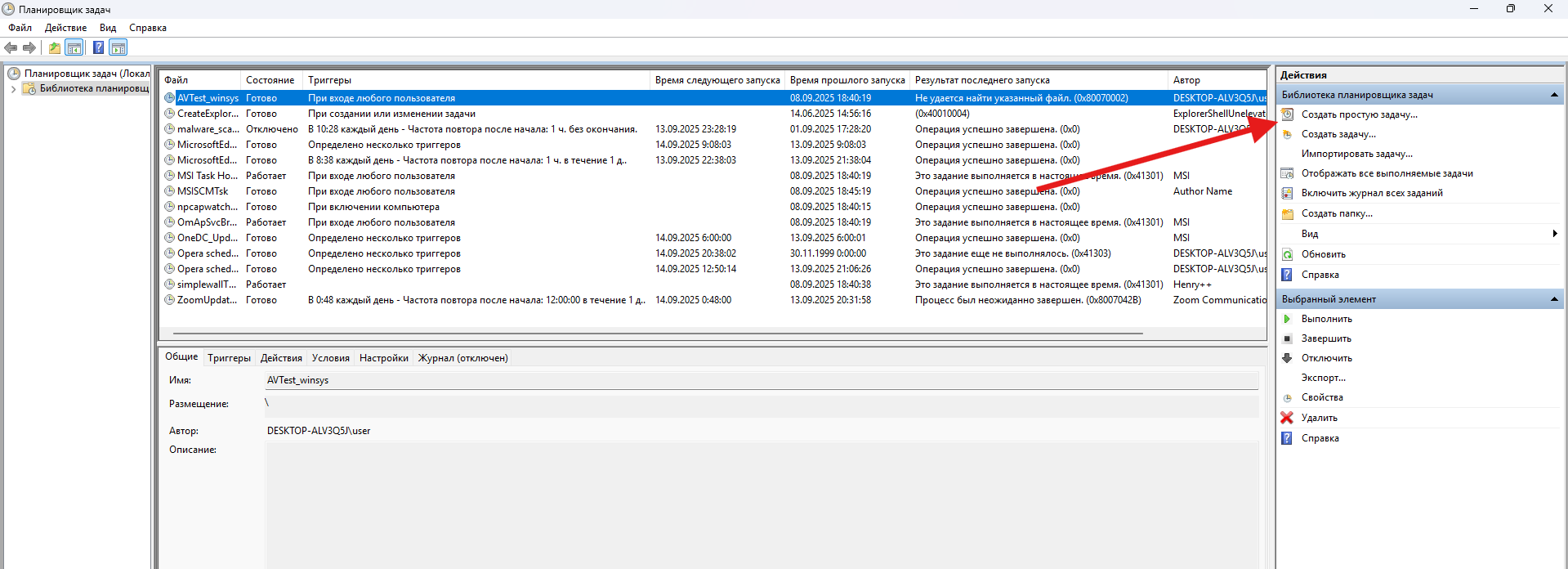
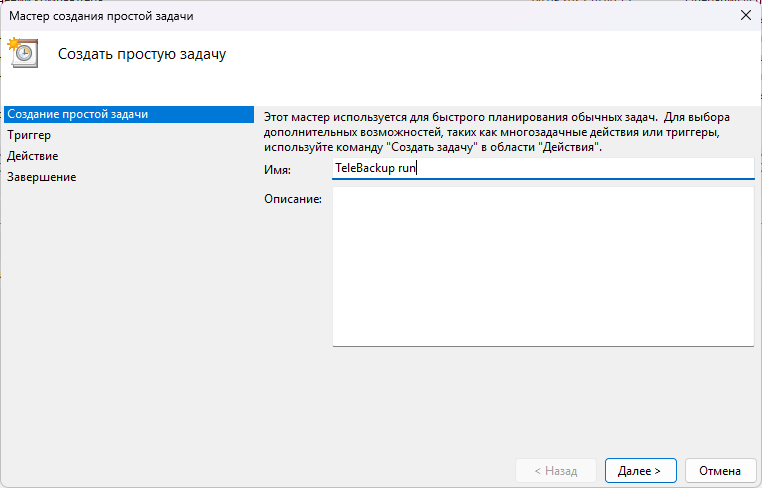
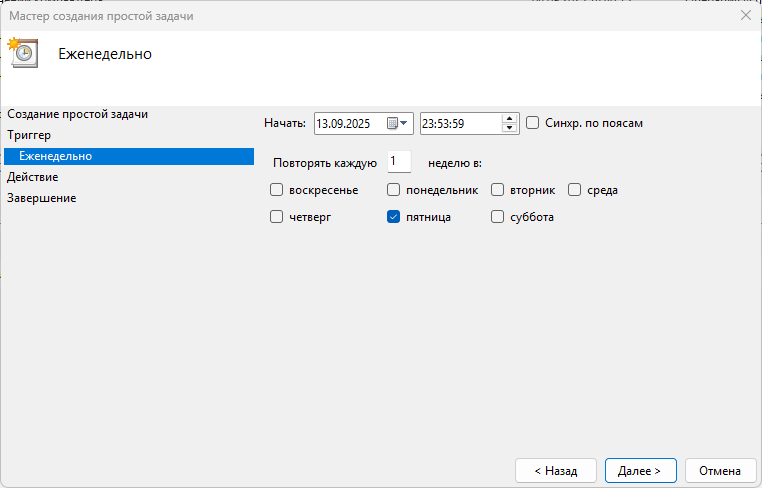
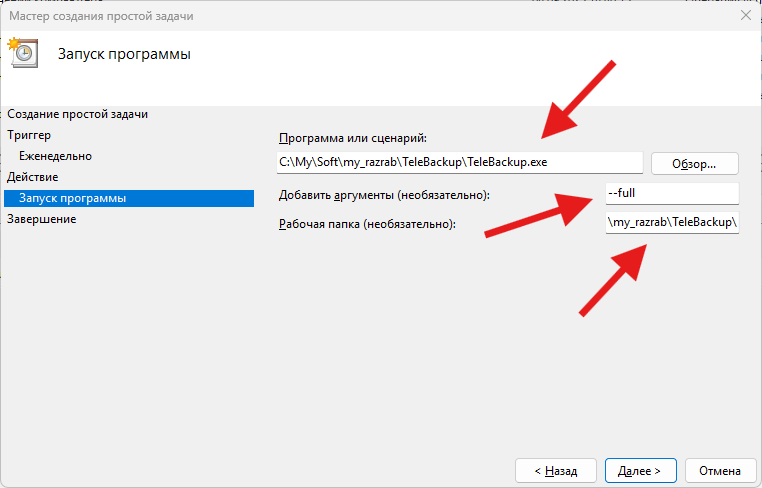
