



RuBackup

**Система резервного копирования
и восстановления данных**

REST API

ВЕРСИЯ 2.4.0, 02.07.2025

Содержание

1. Системные требования	18
2. Установка	19
2.1. Установка на выделенный хост	20
3. Настройка переменных окружения	22
3.1. Настройка на выделенном хосте	23
4. Настройка журналирования	25
4.1. Логгеры	25
4.2. Хэндлеры	25
4.3. Форматеры	25
5. Запуск и остановка сервиса	27
6. Токены и аутентификация	29
6.1. Выпуск токенов	29
6.1.1. Выпуск токенов в браузере	29
6.1.2. Выпуск токенов в командной строке	31
6.2. Обновление токенов	33
6.2.1. Обновление токенов в браузере	33
6.2.2. Обновление токенов в терминале	34
6.3. Отзыв токенов	36
6.4. Аутентификация пользователя	38
6.5. Аутентификация с Microsoft Active Directory	40
6.5.1. Выпуск <code>access_token</code> , <code>refresh_token</code> и <code>csrf_token</code> через браузер	40
6.5.2. Выпуск <code>access_token</code> , <code>refresh_token</code> и <code>csrf_token</code> в командной строке	42
7. Отправка запросов	45
7.1. Отправка запросов в браузере	45
7.2. Отправка запросов в терминале	47
8. Описание методов	48
8.1. Авторизация	48
8.1.1. Получение токенов и прохождение авторизации (POST <code>/auth/login</code>)	48
8.1.2. Деактивация имеющихся токенов (DELETE <code>/auth/logout</code>)	49
8.1.3. Обновление access токена после его истечения (POST <code>/auth/refresh</code>)	49
8.1.4. Получение списка прав пользователя (GET <code>/auth/permissions</code>)	49
8.2. Пользователи	49
8.2.1. Администраторы	49
Получение списка администраторов (GET <code>/administrators</code>)	49

Удаление списка ролей администратор (DELETE /administrators)	50
Назначение пользователю роли администратора (POST /administrators)	50
Получение информации об одном администраторе (GET /administrators/{id})	50
Удаление роли администратора. (DELETE /administrators/{id})	50
8.2.2. Супервайзеры	51
Удаление списка пользователей роли супервайзера (DELETE /supervisors)	51
Получение списка супервайзеров (GET /supervisors)	51
Назначение пользователю роли супервайзера (POST /supervisors)	51
Удаление пользователю роли супервайзера (DELETE /supervisors/{id})	51
Получение информации о супервайзере (GET /supervisors/{id})	52
8.2.3. Сопровождающие	52
Удаление списка пользователей роли сопровождающего (DELETE /maintainers)	52
Получение списка сопровождающих (GET /maintainers)	52
Назначение пользователю роли сопровождающего (POST /maintainers)	52
Удаление пользователю роли сопровождающего (DELETE /maintainers/{id})	53
Получение информации о сопровождающем (GET /maintainers/{id})	53
8.2.4. Аудиторы	53
Назначение пользователю роли аудитора (POST /auditors)	53
Получение информации об аудиторе (GET /auditors/{id})	53
Удаление аудитора (DELETE /auditors/{id})	53
Получение списка аудиторов (GET /auditors)	54
Удаление списка аудиторов (DELETE /auditors)	54
8.2.5. Пользователи	54
Удаление списка пользователей (DELETE /users)	54
Получение списка пользователей (GET /users)	55
Добавление пользователя (POST /users)	55
Изменение пароля пользователя (PATCH /users/change_password)	55
Редактирование пользователей (PATCH /users/{id})	56
Удаление пользователя (DELETE /users/{id})	56
Получение информации о пользователе (GET /users/{id})	56
Получение списка прав пользователя (GET /users/permissions)	57
8.3. Сбор сведений	57
8.3.1. Лицензии	57

Получение списка типов лицензий (GET /license_types)	57
Получение информации о типе лицензии (GET /license_types/{id})	57
8.3.2. Информация о сервере	57
Получение информации о сервере (GET /server_hello)	57
8.3.3. Сведения о состоянии приложения	57
Получение информации о состоянии приложения (GET /server_state)	57
8.3.4. Типы резервного копирования	58
Получение списка доступных типов резервного копирования (GET /backup_types)	58
8.3.5. Типы сжатия	58
Получение списка доступных типов сжатия (GET /compression_type)	58
8.3.6. Алгоритмы защиты данных	58
Получение списка доступных криптоалгоритмов (GET /crypto)	58
8.3.7. Алгоритмы дедупликации	58
Получение списка доступных алгоритмов хеш-функций (GET /deduplication_hash_algorithm)	58
8.3.8. Поиск медиасерверов	58
Поиск клиента медиасервера по HWID сервера (GET /servers_hw_id_tmp)	58
Получение информации о сервере и лицензии (GET /servers_hw_id_tmp/{id})	58
8.3.9. Доступные типы ресурсов	59
Получение списка доступных типов ресурсов (GET /resource_types)	59
Получение информации о типе ресурса (GET /resource_types/{id})	59
8.3.10. Доступные операционные системы	59
Получение списка доступных операционных систем (GET /os_types)	59
8.3.11. Статусы верификации	59
Получение списка доступных статусов верификации (GET /verification_status)	59
8.3.12. Статусы правил	59
Получение списка доступных статусов правил (GET /status_of_rule)	59
8.3.13. Ресурсы модулей	59
Получение списка доступных ресурсов модуля (POST /module_resource_list)	59
8.3.14. Метрики (статистика)	60
Получение статистики по уведомлениям (GET /metrics/notifications)	60
Получение статистики по объектам репозитория (GET /metrics/repository)	60

Получение статистики по очереди задач облаков (GET /metrics/s3_cloud_task_queue)	60
Получение статистики по очереди задач (GET /metrics/task_queue)	60
Получение статистики по очереди задач ленточных библиотек (GET /metrics/tl_task_queue)	60
8.4. Глобальная конфигурация	60
8.4.1. Редактирование параметров глобальной конфигурации (PATCH /global_configuration)	60
8.4.2. Получение параметров глобальной конфигурации (GET /global_configuration)	62
8.4.3. Включение и выключение сервисного режима (PATCH /global_configuration/service_mode)	62
8.5. Клиенты	62
8.5.1. Клиенты	62
Получение списка клиентов (GET /clients)	63
Удаление нескольких клиентов (DELETE /clients)	63
Авторизация клиентов (POST /clients/authorize)	63
Получение дерева клиентов (POST /clients/tree)	63
Редактирование существующего клиента (PATCH /clients/{id})	64
Получение клиента (GET /clients/{id})	64
Удаление клиента (DELETE /clients/{id})	65
Получение доступных типов ресурсов для создания копии (GET /clients/{id}/resource_type)	65
8.5.2. Группы клиентов	65
Получение списка групп клиентов (GET /client_groups)	65
Удаление списка групп клиентов (DELETE /client_groups)	65
Добавление групп клиентов (POST /client_groups)	66
Изменение группы клиентов (PATCH /client_groups/{id})	66
Получение информации о группе клиентов (GET /client_groups/{id})	66
Удаление группы клиентов (DELETE /client_groups/{id})	67
8.5.3. Ограничения пропускной способности клиента	67
Получение списка ограничений пропускной способности клиента (GET /client_bandwidth)	67
Удаление списка ограничений пропускной способности клиента (DELETE /client_bandwidth)	67
Создание ограничения пропускной способности клиента (POST /client_bandwidth)	67

Изменение ограничения пропускной способности клиента (PATCH /client_bandwidth/{id})	68
Получение информации об ограничении пропускной способности клиента (GET /client_bandwidth/{id})	68
Удаление ограничения пропускной способности клиента (DELETE /client_bandwidth/{id})	68
8.5.4. Неавторизованные клиенты	69
Удаление списка неавторизованных клиентов (DELETE /unauthorised_clients)	69
Получение списка неавторизованных клиентов (GET /unauthorised_clients)	69
Удаление неавторизованного клиента (DELETE /unauthorised_clients/{id})	69
Получение информации о неавторизованном клиенте (GET /unauthorised_clients/{id})	69
8.6. Хранилища	70
8.6.1. Пулы и их группы	70
Доступные типы пулов	70
Получение списка доступных типов пулов (GET /destination_storage_types)	70
Пулы	70
Удаление списка пулов (DELETE /pool_list)	70
Получение списка пулов (GET /pool_list)	70
Добавление пула (POST /pool_list)	70
Получение списка доступных для перемещения и копирования пулов (GET /pool_list/available_for_copy_or_move)	71
Редактирование пула (PATCH /pool_list/{id})	71
Удаление пула (DELETE /pool_list/{id})	72
Получение информации о пуле (GET /pool_list/{id})	72
Подмена пулов	72
Удаление списка подмены пулов (DELETE /pool_substitution)	72
Получение списка подмены пулов (GET /pool_substitution)	73
Добавление подмены пулов (POST /pool_substitution)	73
Удаление подмены пулов (DELETE /pool_substitution/{id})	73
Получение информации о подмене пулов (GET /pool_substitution/{id})	73
Динамические группы пулов	73

Получение списка динамических групп пулов (GET /dynamic_pool_groups)	73
Удаление списка динамических групп пулов (DELETE /dynamic_pool_groups)	73
Добавление динамической группы пулов (POST /dynamic_pool_groups)	74
Изменение динамической группы пулов (PATCH /dynamic_pool_groups/{id})	74
Получение информации о динамической группе пулов (GET /dynamic_pool_groups/{id})	75
Удаление динамической группы пулов (DELETE /dynamic_pool_groups/{id})	75
Пулы в динамических группах	75
Получение списка добавленных в динамическую группу пулов (GET /dynamic_pool_groups_records)	75
Удаление списка добавленных в динамическую группу пулов (DELETE /dynamic_pool_groups_records)	75
Добавление пула в динамическую группу пулов (POST /dynamic_pool_groups_records)	76
Получение информации о добавленном в динамическую группу пуле (GET /dynamic_pool_groups_records/{id})	76
Удаление добавленного в динамическую группу пула (DELETE /dynamic_pool_groups_records/{id})	76
8.6.2. Локальные файловые хранилища	76
Удаление нескольких локальных файловых хранилищ (DELETE /storage_local_catalogs)	76
Получение списка локальных файловых хранилищ (GET /storage_local_catalogs)	77
Добавление локального файлового хранилища (POST /storage_local_catalogs)	77
Редактирование существующего локального файлового хранилища (PATCH /storage_local_catalogs/{id})	77
Удаление локального файлового хранилища (DELETE /storage_local_catalogs/{id})	78
Получение информации о локальном файловом хранилище (GET /storage_local_catalogs/{id})	78
8.6.3. Блочные устройства	78
Удаление списка блочных устройств (DELETE /storage_block_devices)	78

