

Audion DevOps Tools — руководство

[English](#) · [О программе](#) · [Справочник команд](#)

Содержание

- [Быстрый старт](#)
- [Главное правило](#)
- [Риски и подтверждения](#)
- [Как читать интерфейс](#)
- [Где рабочие данные](#)
- [Разделы](#)
- [Терминал](#)
- [Проверка](#)
- [Техническая часть](#)

Как работать: запуск, главное правило, разметка риска, разделы и что читать по каждому из них.

Быстрый старт

```
launcher_gui.cmd
```

Окно поднимается от имени администратора — большинству операций нужны права: развёртывание, возможности подсистемы Linux, файл `hosts`, сетевые адаптеры, политики умолчаний и драйверов, среда восстановления, работа с дисками.

Только чтение, без запроса прав:

```
set AUDION_GUI_NO_ELEVATE=1
launcher_gui.cmd
```

Другой Python:

```
set AUDION_GUI_PYTHON=C:\Path\To\python.exe  
launcher_gui.cmd
```

Окно открывается на `http://127.0.0.1:8092/`.

Главное правило

Все операции идут средствами проекта:

- через окно;
- через `launcher_*.cmd`;
- через `runtime\python.exe system_core\cli_operation.py <операция>`.

Это особенно важно для подсистемы Linux и системных операций. Проверая проектный слой, не обходите его ручными командами — иначе проверяется не проект, а удача в текущей консоли.

Опасная операция прямо из командной строки требует явного признака:

```
runtime\python.exe system_core\cli_operation.py <операция> --yes-i-understand
```

Риски и подтверждения

Каждая операция в описании помечена родом:

род	что значит
safe	чтение, диагностика или запись только в проектную область
dangerous	меняет пользователя, систему, сеть, диски, политики или секреты

И уточняющим уровнем:

уровень	что меняется
project_write	пишет в проект
user_write	меняет текущего пользователя

уровень	что меняется
system_change	меняет политику Windows
destructive	может удалять данные или разделы
secret_export	выгружает секреты, например ключи SSH

Опасное требует отдельного подтверждения. Часть внешних мастеров дополнительно просит набрать слово вручную. Встроенные операции без вопросов должны иметь явные проектные признаки, а не скрытые запросы «да/нет».

Как читать интерфейс

Интерфейс намеренно гибридный. Названия команд и параметров остаются английскими: они совпадают с именем операции, строкой журнала, вызовами Windows и документацией Microsoft. Подсказка раскрывает их нормальным русским.

Типичная тройка:

кнопка	Default Apps Guard · Enable / repair defaults protection
подсказка	раздел Windows «Приложения по умолчанию», эталон AppAssociations.xml, политика DefaultAssociationsConfiguration
журнал	default_apps_apply_policy

Короткое имя сохраняет техническую точность, подсказка объясняет, что именно изменится.

Что означают рамки и цвета

Раскраска здесь не украшение, а приборная маркировка.

Рамка вокруг группы означает пару или жизненный цикл: резерв и восстановление, выгрузка и загрузка, запрет и снятие, слепок и применение. Обратный ход всегда на виду.

цвет	смысл
обычная синяя	вход в раздел, переход — не действие
зелёный	щадящее, только чтение, состояние; либо намеренная отмена в паре
бирюзовый	обычное рабочее изменение

цвет	смысл
янтарный	сильное вмешательство, разрушающее действие или выгрузка секретов

Цвет не заменяет род операции, подтверждение и журнал — он помогает быстро понять характер соседних кнопок внутри одной рамки.

Видимое описание команды намеренно короткое: его задача — помочь решиться за пару секунд. Полное объяснение, термины Windows, риск и порядок отката живут в подсказке. Диалог подтверждения опасного показывает полный текст.

Устойчивые пары слов

слово	значение
резерв	слепок состояния внутри проекта
выгрузка	вынести состояние наружу, в файл или папку
загрузка	занести файл обратно в систему или профиль
восстановление	применить сохранённое состояние обратно
запрет / снятие	поставить или снять ограничение
включить / выключить	режим или компонент
слепок / применение	сохранить слой и применить его обратно
очистка	уборка в выбранной области; сначала ищите проверку без удаления

Где рабочие данные

config\	описание команд, история терминала, настройки окна
system_core\	службы, окно, модули PowerShell, файлы подсистемы Linux
tools\	средства проекта и обёртки
profiles\	управляемые профили, например эталон ассоциаций
backup\	резервы системных и пользовательских состояний
output\	отчёты и выгруженное
logs\	журналы операций
report\	журналы с правами и диагностика
workspace\	рабочие папки инструментов

backup — ключевая папка наравне с входной и выходной: она заводится при подготовке, открывается отдельной кнопкой и содержит то, чем откатываются изменения. Это не кэш и не временные файлы: слепки сети, ветки реестра, выгрузки хранилища драйверов, закладки браузера до и после, сертификаты, материал SSH, состояние виртуализации и файла **hosts**.

Разделы

Каждый раздел разобран отдельным документом в **tools** — там порядок действий, особенности Windows и грабли.