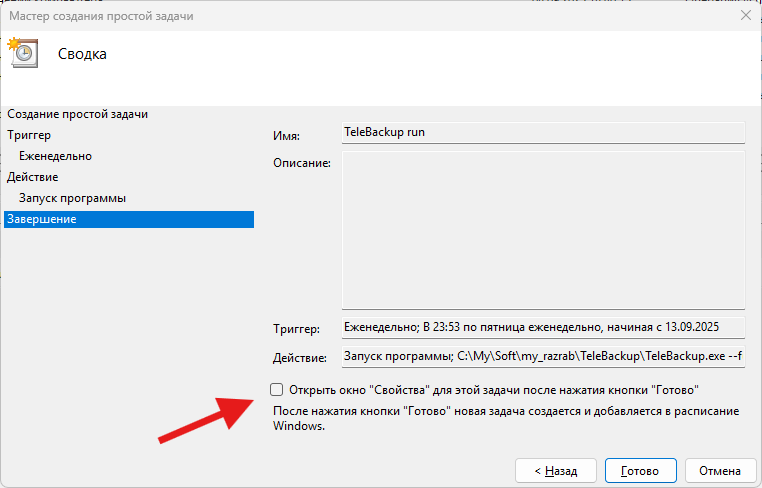
В Триггерах выберите когда производить резервное копирование. Например, Еженедельно. Укажите дни, когда запускать резервное копирование.