Получение списка блочных устройств (GET /storage_block_devices).....	79
Добавление блочного устройства (POST /storage_block_devices)	79
Редактирование блочного устройства (PATCH /storage_block_devices/{id})	79
Удаление блочного устройства (DELETE /storage_block_devices/{id})...	80
Получение информации о блочном устройстве (GET /storage_block_devices/{id})	80
8.6.4. Ленточные хранилища	80
Ленточные библиотеки	80
Удаление списка ленточных библиотек (DELETE /tape_libraries)	80
Получение списка ленточных библиотек (GET /tape_libraries)	80
Добавление ленточной библиотеки (POST /tape_libraries)	81
Удаление ленточной библиотеки (DELETE /tape_libraries/{id})	81
Получение информации о ленточной библиотеке (GET /tape_libraries/{id})	81
Получение списка доступных для добавления роботов и приводов (GET /tape_libraries/candidates)	81
Получение информации о роботе и слотах (POST /tape_libraries/candidates/info)	82
Получение дерева ленточных библиотек (POST /tape_libraries/tree) ..	82
Синхронизация данных ленточной библиотеки (POST /tape_libraries/{id}/sync)	82
Типы картриджей	82
Получение списка доступных типов картриджей (GET /tape_types)	83
Картриджи	83
Удаление списка картриджей (DELETE /tape_cartridges)	83
Получение списка доступных картриджей (GET /tape_cartridges)	83
Добавление ленточного картриджа (POST /tape_cartridges)	83
Редактирование картриджа (PATCH /tape_cartridges/{id})	83
Получение информации о ленточном картридже (GET /tape_cartridges/{id})	84
Операции с ленточными приводами	84
Получение списка ленточных приводов (GET /library_tape_drives) ...	84
Получение информации о ленточном приводе (GET /library_tape_drives/{id})	84
Добавление привода (POST /library_tape_drives)	84
Удаление списка ленточных приводов (DELETE /library_tape_drives)..	85

Получение списка ленточных приводов, доступных для добавления (GET /library_tape_drives/candidates)	85
Редактирование ленточного привода (PATCH /library_tape_drives/{id})	85
Очистка ленточного привода (PATCH /library_tape_drives/{id}/clean)	86
Картриджи	86
Получение списка слотов (GET /library_slots)	86
Получение информации о слоте (GET /library_slots/{id})	86
Проверка ltfs на картридже (PATCH /library_slots/check_ltfs)	86
Стирание картриджа (PATCH /library_slots/erase_cartridge)	87
Экспорт картриджа (PATCH /library_slots/export_cartridge)	87
Импорт картриджа (PATCH /library_slots/import_cartridge)	87
Форматирование картриджа (PATCH /library_slots/format_cartridge)	88
Перемещение картриджа в другой слот (PATCH /library_slots/move_cartridge)	88
Роботы	89
Получение списка роботов (GET /medium_changers)	89
Получение информации о роботе (GET /medium_changers/{id})	89
Изменение робота (PATCH /medium_changers/{id})	89
8.6.5. Облачные хранилища	89
Облачное хранилище	89
Удаление списка облаков (DELETE /s3_clouds)	89
Получение списка доступных облаков (GET /s3_clouds)	90
Добавление нового облака (POST /s3_clouds)	90
Получение списка доступных бакетов не добавленного в СРК облака (POST /s3_clouds/buckets)	90
Удаление облака DELETE /s3_clouds/{id}	91
Получение информации о существующем облаке (GET /s3_clouds/{id})	91
Редактирование параметров облака (PATCH /s3_clouds/{id})	91
Получение списка доступных бакетов существующего облака (GET /s3_clouds/{id}/buckets)	91
Статус задач облачного хранилища	92
Получение списка доступных статусов задач на облачные хранилища (GET /s3_clouds_task_status)	92

Задачи облачного хранилища	92
Получение задач очереди облаков (GET /s3_clouds_task_queue)	92
Получение информации о задаче из очереди облаков (GET /s3_clouds_task_queue/{id})	92
Удаление списка задач из очереди облаков (DELETE /s3_clouds_task_queue)	92
Удаление ошибочных задач из очереди облаков (DELETE /s3_clouds_task_queue/errors_deletion)	92
Удаление задачи из очереди облаков (DELETE /s3_clouds_task_queue/{id})	92
8.6.6. Клиентские хранилища	93
Получение списка клиентских хранилищ (GET /client_defined_storages)	93
Добавление клиентского хранилища (POST /client_defined_storages)	93
Удаление списка клиентских хранилищ (DELETE /client_defined_storages)	93
Изменение клиентского хранилища (PATCH /client_defined_storages/{id})	93
Получение информации о клиентском хранилище (GET /client_defined_storages/{id})	94
Удаление клиентского хранилища (DELETE /client_defined_storages/{id})	94
8.6.7. Клиентские блочные устройства	94
Получение списка доступных блочных устройств указанного клиента (POST /list_client_block_devices)	94
8.6.8. Клиентские файловые системы	95
Получение списка доступных директорий указанного клиента (POST /list_client_filesystem)	95
8.6.9. Медиасерверы	95
Удаление нескольких медиасерверов (DELETE /media_servers)	95
Получение списка медиасерверов (GET /media_servers)	96
Авторизация медиасервера (POST /media_servers/authorize)	96
Получение дерева медиасерверов (GET /media_servers/tree)	96
Получение списка медиасерверов, имеющих пул типа ленточная библиотека (GET /media_servers/filtered_by_tape_library_pool)	96
Редактирование медиасервера (PATCH /media_servers/{id})	96
Удаление медиасервера (DELETE /media_servers/{id})	97
Получение информации о медиасервере (GET /media_servers/{id})	97

8.6.10. Неавторизованные медиасерверы	97
Удаление списка неавторизованных медиасерверов (DELETE /unauthorised_media_servers)	97
Получение списка неавторизованных медиасерверов (GET /unauthorised_media_servers)	98
Удаление неавторизованного медиасервера (DELETE /unauthorised_media_servers/{id})	98
Получение информации о неавторизованном медиасервере (GET /unauthorised_media_servers/{id})	98
8.6.11. Репозиторий	98
Получение списка объектов репозитория (GET /repository)	98
Удаление списка резервных копий из репозитория (DELETE /repository) ..	98
Получение списка зависимостей для нескольких цепочек (POST repository/deletion_reference_chain)	99
Получение дополнительных параметров восстановления (GET /repository/extension)	99
Получение информации об объекте репозитория (GET /repository/{id}) ..	99
Удаление резервной копии из репозитория (DELETE /repository/{id}) ...	99
Изменение периода хранения резервной копии (PATCH /repository/{id})	99
Копирование полной резервной копии (POST /repository/{id}/copying) ..	100
Перемещение полной резервной копии (POST /repository/{id}/moving) ..	100
Восстановление резервной копии (POST /repository/{id}/restoring) ..	100
Верификация резервной копии (POST /repository/{id}/verification) ...	101
8.6.12. Содержимое копии для гранулярного восстановления	101
Получение списка ресурсов, включенных в копию для гранулярного восстановления (GET /repository_record_file_list)	101
8.7. Стратегии	101
8.7.1. Стратегии резервного копирования	101
Изменение статуса стратегии (PATCH /backup_strategies)	102
Получение списка стратегий (GET /backup_strategies)	102
Удаление списка стратегий (DELETE /backup_strategies)	102
Создание стратегии (POST /backup_strategies)	102
Изменение параметров стратегии (PATCH /backup_strategies/{id}) ...	104
Получение информации о стратегии (GET /backup_strategies/{id})	106
Удалении одной стратегии (DELETE /backup_strategies/{id})	106
8.7.2. Копирование в пул по стратегиям	106

Получение списка настроенного копирования резервных копий по стратегиям в пул (GET /backup_strategies_replication).....	106
Удаление списка настроенного копирования резервных копий по стратегиям (DELETE /backup_strategies_replication).....	106
Создание правила копирования резервных копий по стратегиям в пул (POST /backup_strategies_replication)	107
Получение информации о копировании резервных копий по стратегии в пул (GET /backup_strategies_replication/{id}).....	107
Удаление настроенного копирования резервных копий по стратегиям (DELETE backup_strategies_replication/{id}).....	107
Изменение параметров правила копирования резервных копий по стратегии в пул (PATCH backup_strategies_replication/{id}).....	107
8.7.3. Правила стратегий	108
Удаление списка правил стратегии (DELETE /strategy_rules)	108
Получение списка правил стратегий (GET /strategy_rules)	108
Добавление правила в стратегию (POST /strategy_rules)	108
Получение дополнительных параметров правила стратегии (GET /strategy_rules/extension)	109
Изменение правила стратегии (PATCH /strategy_rules/{id})	110
Удаление правила стратегии (DELETE /strategy_rules/{id})	111
Получение информации о правиле стратегии (GET /strategy_rules/{id})	111
8.7.4. Администраторы стратегий	111
Получение списка администраторов для стратегии (GET /strategy_administrators).....	111
Получение информации об администраторе стратегии (GET /strategy_administrators/{id}).....	111
Назначение администраторов для стратегии (POST /strategy_administrators)	112
Удаление списка администраторов стратегии (DELETE /strategy_administrators)	112
Удаление администратора стратегии (DELETE /strategy_administrators/{id}).....	112
8.8. Глобальное расписание	113
8.8.1. Правила глобального расписания.....	113
Изменение статуса правила глобального расписания (PATCH /global_schedule)	113
Получение списка правил глобального расписания (GET	

