

MetrosDPI

Обзор

MetrosDPI — графическое приложение для управления стратегиями обхода DPI (Deep Packet Inspection) в операционной системе Windows. Приложение предоставляет единый интерфейс для выполнения, тестирования и поддержки множественных стратегий обхода на основе фреймворка zapret.

Назначение

Приложение выполняет функции уровня управления для стратегий zapret, обеспечивая пользователям возможность:

- Выполнять и отслеживать стратегии обхода через графический интерфейс
- Тестировать эффективность стратегий на множественных целевых конечных точках
- Автоматизировать обновление стратегий из удаленных репозиториев
- Настраивать параметры интеграции на уровне системы

Системные требования

Операционная система

- Windows 10/11 (64-bit)
- Права администратора (обязательно)

Зависимости

- Python 3.8 или выше (только для разработки)
- PyQt6 >= 6.6.0
- psutil >= 5.9.0
- requests >= 2.31.0

Предварительные условия

- Директория `winws`, содержащая стратегии zapret
- Исполняемый файл `winws.exe`, расположенный в `winws/bin/`
- Сетевое подключение для функциональности обновлений

Установка

Из исходного кода

1. Клонировать репозиторий:

```
git clone https://github.com/th6c/metrosdpi.git
cd metrosdpi
```

2. Установите зависимости:

```
pip install -r requirements.txt
```

3. Убедитесь, что структура директории `winws` существует:

```
metrosdpi/
├── winws/
│   ├── bin/
│   │   └── winws.exe
│   ├── *.bat (файлы стратегий)
│   └── ...
```

4. Запустите приложение:

```
python main.pyw
```

Примечание: Приложение должно быть запущено с правами администратора.

Сборка исполняемого файла

Для создания автономного исполняемого файла:

```
pyinstaller -y -w -F --icon=app/resources/assets/icon.ico main.pyw
```

Скомпилированный исполняемый файл будет создан в директории `dist/`.

Основная функциональность

Управление стратегиями

Приложение предоставляет механизмы для:

- Выбора стратегии через выпадающее меню
- Управления жизненным циклом процесса (операции запуска/остановки)
- Автоматического перезапуска при сбоях процесса
- Мониторинга состояния стратегии через отслеживание процессов

Фреймворк тестирования

Встроенный модуль тестирования выполняет:

- Проверку подключения HTTP/1.1
- Валидацию поддержки протоколов TLS 1.2 и TLS 1.3
- Измерение сетевой задержки (ping-тесты)
- Параллельное выполнение тестов для оптимизации производительности
- Агрегацию результатов и ранжирование

Целевые объекты тестирования настраиваются через `winws/utils/targets.txt` в следующем формате:

```
name=Название цели
url=https://target.url
ping_target=target.hostname
```

Система обновлений

Механизм обновления поддерживает:

- Автоматическую проверку версий через GitHub API
- Ручную установку обновлений из локальных ZIP-архивов
- Создание резервных копий перед операциями обновления
- Применение конфигурации после обновления (инъекция флагов, удаление проверок обновлений)

Источник обновлений: [Flowseal/zapret-discord-youtube](https://github.com/Flowseal/zapret-discord-youtube)

Управление конфигурацией

Настройки приложения сохраняются в `app/config/app.json`:

```
{
  "language": "ru",
  "show_in_tray": true,
  "close_winws_on_exit": false,
  "start_minimized": true,
  "auto_start_last_strategy": true,
  "add_b_flag_on_update": true,
  "last_strategy": "strategy_name",
  "autostart_enabled": true,
  "remove_check_updates": false,
  "game_filter_enabled": false,
  "ipset_filter_mode": "loaded"
}
```

Интеграция с системой

- **Системный трей:** Поддержка фоновой работы с доступом через контекстное меню
- **Автозапуск Windows:** Интеграция с Планировщиком задач для автоматического запуска
- **Управление процессами:** Автоматический мониторинг и перезапуск процесса winws.exe
- **Управление фильтрами:** Конфигурация Game Filter и IPSet Filter

Архитектура

Структура модулей

- `main.pyw` : Точка входа, проверка прав администратора, инициализация приложения
- `main_window.py` : Главное окно и основная логика приложения
- `test_window.py` : Интерфейс тестирования стратегий
- `system_tray.py` : Интеграция с системным треем
- `config_manager.py` : Слой персистентности конфигурации
- `autostart_manager.py` : Интеграция с Планировщиком задач Windows
- `zapret_updater.py` : Реализация механизма обновления
- `translator.py` : Система интернационализации
- `path_utils.py` : Утилиты разрешения путей (совместимость с PyInstaller)

Паттерны проектирования

- Модульная архитектура с разделением ответственности
- Конфигурационно-управляемое поведение
- Обновления интерфейса на основе событий
- Мониторинг процессов через механизм опроса

Параметры конфигурации

Настройки языка

- `language` : Язык интерфейса (`ru` или `en`)

Настройки трея

- `show_in_tray` : Отображать иконку системного трея
- `start_minimized` : Запускать свернутым в трей

Поведение при выходе

- `close_winws_on_exit` : Завершать процесс winws при выходе из приложения

Настройки автозапуска

- `autostart_enabled` : Включить автозапуск Windows
- `auto_start_last_strategy` : Запускать последнюю использованную стратегию при старте
- `auto_restart_strategy` : Перезапускать стратегию при завершении процесса

Настройки обновления

- `add_b_flag_on_update` : Инжектировать флаг `/B` в стратегии при обновлении
- `remove_check_updates` : Удалять вызовы проверки обновлений из стратегий

Настройки фильтров

- `game_filter_enabled` : Включить функциональность Game Filter
- `ipset_filter_mode` : Режим фильтра IPSet (`loaded` , `any` , `none`)

Структура меню

Инструменты

- Запустить тестирование: Открыть окно тестирования стратегий
- Запустить диагностику: Выполнить системную диагностику

Настройки

- Параметры программы: Открыть диалог настроек
- Game Filter: Переключить состояние Game Filter
- IPSet Filter: Переключить состояние IPSet Filter

Обновление

- Проверить обновления zapret: Проверить наличие обновлений стратегий
- Обновить список IPSet: Загрузить последний список IPSet
- Обновить файл Hosts: Загрузить последний файл hosts
- Ручное обновление: Установить из локального ZIP-архива

Справка

- Открыть Github: Репозиторий проекта
- Открыть Github Flowseal zapret: Репозиторий стратегий

Диагностика

Модуль диагностики выполняет проверки на уровне системы:

- Статус службы Base Filtering Engine
- Проверка установки драйвера WinDivert
- Обнаружение конфликтующих служб (Killer, Intel Connectivity, Check Point, SmartByte)
- Идентификация VPN-служб
- Анализ конфигурации прокси
- Валидация ключей реестра

Соображения безопасности

- Приложение требует прав администратора для операций с сетевым драйвером
- Перехват сетевого трафика на уровне ядра
- Использование только проверенных стратегий из официальных источников
- Рекомендуется регулярное обновление стратегий для поддержания безопасности

Характеристики производительности

- Ping-тесты выполняются параллельно (до 20 одновременных операций)
- HTTP/TLS-тесты выполняются последовательно для каждой стратегии
- Обновление результатов в реальном времени во время тестирования
- Интервал мониторинга процессов: 1 секунда

Связанные проекты

- **zapret**: <https://github.com/bol-van/zapret>
- **Flowseal zapret-discord-youtube**: <https://github.com/Flowseal/zapret-discord-youtube>

Лицензия

Проект распространяется под лицензией MIT. Подробности см. в файле LICENSE.