В Действиях укажите путь к программе. В аргументах укажите **–full**  
Рабочую папку указать обязательно (где находится файл TeleBackup.exe). Путь можно скопировать из пути запуска программы (форма чуть выше).

В окне «Завершение» можете поставить галку «Открыть окно Свойства….»

Если поставите эту галку, то обратите внимания на дополнительные настройки. Можете выбрать «Немедленно запускать задачу, если запущен плановый запуск».

После создания Задания кликните по нему правой кнопкой мыши и запустите, проверьте работу.



## 🔄 Восстановление данных

### 1. Скачивание из Telegram

* Откройте канал в Telegram
* Скачайте все части архива (если разбит)
* Объедините части: copy /b backup.zip.part\* backup.zip (если установлен WinRAR или 7zip, то объединять нет необходимости)

### 2. Распаковка архива

cmd

# Используйте 7-Zip или другой архиватор

# Пароль: из config.yaml или файла с паролем

# Или через командную строку:

7z x backup.zip -p"ваш\_пароль"

### 3. Проверка целостности

Регулярно тестируйте восстановление чтобы убедиться, что:

* Архивы не повреждены
* Пароль правильный
* Данные восстанавливаются полностью

## ⚠️ Важные советы

### Безопасность

* **НЕ делитесь** токеном бота и паролем. В конфиге эти данные хранятся в открытом виде, поэтому на компьютере необходимо соблюдать правила компьютерной безопасности.
* **Храните пароль** в менеджере паролей или в бумажной записной книжке. Необходимо записать пароль! Если он будет храниться только на устройстве-источнике бэкапов, то в случае поломки этого устройства у вас будут бэкапы (это супер!), но наверняка, вы не вспомните пароль к ним (а это уже грустно) – данные не восстановите. Вообще, если с паролями у вас беда, то лучше вообще без пароля.
* **Регулярно проверяйте** восстановление данных

### Производительность

* Исключайте временные файлы (\*.tmp, \*.log)
* Не архивируйте системные папки
* Используйте умное отслеживание изменений

### Мониторинг

* Проверяйте файл backup.log
* Настройте уведомления об ошибках
* Регулярно обновляйте зависимости

**TeleBackup** готов к использованию! 🚀 Ваши данные теперь под надежной защитой с автоматическим резервным копированием в Telegram!

## ⚠️ Коды возврата

* 0 - Успешное выполнение
* 1 - Общая ошибка
* 2 - Ошибка конфигурации
* 3 - Ошибка Telegram
* 4 - Ошибка шифрования
* 5 - Ошибка файловой системы

**Справка по параметрам**

# TeleBackup — Инструмент резервного копирования с шифрованием и отправкой в Telegram

**Описание:**  
TeleBackup создаёт резервные копии файлов и папок, шифрует их AES-256 и может отправлять в Telegram. Поддерживает "умный" режим бэкапа, когда архивируются только изменённые или новые файлы.

## Команды и аргументы

TeleBackup.exe [опции]

### Опции

| **Опция** | **Описание** |
| --- | --- |
| --config, -c | Путь к файлу конфигурации YAML. По умолчанию config.yaml. |
| --backup | Создать резервную копию без загрузки в Telegram. |
| --upload | Загрузить существующий архив в Telegram. Использовать с --file. |
| --file, -f | Путь к конкретному архиву для загрузки при --upload. |
| --full | Полный цикл: создание резервной копии и загрузка в Telegram (если нужно). |
| --cleanup | Удалить старые локальные резервные копии согласно настройкам. |
| --force | Принудительно создать и отправить архив даже если файлы не изменились. |

## Настройка конфигурации (config.yaml)

Пример конфигурации:

backup:

compression: deflated # Тип сжатия: deflated или stored

exclude:

- '\*.tmp' # Исключить временные файлы

- '\*.log' # Исключить логи

- Thumbs.db

min\_change\_percent: 1 # Минимальный процент изменения для smart\_backup

smart\_backup: true # True — сохраняем только изменённые файлы

sources:

- C:\tmp\AutoLogger.zip

- C:\Users\user\Downloads\Wub\

- C:\Users\user\Downloads\VirtualBox-7.0.0-153978-Win.exe

logging:

file: backup.log

level: INFO

security:

encryption: aes256 # Метод шифрования (только aes256 поддерживается)

password: '112233' # Пароль для шифрования

# password\_file: 'secret.txt' # Можно использовать вместо password

storage:

backup\_dir: ./backups # Папка для локальных архивов

keep\_local\_days: 7 # Хранить локальные архивы n дней

local\_copy: true # Сохранять локально архив после загрузки

telegram:

bot\_token: 123456:ABCDEF... # Токен бота

channel\_id: '@channel\_name' # Канал/чат для отправки

max\_file\_size: 48 MB # Максимальный размер одного файла

split\_chunks: true # Разделять большие файлы на части

advertise\_in\_telegram: true # Включить подпись AKSoft

## Примеры использования

### 1. Полный бэкап с загрузкой в Telegram

TeleBackup.exe --full

* Smart backup включен (smart\_backup: true): загружаются только изменённые файлы.
* Smart backup выключен (smart\_backup: false): архив создаётся полностью, Telegram получает файл только если есть изменения.