/global_schedule)	113
Удаление списка правил глобального расписания (DELETE	
/global_schedule)	113
Добавление правила глобального расписания (POST /global_schedule) ..	114
Получение дополнительных параметров глобального расписания по id	
правила (GET /global_schedule/extension)	115
Изменение правила глобального расписания (PATCH	
/global_schedule/{id})	115
Получение информации о правиле глобального расписания (GET	
/global_schedule/{id})	117
Удаление правила глобального расписания (DELETE	
/global_schedule/{id})	117
Создание полной резервной копии по правилу глобального расписания	
(POST /global_schedule/{id}/execution)	118
8.8.2. Репликация правил глобального расписания	118
Получение списка настроенных репликаций правил глобального	
расписания (GET /global_schedule_replication)	118
Настройка репликации правил глобального расписания (POST	
/global_schedule_replication)	118
Удаление списка настроенных репликаций правил глобального	
расписания (DELETE /global_schedule_replication)	119
Получение информации о настроенной репликации правила глобального	
расписания (GET /global_schedule_replication/{id})	119
Удаление настроенной репликации правила глобального расписания	
(DELETE /global_schedule_replication/{id})	119
Изменение параметров репликации правил глобального расписания	
(PATCH /global_schedule_replication/{id})	119
8.8.3. Запросы на добавление правил глобального расписания	120
Получение списка запросов клиента на добавление правила глобального	
расписания (GET /new_rule_requests)	120
Одобрение запроса клиента на добавление правила глобального	
расписания (POST /new_rule_requests/approve)	120
Отклонение запроса клиента на добавление правила глобального	
расписания (DELETE /new_rule_requests/decline)	121
Получение информации о запросе клиента на добавление правила	
глобального расписания (GET /new_rule_requests/{id})	121
8.8.4. Запросы на удаление правил глобального расписания	121

Получение списка запросов клиента на удаление правила глобального расписания (GET <code>/delete_rule_requests</code>)	121
Одобрение запроса клиента на удаление правила глобального расписания (POST <code>/delete_rule_requests/approve</code>)	121
Отклонение запроса клиента на удаление правила глобального расписания (DELETE <code>/delete_rule_requests/decline</code>)	122
Получение информации о запросе клиента на удаление правила глобального расписания (GET <code>/delete_rule_requests/{id}</code>)	122
8.8.5. Ограничения пропускной способности	123
Получение списка ограничений пропускной способности правила глобального расписания (GET <code>/rule_bandwidth</code>)	123
Удаление списка ограничений пропускной способности правила глобального расписания (DELETE <code>/rule_bandwidth</code>)	123
Ограничение пропускной способности правила глобального расписания (POST <code>/rule_bandwidth</code>)	123
Изменение ограничения пропускной способности правила глобального расписания (PATCH <code>/rule_bandwidth/{id}</code>)	123
Удаление ограничения пропускной способности правила глобального расписания (DELETE <code>/rule_bandwidth/{id}</code>)	124
Получение информации об ограничении пропускной способности правила глобального расписания (GET <code>/rule_bandwidth/{id}</code>)	124
8.9. Задачи	125
8.9.1. Задачи	125
Удаление нескольких задач (DELETE <code>/task_queue</code>)	125
Получение очереди задач (GET <code>/task_queue</code>)	125
Создание срочной полной резервной копии (POST <code>/task_queue</code>)	125
Получение дополнительных параметров созданной задачи (GET <code>/task_queue/extension</code>)	126
Удаление задачи (DELETE <code>/task_queue/{id}</code>)	126
Получение информации о задаче из очереди (GET <code>/task_queue/{id}</code>)	127
Удаление задачи с определенным статусом (DELETE <code>/task_queue/{status}</code>)	127
8.9.2. Статусы задач	127
Получение списка возможных статусов задач (GET <code>/task_status</code>)	127
8.9.3. Перезапуск задач	127
Перезапуск задачи (POST <code>/restart_task</code>)	127
8.9.4. Приостановка задач	128

Приостановка выполнения задачи (POST <code>/put_task_on_pause</code>)	128
8.9.5. Возобновление выполнения задач	128
Возобновление приостановленной задачи (POST <code>/continue_task_working</code>)	128
8.9.6. Принудительное завершение задач	129
Убийство задачи (POST <code>/kill_task</code>)	129
8.9.7. Очередь задач ленточных библиотек	129
Получение списка задач из очереди ленточных библиотек (GET <code>/tl_task_queue</code>)	129
Получение информации о задаче из очереди ленточных библиотек (GET <code>/tl_task_queue/{id}</code>)	129
8.10. Уведомления	129
8.10.1. Статус уведомлений	129
Получение списка статусов уведомлений (GET <code>/notifications_status</code>)	129
8.10.2. Уведомления	130
Получение списка уведомлений (GET <code>/notifications</code>)	130
Получение информации об уведомлении (GET <code>/notifications/{id}</code>)	130
8.10.3. Уведомления групп	130
Удаление списка групп для уведомлений (DELETE <code>/user_groups</code>)	130
Получение списка групп для уведомлений (GET <code>/user_groups</code>)	130
Добавление группы уведомлений (POST <code>/user_groups</code>)	130
Изменение группы для уведомлений (PATCH <code>/user_groups/{id}</code>)	131
Удаление группы для уведомлений (DELETE <code>/user_groups/{id}</code>)	131
Получение информации о группе уведомлений (GET <code>/user_groups/{id}</code>)	131
8.11. Журналы	132
8.11.1. Журнал операций клиента	132
Получение журнала со списком операций клиентов (GET <code>/clients_log</code>)	132
Получение одной записи из журнала с операций клиентов (GET <code>/clients_log/{id}</code>)	132
8.11.2. Журнал глобального расписания	132
Получение списка записей в журнале глобального расписания (GET <code>/global_schedule_log</code>)	132
Получение информации о записи в журнале глобального расписания (GET <code>/global_schedule_log/{id}</code>)	132
8.11.3. Журнал операций медиасерверов	132
Получение журнала со списком операций медиасерверов (GET <code>/media_servers_log</code>)	132

Получение журнала с операцией медиасервера (GET /media_servers_log/{id})	133
8.11.4. Журнал репозитория	133
Получение списка записей в журнале репозитория (GET /repository_log)	133
Получение информации о записи в журнале репозитория (GET /repository_log/{id})	133
8.11.5. Журнал задач	133
Получение информации о записи в журнале задач (GET /task_log/{id})	133
Получение списка записей в журнале задач (GET /task_log)	134
8.11.6. Журнал операций сервера	134
Получение журнала серверных операций по задаче (GET /server_task_info)	134
8.11.7. Журнал клиентских операций	134
Получение журнала клиентских операций по задаче (GET /client_task_info)	134
8.11.8. Журнал мониторинга сервера	134
Получение списка записей в журнале мониторинга сервера (GET /rubackup_server_system_monitoring)	134
Получение информации о записи в журнале мониторинга сервера (GET /rubackup_server_system_monitoring/{id})	135
8.11.9. Журнал событий информационной безопасности	135
Удаление списка записей из журналов событий ИБ (DELETE /security_audit_log)	135
Просмотр журналов событий ИБ (GET /security_audit_log)	135
Удаление записи из журнала событий ИБ (DELETE /security_audit_log/{id})	135
Просмотр записи из журнала событий ИБ (GET /security_audit_log/{id})	135
8.12. Отчеты	136
8.12.1. Изменение статуса отчета (PATCH /reports)	136
8.12.2. Удаление списка отчетов (DELETE /reports)	136
8.12.3. Получение списка отчетов (GET /reports)	136
8.12.4. Создание отчета (POST /reports)	137
8.12.5. Изменение отчета (PATCH /reports/{id})	137
8.12.6. Получение информации об отчете (DELETE /reports/{id})	138
8.12.7. Удаление отчета (GET /reports/{id})	138

8.13. Microsoft Active Directory	138
8.13.1. Настройка LDAP	138
Удаление соединения (DELETE /ldap/configuration)	138
Настройка соединения с MS AD (POST /ldap/configuration)	138
Получение параметров настроенного соединения с MS AD (GET /ldap/configuration)	139
8.13.2. Ассоциация групп MS AD и ролей RuBackup	139
Удаление ассоциации групп MS AD и ролей RuBackup (DELETE /ldap/assosiations)	139
Получение списка ассоциаций групп MS AD и ролей RuBackup (POST /ldap/assosiations)	139
Ассоциация групп MS AD и ролей RuBackup (GET /ldap/assosiations)...	140

Система резервного копирования и восстановления данных RuBackup предоставляет пользователю возможность взаимодействия с активным сервером резервного копирования посредством HTTP-запросов к его ресурсам.

Программный интерфейс RuBackup реализован и документирован с использованием Swagger.

Настоящее руководство описывает базовые шаги установки, настройки и эксплуатации RuBackup REST API. Руководство предназначено для системных администраторов, отвечающих за сопровождение СРК.

Глава 1. Системные требования

Перед инсталляцией RuBackup API необходимо убедиться, что выполнены все действия для установки ЦПК RuBackup согласно документации «Руководстве по установке серверов резервного копирования и Linux клиентов»:

- скачаны все необходимые пакеты актуальной версии ЦПК: rubackup-common, rubackup-client, rubackup-server и rubackup-rest-api;
- пакеты rubackup-common, rubackup-client, rubackup-server установлены;
- проведена настройка основного сервера с помощью интерактивной утилиты `rb_init`.

Подробнее о вариантах установки REST API см. [Глава 2](#).

Также проверьте наличие файлов-сертификатов RuBackup, которые используются программным интерфейсом для создания защищённого соединения.

Расположение сертификатов в файловой системе:

- `/opt/rubackup/keys/server/serverCert.crt`
- `/opt/rubackup/keys/server/serverKey.key`
- `/opt/rubackup/keys/client/clientCert.crt`
- `/opt/rubackup/keys/client/clientKey.key`
- `/opt/rubackup/keys/rootCA/serverRootCACert.crt`

Глава 2. Установка



Перед использованием `rubackup-rest-api` рекомендуется внести изменения в конфигурационный файл `postgresql.conf` для увеличения количества зарезервированных подключений суперпользователя, например, командой:

```
nano /etc/postgresql/<номер версии>/main/postgresql.conf
```

где `<номер версии>` — это номер версии PostgreSQL.