раздел	о чём	подробно
Подсистема Linux	возможности и обновление, установка из сети и из файла, состояние, резерв, клонирование, перенос, регистрация дисков	tools\WSL_TOOLKIT_RU.md
Виртуализация	состояние без изменений, режим Hyper-V или сторонних машин, совместимость, переключатели с копией загрузчика	tools\VIRTUALIZATION_SWITCHER_RU.md
Сеть	диагностика, резерв и восстановление, прокси	tools\NETWORK_CONNECTIVITY_RU.md
Подключение и адаптеры	адаптеры, вход в общие папки, профили Wi-Fi, быстрые режимы	tools\NETWORK_CONNECTIVITY_RU.md
Узлы и Bitrix	подмена адреса, определение точки, состояние имён и портов, побитовый возврат	tools\BITRIX_HOSTS_RU.md
Приложения по умолчанию	слепок и сверка умолчаний, защита политикой, сравнение	tools\DEFAULT_APPS_GUARD_RU.md
Защита ассоциаций	встроенные приложения Microsoft, запрет переустановки, отслеживание смены	tools\ASSOCIATION_DEFENSE_RU.md
Железо и драйверы	запрет драйверов из обновлений, ограничения видеокарты, резерв хранилища драйверов, звук HDMI	tools\HARDWARE_DRIVER_GUARD_RU.md

раздел	о чём	подробно
Диски	опись, среда восстановления, мастер SSD	<code>tools\STORAGE_DISK_PROCEEDURES_RU.md</code>
Ключи SSH	выгрузка и загрузка ключей	<code>tools\OPENSSH_KEYKIT_RU.md</code>
Сертификаты	выгрузка и загрузка, работа с хранилищами	<code>tools\CERTIFICATE_KEYKIT_RU.md</code>
Закладки браузеров	сборка эталона, слияние, восстановление	<code>tools\BROWSER_BOOKMARKS_MASTER_RU.md</code>
Обслуживание	очистка проекта, резерв памяти помощников	<code>tools\MAINTENANCE_CLEANUP_RU.md</code>
Переезд	сборка всех доступов в одну папку с описью	<code>tools\MIGRATION_RU.md</code>

Два коротких порядка стоит знать наизусть.

Умолчания после чистой установки Windows. Проверить защиту → перезаписать эталон текущими умолчаниями → оставить удаление предложенного включённым → включить защиту → выйти из учётной записи и войти или перезагрузиться → проверить защиту снова.

Подмена адреса для Bitrix. Определить точку → посмотреть состояние имён и портов → включить подмену → поработать → выключить подмену. Выключение возвращает `hosts` побитово из копии, названной в служебном комментарии.

Терминал

Вывод показывается живьём, с разбором кодировок Windows и подсистемы Linux — без кракозябр. Итоговое состояние живёт под терминалом и держится до следующего запуска: серый — ожидание, синяя пульсация — работа, зелёный — успех, красный — ошибка. Всплывающее сообщение источником истины не считается: если окно было неактивно, его можно не увидеть.

Есть строка для ручного запуска команды — с историей и кэшем частых.

Проверка

```
runtime\python.exe -m py_compile system_core\ui_nicegui\app.py
system_core\services\devops_tools.py system_core\core\jobs.py
runtime\python.exe system_core\ui_nicegui\app.py --smoke
runtime\python.exe system_core\doctor.py
```

Что именно прогоняется перед выпуском — [список проверок](#).

Техническая часть

Запуск отдельных разделов

```
cli\launcher_wsl.cmd
cli\launcher_bitrix.cmd
cli\launcher_default_apps.cmd
cli\launcher_association_defense.cmd
cli\launcher_hardware.cmd
cli\launcher_docs_pdf.cmd
cli\launcher_codex_nuke.cmd
cli\launcher_python_nuke.cmd
```

Первые идут через тот же слой описаний и служб, что и окно. Последние два — обёртки над встроенными средствами очистки с запросом прав и подтверждением вводом.

Операции из командной строки

```
runtime\python.exe system_core\cli_operation.py <операция>
runtime\python.exe system_core\cli_operation.py <операция> --yes-i-understand
```

Полный перечень операций и их полей — [справочник команд](#). Он собирается из `config\tool_manifest.yaml` и включает подсказки, разметку риска, наследуемые поля, значения по умолчанию и варианты выбора.

Обслуживание

<code>init_folders.cmd</code>	создать недостающие рабочие папки
<code>cleanup_project.cmd</code>	деликатная очистка
<code>cleanup_project.cmd /DRYRUN /Y</code>	показать план, ничего не удаляя
<code>cleanup_project.cmd /BACKUP</code>	только папка резервов, со своим подтверждением

Очистка оставляет скрипты, настройки, документацию, лицензии и структуру папок. Удаляет порождённое и скачанное: рантайм, склад пакетов, сборки, загрузки, вложенные двоичные средства, журналы, отчёты, содержимое рабочих папок и кэши. Соседние средства не трогает.

Названия в рабочей области

Единый словарь для всех проектов Audion: **Источник**, **Добавить файл...**, **Назначение**, **Сбросить**, **Удалить**, **Список**. По-английски: **Source**, **Add file...**, **Target**, **Reset**, **Delete**, **List**.

Сбросить возвращает проектные папки и файлов не трогает. **Удалить** очищает текущие источник и назначение только после подтверждения. Слова «Цель», «Очистить», **Destination** и **Clear** для этих кнопок не используются.