### 2. Создать резервную копию локально без Telegram

TeleBackup.exe --backup

* Использует настройки smart\_backup.
* Локальный архив создаётся в папке backup\_dir.

### 3. Загрузить существующий архив в Telegram

TeleBackup.exe --upload --file ./backups/backup\_20250912\_162800.zip

### 4. Очистка старых локальных архивов

TeleBackup.exe --cleanup

* Удаляются файлы старше keep\_local\_days.

### 5. Принудительная отправка архива даже без изменений

TeleBackup.exe --full --force

* Telegram получит файл, даже если все файлы неизменны.
* Smart backup при этом не отключается, просто принудительная загрузка.

## Как работает Smart Backup

* **smart\_backup: true**
  + Отслеживаются хеши файлов.
  + Создаётся архив только для новых и изменённых файлов.
  + Если изменений нет — Telegram не получает дубликат.
* **smart\_backup: false**
  + Создаётся полный архив.
  + Telegram получит файл **только если есть изменения** в файлах по сравнению с последним бэкапом (по хешам в backup\_state.json).
  + --force позволяет загрузить даже без изменений.

## Примечания

1. Все состояния файлов хранятся в backup\_state.json.
2. Telegram загружает файлы по частям, если они превышают max\_file\_size.
3. Архивы шифруются AES-256 с паролем из конфигурации.
4. Логи сохраняются в файл backup.log и выводятся в консоль.

Еще примеры:

**1. Файлы изменились:**

TeleBackup.exe --full

*# ✅ Создаст архив*

*# ✅ Отправит в Telegram*

**2. Файлы не изменились:**

TeleBackup.exe --full

*# ✅ Создаст архив (одинаковый)*

*# ❌ НЕ отправит в Telegram*

**3. Принудительная отправка:**

TeleBackup.exe --full --force

*# ✅ Создаст архив*

*# ✅ Отправит в Telegram (даже если identical)*

**Советы**

* Если что-то заглючило, то удалите файл **backup\_state.json** и снова запустите утилиту.
* Допустим у вас есть архив семейных видео и фото, которые лежат в определенной папке. Эта папка занимает, например, 120 ГБ. Грузить эти файлы через TeleBackup смысла нет, т. к. для ботов есть ограничения на файлы 50 МБ, поэтому TeleBackup будет разбивать этот архив с бэкапом на файлы по 49 МБ – представляете сколько файлов будет создано и загружено. Эти большие данные (которые уже есть) нужно забэкапить один раз, прямо сейчас. Поэтому сделайте это руками, т.к. при ручной загрузке лимит Телеграма уже заметно выгоднее (до 2 ГБ на один файл). Т.е. с помощью WinRAR создаете архив с этими фото и видео. Важно: выберите в WinRAR алгоритм сжатия – без сжатия (фото и видео все равно не сжимаются, а время потратите больше) и обязательно: разделение архива на файлы по 1,9 ГБ. Затем все полученные файлы архива выделите в Проводнике и перетащите в Телеграм (в ваш канал для резервного копирования). Будет долгая загрузка (если объем большой), можете поставить на ночь.
* Еще раз, будьте аккуратны с паролями. Если шифруете бэкапы паролем, то его нужно обязательно куда-то записать.

## **🤖 Что нового**

v.1.1

+ В сообщение телеграма включена полезная техническая информация (имя компьютера, имя пользователя, дата, время, размер архива, количеств файлов в архиве, алгоритм шифрования).

+ В архиве программы добавлен скрипт Install\_Backup\_Task.cmd для создания заданий Windows. Я сильно его не тестировал, поэтому после создания Задания проверьте его по настройкам. Позволяет делать скрытое Задание (без окна консоли во время работы).

AKSoft — надежные бэкапы и помощь

Telegram: https://t.me/avhelpnew

GitHub: <https://github.com/artemsoft2025/>

[artemsoft@yahoo.com](mailto:artemsoft@yahoo.com)