Необходимо выставить значение:

```
superuser_reserved_connections = 50
```

Дистрибутив RuBackup REST API поставляется в виде deb и rpm-пакетов. Для разных дистрибутивов Linux, по причине их отличий друг от друга, предусмотрены специально подготовленные пакеты RuBackup.

В зависимости от типа используемого пакетного менеджера в вашем дистрибутиве Linux, процедура установки и удаления пакетов может использовать команды `dpkg`, `rpm`, `apt`, `yum` и пр. В настоящем руководстве процедуры установки описаны для пакетного менеджера, который оперирует пакетами deb. Например, команда установки пакета в операционной системе Ubuntu 20.04 выглядит следующим образом:

```
sudo dpkg -i rubackup-rest-api.deb
```

Для установки API в ОС с пакетным менеджером, который оперирует rpm-пакетами, вместо вышеуказанной команды следует выполнить команду:

```
sudo rpm -i rubackup-rest-api.rpm
```

Процедуры удаления пакетов в настоящем руководстве описаны для пакетного менеджера, который оперирует пакетами deb. Например, процедура удаления пакета RuBackup API выглядит следующим образом:

```
sudo apt remove rubackup-rest-api
```

Для удаления RuBackup API в операционной системе с пакетным менеджером,

который оперирует rpm пакетами, вместо вышеуказанной команды следует выполнить:

```
sudo yum remove rubackup-rest-api
```

либо:

```
sudo rpm -e rubackup-rest-api
```

Некоторые операционные системы, например Alt Linux, используют пакетную систему rpm, но вместо `yum` используют `apt`. Перед установкой или удалением пакета RuBackup REST API следует уточнить, какие команды необходимо использовать для вашего дистрибутива Linux.

Для инсталляции RuBackup API установите пакет `rubackup-rest-api` командой:

```
sudo dpkg -i rubackup-rest-api.deb
```

Имя файла пакета может отличаться в зависимости от сборки.

После установки пакета вы можете сразу запустить процесс RuBackup API, если у вас уже определён FQDN для хоста и он явно указан в файле `/etc/hosts`.

2.1. Установка на выделенный хост

Перед инсталляцией RuBackup API убедитесь, что выполнены все действия для установки СРК RuBackup согласно документу «Руководство по установке серверов резервного копирования и Linux-клиентов»:

- скачаны все необходимые пакеты актуальной версии `rubackup-common`, `rubackup-client`, `rubackup-server` и `rubackup-rest-api`;
- пакеты `rubackup-common`, `rubackup-client`, `rubackup-server` установлены. Проводить настройку с помощью утилиты `rb_init` не нужно;
- существует хост с установленным, настроенным и запущенным основным сервером `rubackup`;
- существует хост с базой данных `rubackup`.

Для инсталляции RuBackup API установите пакет `rubackup-rest-api`, например, командой:

```
sudo dpkg -i rubackup-rest-api.deb
```



Имя файла пакета может отличаться в зависимости от сборки.

Глава 3. Настройка переменных окружения

Настройка RuBackup API осуществляется пользователем при помощи изменения переменных окружения.



Для RuBackup API данные из конфигурационного файла RuBackup (/opt/rubackup/etc/config.file) не используются.

Ниже представлен перечень переменных окружения доступных для изменения из файла /opt/rubackup/etc/rubackup_api.env:

Таблица 1. Переменные окружения

Имя переменной	Описание	Возможные значения
APP_HOST	Желаемый IP-адрес или FQDN, который будет использоваться как часть адреса сервера API. Если IP-адрес или FQDN указан некорректно, то при запуске RuBackup API не будут записываться access_token и refresh_token в cookies.	IP/FQDN (localhost)
APP_PORT	Желаемый порт, который будет использоваться как часть адреса сервера API	Порт (5656)
DB_HOST	IP или FQDN сервера PostgreSQL с базой данных RuBackup	IP/FQDN (localhost)
DB_PORT	Порт сервера PostgreSQL с базой данных RuBackup	Порт (5432)
RB_SERVER_HOST	IP или FQDN основного сервера RuBackup	IP/FQDN (localhost)
DEBUG	Режим расширенного логирования	true, false

Указанные переменные могут быть применены локально с помощью export:

```
export APP_HOST=api.rubackup.local
```

Пользователь может зафиксировать значения переменных глобально, описав их в файле /opt/rubackup/etc/rubackup_api.env.

Пример 1. Задание переменных окружения в rubackup_api.env

```
APP_HOST=localhost
APP_PORT=5656
DB_HOST=localhost
DB_PORT=5432
RB_SERVER_HOST=localhost
```

```
DEBUG=False
```

Запуск Swagger и Tuscany будет произведён по адресу, указанному в параметре `APP_HOST`.

Чтобы запуск был произведен по доменному имени, достаточно указать его:

```
APP_HOST=api.rubakup.local
```

Тот же FQDN должен быть указан `/etc/hosts`.

Для запуска через `localhost` можно оставить параметры по умолчанию:

```
APP_HOST=localhost
```

Для запуска через IP-адрес нужно указать IP-адрес хоста:

```
APP_HOST=10.177.33.100
```

3.1. Настройка на выделенном хосте



Для RuBackup API из конфигурационного файла RuBackup (`/opt/rubakup/etc/config.file`) данные не используются.

Ниже представлен перечень переменных окружения доступных для изменения из файла `/opt/rubakup/etc/rubakup_api.env`:

Таблица 2. Переменные окружения

Имя переменной	Описание	Возможные значения
APP_HOST	Желаемый IP-адрес или FQDN, который будет использоваться как часть адреса сервера API. Если IP-адрес или FQDN указан некорректно, то при запуске RuBackup API не будут записываться <code>access_token</code> и <code>refresh_token</code> в cookies.	IP/FQDN (<code>localhost</code>)

Имя переменной	Описание	Возможные значения
APP_PORT	Желаемый порт, который будет использоваться как часть адреса сервера API	Порт (5656)
DB_HOST	IP или FQDN сервера PostgreSQL с базой данных RuBackup	IP/FQDN (localhost)
DB_PORT	Порт сервера PostgreSQL с базой данных RuBackup	Порт (5432)
RB_SERVER_HOST	IP или FQDN основного сервера RuBackup	IP/FQDN (localhost)
DEBUG	Режим расширенного логирования	true, false

Указанные переменные могут быть применены локально с помощью `export`:

```
export APP_HOST=api.rubakup.local
```

У пользователя есть возможность зафиксировать значения описанных переменных, описав их в файле `.bashrc`.

Пример 2. Файл `.bashrc`

```
# RuBackup API Settings
export APP_HOST=192.168.10.11
export APP_PORT=5655
export DB_HOST=localhost
export DB_PORT=5432
```

После этого необходимо перезагрузить переменные окружения:

```
. ~/.bashrc
```

Для запуска Swagger и Tuscany на выделенном хосте в файл переменных окружения `/opt/rubakup/etc/rubakup_api.env` нужно установить следующие параметры:

```
APP_HOST= # IP или FQDN хоста, на котором установлен и будет запущен rest-api
APP_PORT=5656
DB_HOST= # IP или FQDN хоста с базой данных
DB_PORT=5432
RB_SERVER_HOST= # IP или FQDN хоста основного сервера rubakup
```

Глава 4. Настройка журналирования

Для обеспечения гибкости процесса журналирования действий сервера предусмотрен специальный конфигурационный файл, расположенный по пути `/opt/rubackup/etc/rubackup_api_logger.conf`. Лог-файлы располагаются в директории по пути `/opt/rubackup/logs/rubackup-api`.

4.1. Логгеры

В конфигурационном файле присутствует четыре типа логгеров:

Таблица 3. Типы логгеров

logger_root	логгер, от которого должны наследоваться все остальные. По умолчанию записывает сообщения уровня Info и выше
logger_rb_api	логирует общие ошибки и информационные сообщения по API. По умолчанию записывает сообщения уровня Info и выше. Ведёт запись в консоль и в файл
logger_rb_access	логирует инциденты, связанные с авторизацией. По умолчанию записывает сообщения уровня Info и выше. Ведёт запись в консоль и в файл
logger_tornado	логирует ошибки web-сервера Tornado. По умолчанию записывает сообщения уровня Info и выше. Ведёт запись в консоль и в файл

4.2. Хэндлеры

Для каждого логгера в конфигурационном файле присутствуют хэндлеры, в которых написано, как и какие ошибки обрабатывать и куда их записывать.

Всего есть три хэндлера:

Таблица 4. Типы логгеров

handler_file	перехватывает сообщения уровня Info, записывает их в <code>/opt/rubackup/log/rubackup_api/access/rubackup-api.log</code> , применяет определённое форматирование. Также после 00:00 создаёт новый файл для записи логов. Количество файлов журнала не превышает 15
handler_console	перехватывает сообщения уровня Info, применяет определённое форматирование и выводит их в консоль
handler_access	перехватывает сообщения уровня Info, применяет определённое форматирование, записывает инциденты, связанные с доступом, в <code>/opt/rubackup/log/rubackup_api/access/rubackup-api.access.log</code> . Также после 00:00 создаёт новый файл для записи логов. Количество файлов журнала не превышает 15

4.3. Форматеры

Описывают формат, в котором нужно записывать и/или отображать сообщения в

файле и/или консоли.

Дополнительную информацию по журналированию можно найти по ссылке <https://docs.python.org/3/library/logging.config.html#configuration-file-format>

Глава 5. Запуск и остановка сервиса

После установки пакета и настройки переменных окружения можно производить запуск RuBackup API.

Сервер RuBackup API представляет собой фоновое приложение (сервис, демон).

Расположение

```
/opt/rubackup/bin/rubackup_api
```

Запуск в терминальном режиме

```
rubackup_api --start
```

Остановка

```
rubackup_api --stop
```

Перезагрузка

```
rubackup_api --restart
```

Для штатной эксплуатации RuBackup API рекомендуется запустить его как сервис. Для этого выполните следующие действия:

Включите сервис RuBackup API:

```
sudo systemctl enable \  
/opt/rubackup/etc/systemd/system/rubackup_api.service
```

Перезагрузите systemctl:

```
sudo systemctl daemon-reload
```

Запустите сервис rubackup_api:

```
sudo systemctl start rubackup_api.service
```

Уточнить статус RuBackup API можно при помощи команды:

```
systemctl status rubackup_api.service
```

```
rubackup_api.service - RuBackup API
Loaded: loaded (/etc/systemd/system/rubackup_api.service; enabled; vendor
preset: enabled)
Active: active (running) since Tue 2024-05-07 22:06:24 MSK; 24min ago
Main PID: 69213 (rubackup_api)
Tasks: 2 (limit: 9430)
Memory: 61.0M
CGroup: /system.slice/rubackup_api.service
   69213 /bin/sh /opt/rubackup/bin/rubackup_api --start
   69214 /opt/rubackup/lib/rubackup_rest_api_lib/rubackup_api.bin --start
мая 07 22:06:24 rb-primary systemd[1]: Started RuBackup API.
мая 07 22:06:25 rb-primary rubackup_api[69214]: RuBackup API Logger
initializing
мая 07 22:06:26 rb-primary rubackup_api[69214]: 2024-05-07 22:06:26,066 -
[WARNING] - 'The rubackup database has not been initialized. Please
authenticate'
мая 07 22:06:26 rb-primary rubackup_api[69214]: 2024-05-07 22:06:26,070 -
[INFO] - 'RuBackup REST API is running on
[https://rubackup.api.local:5656/[https://rubackup.api.local:5656]]'
```



Сообщение 'The rubackup database has not been initialized. Please authenticate' является предупреждением пользователя о необходимости пройти аутентификацию хотя бы один раз для продолжения работы с сервисом. Для прохождения аутентификации воспользуйтесь [методом POST /auth/login](#) напрямую или Tucana.

Глава 6. Токены и аутентификация

6.1. Выпуск токенов

Перед тем как пользователь сможет обратиться к ресурсам сервера RuBackup, он должен пройти аутентификацию и получить токены доступа: `access_token`, `refresh_token` и `csrf_token`.

Сгенерированный `access_token` будет действовать в течение 15 минут с момента получения, `refresh_token` — 24 часа с момента получения, `csrf_token` действует до перезагрузки сервиса `rubackup_api`.

По истечении срока жизни `access_token` его можно перевыпустить с помощью `refresh_token`. Если истек срок жизни `refresh_token`, необходимо перевыпустить новую пару токенов с помощью логина и пароля. После перезапуска `rubackup_api` также необходимо перевыпустить новую пару токенов с помощью логина и пароля.

6.1.1. Выпуск токенов в браузере

1. Перейдите по адресу `https://<app_host>:<app_port>/api/v1/` (Рисунок 1):

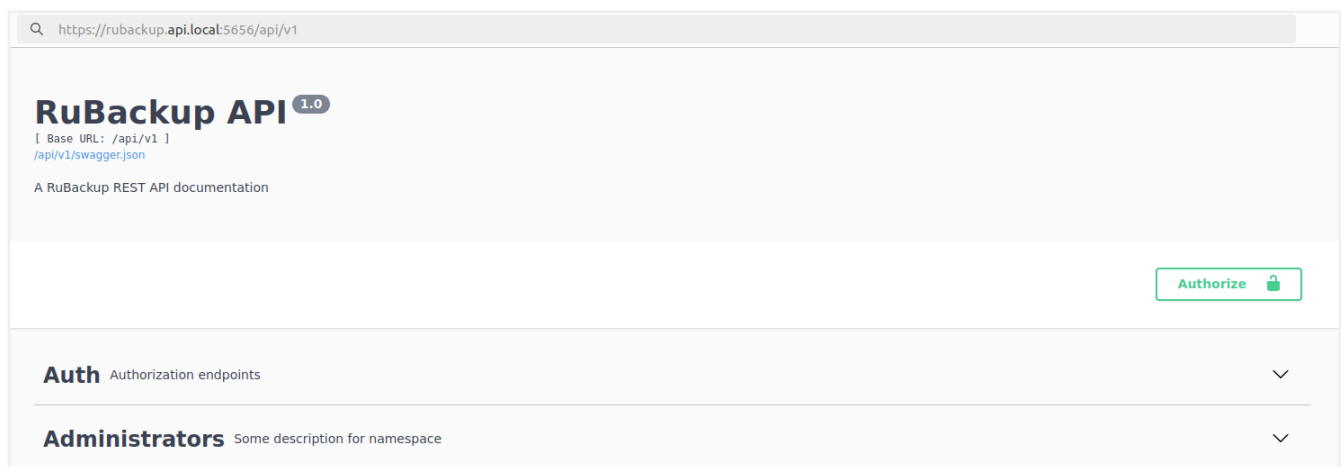


Рисунок 1. Внешний вид Swagger

1. Перейдите на вкладку «Auth» и выберите эндпоинт `/auth/login`.
2. Нажмите кнопку «Try it out», заполните payload актуальными данными и нажмите «Execute» (Рисунок 2):

Auth Authorization endpoints

POST /auth/login

Endpoint description

Possible authentication types:

- domain;
- database.

Parameters

payload object (body)

```
{
  "rubackup_database": "rubackup",
  "rubackup_user": "rubackup",
  "password": "Testing12345",
  "auth_type": "database",
  "domain": "rubackup"
}
```

Parameter content type: application/json

Execute

Рисунок 2. Заполнение payload авторизационными данными

В результате проделанных операций будут получены `access_token`, `refresh_token` и `csrf_token`, а также сопутствующая информация о пользователе прошедшем авторизацию (Рисунок 3). Также сервис автоматически разместит `access_token` и `refresh_token` в cookie-файлах.

Request URL

https://10.10.10.10:5656/api/v2/auth/login

Server response

Code **Details**

200

Response body

```
{
  "data": {
    "access token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJmcmVzaCI6ZmFsc2UsImhhdCI6MTczMzkyNDYyMiwianRpIjoiaW90MzY4LTkwYzYtMzQwZGI0ZDQ5Y2IiwiidHlwZSI6ImFjY2VzcyIsInN1YmFja3VwIiwibmJmIjozNzU0MTI0MDIyLCJpc3MiOiMTQ4ZmYxOTgtYzY3ZS00MDk4LWE3MDItYmV0eXNzZXhwIjozNzU0MTI0MDIyLCJmYW1pbHkiOiJpbGxldHJxcGRjcXZwemtnIn0. uBKkGzb9JMLTyN9M92N3bAct8Fbnn8fmm10KVw3Ams",
    "csrf token": "140ff198-c67e-4098-a702-beb8a14c7df4",
    "last fail login": null,
    "last success login": "2024-12-11T16:12:50.649160",
    "package version": "2",
    "refresh token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJmcmVzaCI6ZmFsc2UsImhhdCI6MTczMzkyNDYyMiwianRpIjoiaW90MzY4LTkwYzYtMzQwZGI0ZDQ5Y2IiwiidHlwZSI6ImFjY2VzcyIsInN1YmFja3VwIiwibmJmIjozNzU0MTI0MDIyLCJpc3MiOiMTQ4ZmYxOTgtYzY3ZS00MDk4LWE3MDItYmV0eXNzZXhwIjozNzU0MTI0MDIyLCJmYW1pbHkiOiJpbGxldHJxcGRjcXZwemtnIn0. uBKkGzb9JMLTyN9M92N3bAct8Fbnn8fmm10KVw3Ams",
    "role": "superuser",
    "rubackup_server_address": "10.10.10.10",
    "user_name": "rubackup"
  },
  "errors": {},
  "is error": false,
  "message": ""
}
```

Response headers

```
access-control-allow-credentials: true
access-control-allow-origin: https://10.10.10.10:5656
content-length: 1174
content-type: application/json
server: TornadoServer/6.2
vary: Origin
```

Рисунок 3. Ответ сервера с выпущенными токенами



Об авторизации через `auth_type: domain` — в Раздел 6.5.

6.1.2. Выпуск токенов в командной строке

Чтобы произвести выпуск пары токенов через терминал, необходимо отправить POST-запрос с помощью консольной утилиты `curl` или любым другим удобным способом. В данном примере используется `curl`:

```
curl -k -X POST 'https://api.rubackup.local:5656/api/v1/auth/login' \ ❶ ❷ ❸
-H 'accept: application/json' \ ❹
-H 'Content-Type: application/json' \ ❹
-d '{
  "rubackup_database": "rubackup", \ ❺
  "rubackup_user": "rubackup", \ ❻
  "password": "Testing12345", \ ❼
  "auth_type": "database", \ ❽
  "domain": "rubackup" ❾
}'
```

- ❶ `https://api.rubackup.local` — адрес, где запущен `rubackup_api`
- ❷ `5656` - порт, который будет использоваться как часть адреса сервера API
- ❸ `/api/v1/auth/login` — путь до запроса на авторизацию
- ❹ `-H <argument>` - значения, передаваемые в заголовках
- ❺ `rubackup_database` — имя служебной базы данных
- ❻ `rubackup_user` — имя суперпользователя
- ❼ `password` — пароль суперпользователя
- ❽ `auth_type` — тип аутентификации (база данных или контроллер домена)
- ❾ `domain` — домен для авторизации через контроллер домена

Результат вернётся в формате JSON:

```
{
  "data": {
    "access_token":
      "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJmcmVzaCI6ZmFsc2UsIm1hdCI6MTcxNTE0Dc
      2MCwianRpIjoInThjZjQ5MDAtZjE5Ni00OTZlLThkZjQtMTMhZjY2M3Y2QxIiwidHlwZSI6ImFj
      Y2VzcyIsInN1YiI6InJ1bmFja3VwIiwibmJmIjojNzE1MTU0NzYwLCJjc3JmIjoInGM4YzMTctN
      DAwNS00YzZiLTk2MWMtMGM1MDU5Njc5ZTA2IiwiaXhwIjojNzE1MTU5NjYwLCJmYW1pbHkiOiJ4cn
      ByeHR3ZHJyZ3FsamN4In0.VK5K6v-0_NxSx42bU5dEMIQAYYzxn-GTxmbjhxXjYSs",
    "csrf_token": "4c8c3457-4005-4c6b-961c-0c5059671e06",
    "csrf_token": "148ff198-c67e-4098-a702-beb8a14c7df4",
    "last_fail_login": "2024-12-11T17:39:43.564638",
    "last_success_login": "2024-12-11T17:40:07.036618",
    "package_version": "<версия пакетов>",
  }
}
```

```

    "refresh_token":
    "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJmcmVzaCI6ZmFsc2UsImhhdCI6MTcxNTE0Dc2MCwianRpIjoiotUzZjAxZWYtMTM4Zi00Y2ZlTg4ODItNGI0NWQ0N2YxM2I2IiwidHlwZSI6InJlZnJlc2giLCJzdWIiOiJydWJhY2t1cCIsIm5iZiI6MTcxNTE0Dc2MCwiY3NyZiI6IjRjOGMzNDU3LTQwMDUtNGM2Yi05NjFjLTBjNTA1OTY3MWUwNiIsImV4cCI6MTcxNTI0NTE2MCwiZmFtaWx5IjoieHJwcnh0d2RycmdxbGpjcCJ9.7TmdI0Cmm4knApNINDoYJuJIYdRlzuuc1hS-1c4Y8Ws",
    "role": [
        "superuser"
    ],
    "rubackup_server_address": "10.177.xx.xxx",
    "user_name": "rubackup"
},
"errors": {},
"is_error": false,
"message": ""
}

```

Если необходимо получить еще и access_token и refresh_token из cookies, то `curl` следует передать опцию `--cookie-jar -`, например:

```

curl -k --cookie-jar - -X POST
'https://api.rubackup.local:5656/api/v1/auth/login' \ ❶
-H 'accept: application/json' \
-H 'Content-Type: application/json' \
-d '{
  "rubackup_database": "rubackup", \
  "rubackup_user": "rubackup", \
  "password": "Testing12345", \
  "auth_type": "database", \
  "domain": "rubackup"
}'

```

❶ Добавлена опция `--cookie-jar -`

С этой опцией к выводу добавится следующая информация:

```

#HttpOnly.rubackup.local FALSE / TRUE 0 refresh_token_cookie
eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJmcmVzaCI6ZmFsc2UsImhhdCI6MTcxNTE2MDUz
NywianRpIjoiyZM4MTE0N2YtNDI0MS00Y2NjLTlhZjYtNDcwODAxMWFmOTRjIiwidHlwZSI6InJlZ
nJlc2giLCJzdWIiOiJydWJhY2t1cCIsIm5iZiI6MTcxNTE2MDUzNywiY3NyZiI6IjRjOGMzNDU3LT
QwMDUtNGM2Yi05NjFjLTBjNTA1OTY3MWUwNiIsImV4cCI6MTcxNTI0NjkzNywiZmFtaWx5IjoieXR
qemR4dG1xd212bHlyayJ9.g0aFkoob7jAwZ5Bv6FzbOPo3_Q6y-vBpASqXXk19tyw

#HttpOnly.rubackup.local FALSE / TRUE 0 access_token_cookie

```

```
eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJmcmVzaCI6ZmFsc2UsIm1hdCI6MTcxNTE2MDUz
NywianRpIjoiodmNDI0N2Q0NjJlMy00ZjFjLTk4NzgtNDY4OTM2YjRlNTJmIiwidHlwZSI6ImFjY
2VzcyIsInN1YiI6InJ1YmFja3VwIiwibmJmIjoxNzE1MTYwNTM3LCJjc3JmIjoingM4YzM0NTctND
AwNS00YzZiLTk2MWMtMGM1MDU5Njc0ZTA2IiwiaXNjaXNzE1MTYxNDM3LCJmYW1pbHkiOiJ5dGp
6ZHh0bXF3bXZseXJrIn0.nj-UbudgyYqGmopRpK9du9gPaZcn_j0_5yhp0A5DLw8
```

6.2. Обновление токенов

6.2.1. Обновление токенов в браузере

1. Перейдите по адресу `https://<app_host>:<app_port>/api/v1/`.
2. Перейдите на вкладку «Auth» и выберите эндпоинт `/auth/refresh` (Рисунок 4):

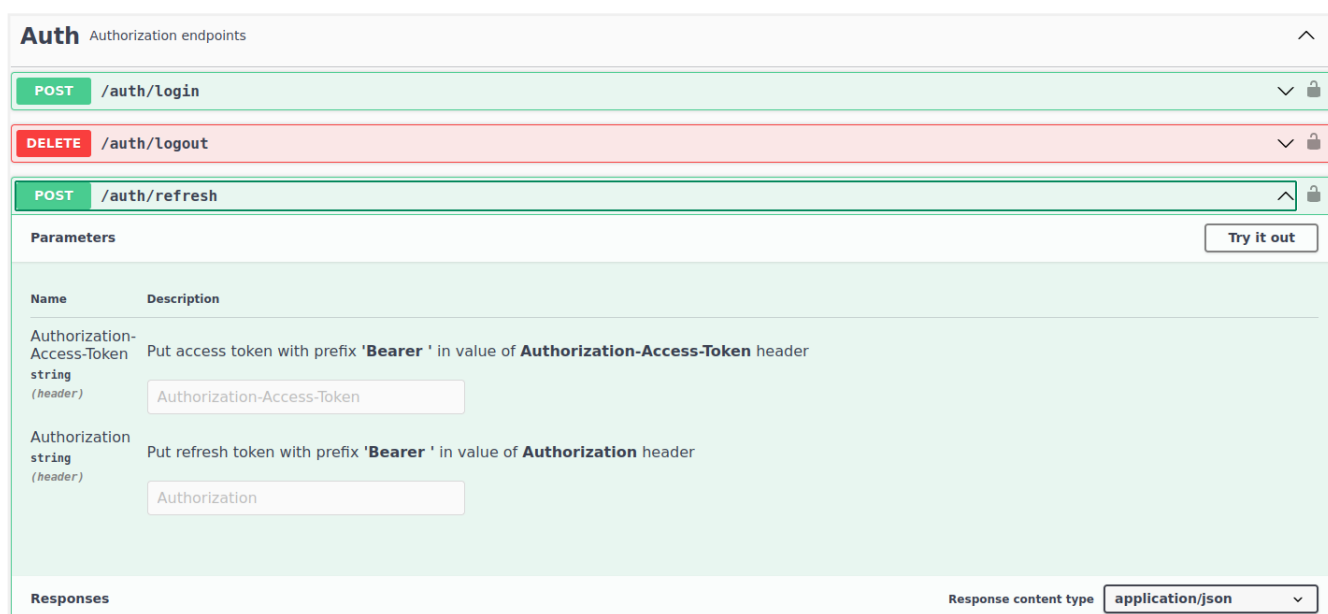


Рисунок 4. Эндпоинт `/auth/refresh`

1. Нажмите кнопку «Try it out». Если авторизация была пройдена в этом же браузере и `access_token` и `refresh_token` все ещё находятся в cookie, то нажмите кнопку «Execute». В ином случае явно укажите токены с префиксом `Bearer` для параметров `Authorization-Access-Token` и `Authorization` (Рисунок 5).

Auth Authorization endpoints

POST /auth/login

DELETE /auth/logout

GET /auth/permissions Current user's permissions

POST /auth/refresh

Parameters

Cancel

Name	Description
Authorization-Access-Token string (header)	Put access token with prefix 'Bearer ' in value of Authorization-Access-Token header
Authorization string (header)	Put refresh token with prefix 'Bearer ' in value of Authorization header

Execute

Рисунок 5. Заполнение полей авторизации

В результате проделанных операций Вам вернётся перевыпущенный `access_token`, перевыпущенные `refresh_token` и `csrf_token`, а также сопутствующая информация о пользователе прошедшем авторизацию (Рисунок 6). Также сервис автоматически разместит перевыпущенные `access_token` и `refresh_token` в cookie-файлах.

```
curl -X 'POST' \
  https://10.177.33.245:5656/api/v1/auth/refresh \
  -H 'accept: application/json' \
  -H 'X-CSRF-TOKEN: 0497040f-ac65-4859-aa0a-1e9b0959bdef' \
  -d ''
```

Request URL

https://10.177.33.245:5656/api/v1/auth/refresh

Server response

Code Details

200 Undocumented Response body

```
{
  "data": {
    "access token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJmcmVzaCI6ZmFsc2UsImhhdCI6MTcyNjI0MDM2NSwianRpIjoizDQwNTI2ZGQ0MTM0ZC00NDhhLWlyNDU0NjE1VTM5MTFiZTRjIiwidHlwZSI6ImFjY3N4Qm9uZG9uZS1rZPxdyYewQSLc0um7o",
    "csrf token": "0497040f-ac65-4859-aa0a-1e9b0959bdef",
    "refresh token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJmcmVzaCI6ZmFsc2UsImhhdCI6MTcyNjI0MDM2NSwianRpIjoizDQwNTI2ZGQ0MTM0ZC00NDhhLWlyNDU0NjE1VTM5MTFiZTRjIiwidHlwZSI6ImFjY3N4Qm9uZG9uZS1rZPxdyYewQSLc0um7o",
    "role": "superuser",
    "ruBackup_server_address": "10.177.33.245:5656",
    "user name": "ruBackup"
  },
  "errors": {},
  "is error": false,
  "message": ""
}
```

Response headers

```
access-control-allow-credentials: true
access-control-allow-origin: https://10.177.33.245:5656
content-length: 1082
content-type: application/json
server: TornadoServer/6.2
vary: Origin
```

Рисунок 6. Ответ с перевыпущенными токенами

6.2.2. Обновление токенов в терминале

Чтобы произвести перевыпуск пары токенов через терминал, необходимо отправить POST-запрос с помощью консольной утилиты `curl` или любым другим удобным способом. В данном примере используется `curl`:

```
curl -k -X POST 'https://api.rubackup.local:5656/api/v1/auth/refresh' \ ① ②
③
-H 'accept: application/json' \ ④
-H 'Authorization-Access-Token: Bearer <access_token>' \ ④ ⑤
-H 'Authorization: Bearer <refresh_token>' ④ ⑥
```

- ① `https://api.rubackup.local` — адрес, где запущен `rubackup_api`
- ② `5656` - порт, который будет использоваться как часть адреса сервера API
- ③ `/api/v1/auth/refresh` — путь до запроса на обновление токена
- ④ `-H <argument>` - значения, передаваемые в заголовках
- ⑤ `Authorization-Access-Token` — значение полученного при авторизации `access_token` с префиксом `Bearer`
- ⑥ `Authorization` — значение полученного при авторизации `refresh_token` с префиксом `Bearer`

Результат вернётся в формате JSON:

```
{
  "data": {
    "access_token":
      "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJmcmVzaCI6ZmFsc2UsIm1hdCI6MTcxNTE2MjAwOSwianRpIjoiodDhlZDcwYzUtM2Y4Yi00MDVmLWJlMjMtYzYzNTA3YzA5MmViIiwidHlwZSI6ImFjY2VzcyIsInN1YiI6InJ1YmFja3VwIiwibmJmIjoxNzE1MTYyMDA5LCJjc3JmIjoingM4YzMTc0NTctNDh1Y250cWVja29ma2p0In0.XHppz9B-eYoEcXZAwcf-nbXKnu8rC_kXIMRWIU4gZXc",
    "csrf_token": "4c8c3457-4005-4c6b-961c-0c5059671e06",
    "package_version": "<version>",
    "refresh_token":
      "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJmcmVzaCI6ZmFsc2UsIm1hdCI6MTcxNTE2MjAwOSwianRpIjoioGNlYmY4NGMtOTM4M000YTLhLWE1OGItNmU4ZDM3OTgxMTI0IiwidHlwZSI6InJlZnJlc2giLCJzdWIiOiJydWJhY2t1cCIsIm5iZiI6MTcxNTE2MjAwOSwiy3NyZiI6IjRjOGMzNDU3LTQwMDUtNGM2Yi05NjFjLTBjNTA1OTY3MWUwNiIsImV4cCI6MTcxNTI0ODM3MCwiZmFtaWx5IjoizXl4dWNudHFLY2tvZmtqdCJ9.H_TPOn-CF70bYUKa5AP4U10MsiPEnzy5foxTZXHi0dE",
    "role": [
      "superuser"
    ],
    "rubackup_server_address": "10.177.xx.xxx",
    "user_name": "rubackup"
  },
  "errors": {},
  "is_error": false,
  "message": ""
}
```

}

Если необходимо получить еще и `access_token` и `refresh_token` из cookies, то в команде `curl` следует передать опцию `--cookie-jar -`, например:

```
curl -k --cookie-jar - -X POST
'https://api.rubakup.local:5656/api/v1/auth/refresh' \
-H 'accept: application/json' \
-H 'Authorization-Access-Token: Bearer <access_token>' \
-H 'Authorization: Bearer <refresh token>'
```

С этой опцией к выводу добавится следующая информация:

```
#HttpOnly.rubackup.local FALSE / TRUE 0 refresh_token_cookie
eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJmcmVzaCI6ZmFsc2UsIm1hdCI6MTcxNTE2MjAwOSwianRpIjoiOGNlYmY4NGMtOTM4MS00YTlhLWE1OGItNmU4ZDM3OTgxMTI0IiwidHlwZSI6InJlZnJlc2giLCJzdWIiOiJydWJhY2t1cCIsIm5iZiI6MTcxNTE2MjAwOSwiY3NyZiI6IjRjOGMzNDU3LTQwMDUtNGM2Yi05NjFjLTBjNTA1OTY3MWUwNiIsImV4cCI6MTcxNTI0ODM3MCwiZmFtaWx5IjoiZX14dWNudHFlY2tvZmtqdCJ9.H_TPOn-CF70bYUKa5AP4U10MsiPEnzy5foxTZXHi0dE

#HttpOnly.rubackup.local FALSE / TRUE 0 access_token_cookie
eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJmcmVzaCI6ZmFsc2UsIm1hdCI6MTcxNTE2MjAwOSwianRpIjoiODhlZDcwYzUtMjY4Yi00MDVmLWJlMjMtYzcvNTA3YzA5MmViIiwidHlwZSI6ImFjY2VzcyIsInN1YiI6InJ1YmFja3VwIiwibmJmIjoxNzE1MTYyMDA5LCJjc3JmIjoiNGM4YzVM0NTctNDAwNS00YzZiLTk2MWMtMGM1MDU5NjcZTA2IiwizXhwIjoxNzE1MTYyOTA5LCJmYW1pbHkiOiJleXh1Y250cWVja29ma2p0In0.XHppz9B-eYoEcXZAwcf-nbXKnu8rC_kXIMRWIU4gZXc
```

6.3. Отзыв токенов

Если ваши токены были скомпрометированы, то можно сделать отзыв двумя способами: через терминал с помощью консольной команды `curl`, либо через веб-интерфейс.

Для отзыва токенов через терминал отправьте DELETE-запрос с RefreshToken с помощью консольной утилиты curl или любым другим удобным для вас способом.

В данном примере используется `curl`:

```
curl -k -X DELETE https://api.rubackup.local:5656/api/v1/logout \
-H 'Content-Type: application/json' \
-H 'Authorization: <refresh token>'
```

Для отзыва токенов с помощью web-интерфейса необходимо выполнить следующие действия:

Перейдите по адресу `https://<app_host>:<app_port>/api/v1/` (Рисунок 7):

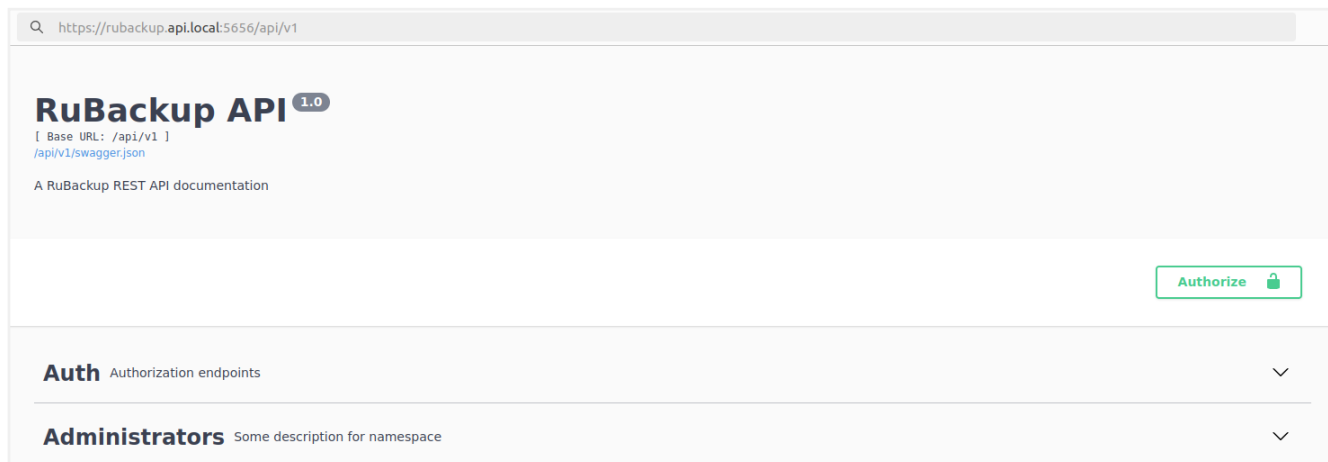


Рисунок 7. Внешний вид Swagger

Перейдите на вкладку «Auth» и выберите эндпоинт `/auth/logout` (Рисунок 8). Предварительно убедитесь, что выполнена авторизация (подробнее об авторизации написано в разделе «Авторизация пользователя в веб-интерфейсе»).

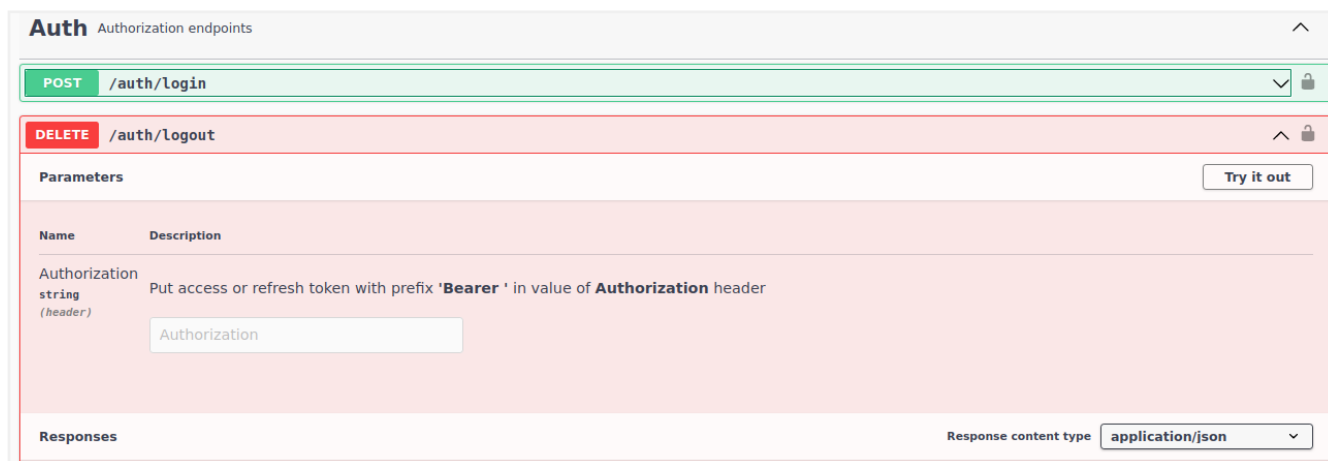


Рисунок 8. Эндпоинт `/auth/logout`

Нажмите кнопку «Try it out». Если авторизация была пройдена в этом же браузере и `access_token` и `refresh_token` все ещё находятся в cookie, то нажмите кнопку «Execute» (Рисунок 9). В ином случае явно укажите отзываемый RefreshToken (с префиксом `Bearer`) для параметра `Authorization`.

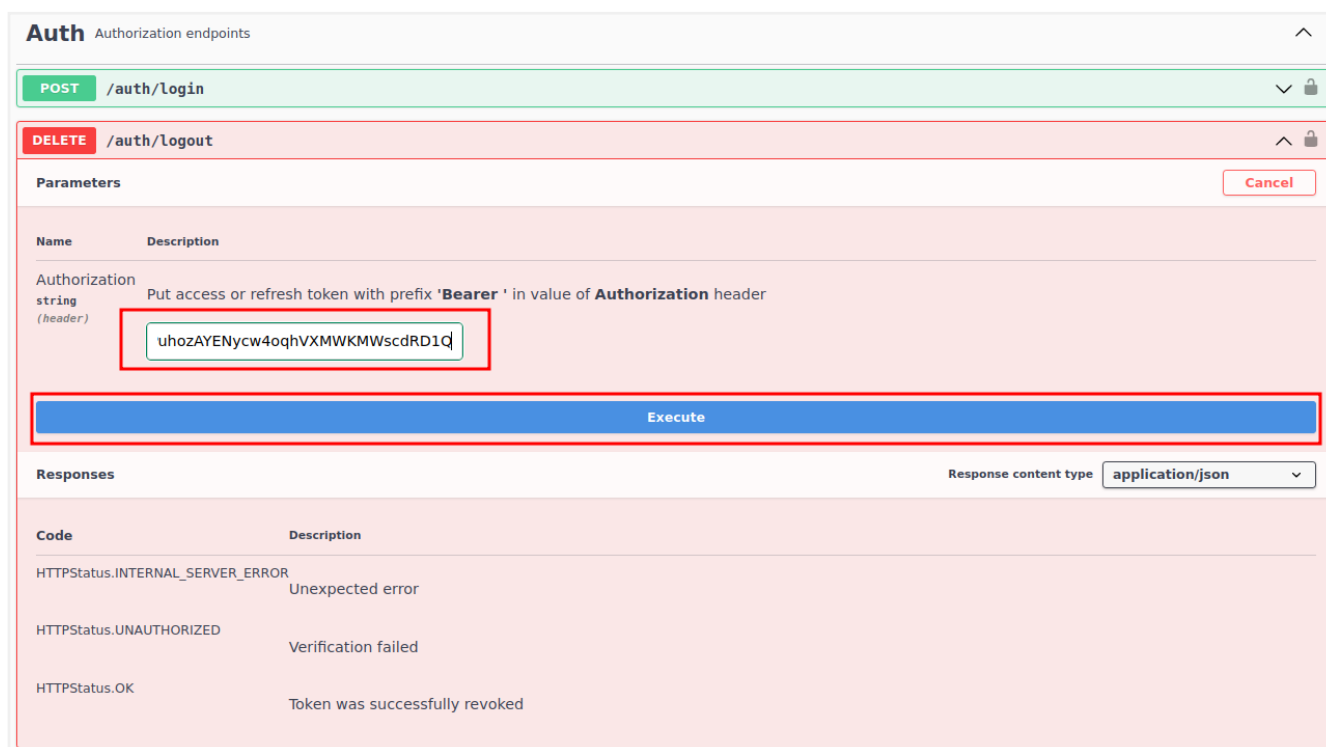


Рисунок 9. Отзываемый токен

В результате проделанных операций вернётся информационное сообщение о том, что токены деактивированы ([Рисунок 10](#)).

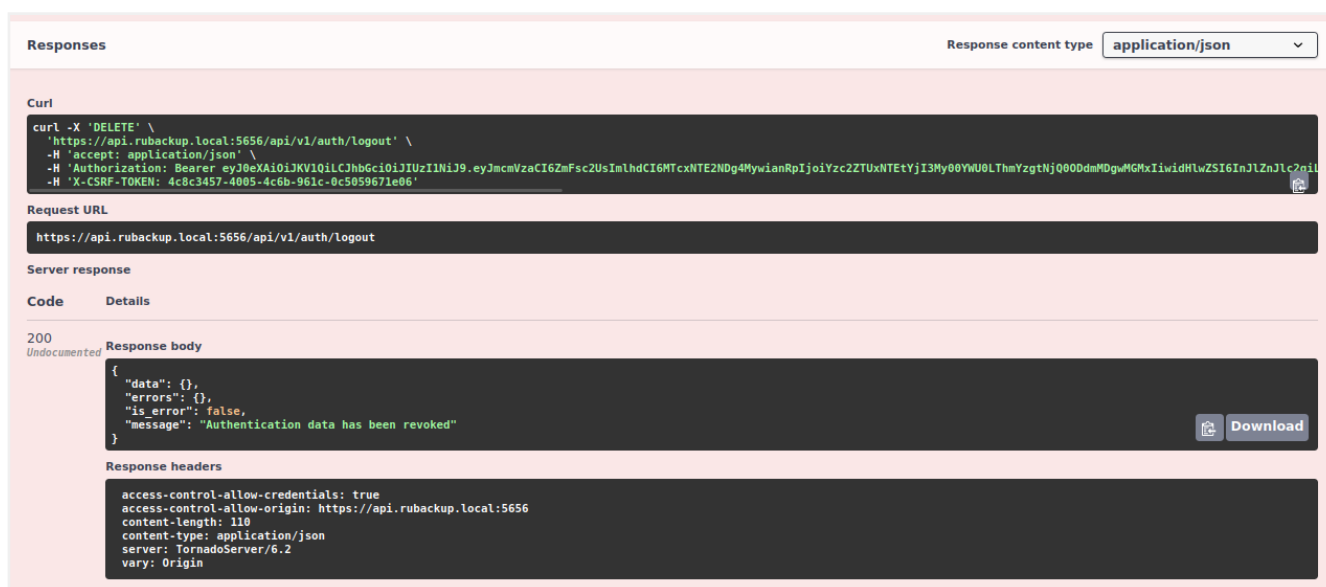


Рисунок 10. Сообщение об успешном отзыве токена

6.4. Аутентификация пользователя

Для начала работы с ресурсами сервера перейдите по адресу `https://<app_host>:<app_port>/api/v1/` в Вашем браузере, выполните получение токенов любым удобным способом и нажмите на кнопку «Authorize» ([Рисунок 11](#)):



Рисунок 11. Кнопка авторизации

В открывшейся форме заполните поле «Value». В данном поле необходимо ввести ранее полученный csrf_token. Затем нажмите кнопку «Authorize» ([Рисунок 12](#)):

Рисунок 12. Форма ввода токена авторизации

Закройте форму нажатием на кнопку «Close» ([Рисунок 13](#)). Теперь вы авторизованы и можете работать с ресурсами сервера RuBackup.

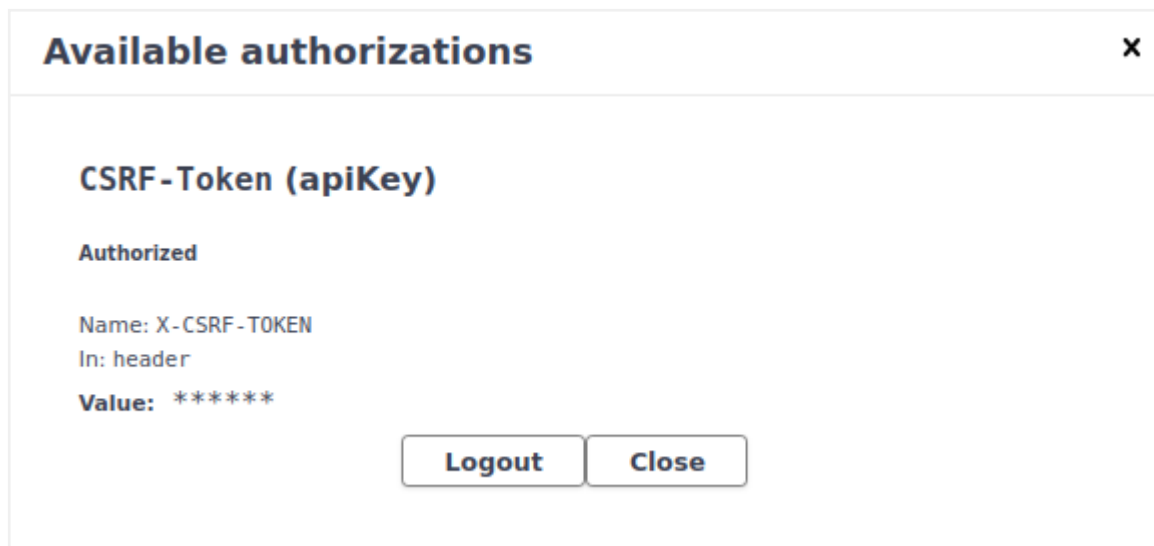


Рисунок 13. Кнопка закрытия формы

6.5. Аутентификация с Microsoft Active Directory

Аутентификация в REST API возможна через контроллер домена Microsoft Active Directory.

CPK поддерживает интеграцию с Microsoft Active Directory версий 2012 R2 или 2016, развернутой на Microsoft Windows Server 2016.



Предварительные условия для аутентификации через контроллер домена

1. Настроен контроллер домена Microsoft Active Directory. В контроллере домена существует хотя бы одна группа, в которой есть хотя бы один пользователь.
2. В CPK настроено соединение с контроллером домена Microsoft Active Directory.
3. В CPK создана ассоциация групп RuBackup и ролей MS Active Directory.

Подробнее о настройке контроллера домена MS Active Directory — в «Интеграция RuBackup со средствами управления доменом Microsoft Active Directory».

6.5.1. Выпуск `access_token`, `refresh_token` и `csrf_token` через браузер

Перед авторизацией убедитесь, что все [предварительные условия](#) для аутентификации через контроллер домена выполнены.

Для примера будут использованы следующие данные MS AD:

- Группа контроллера домена: `test_group`
- Пользователь контроллера домена : `testuser`
- Роль группы: Супервайзер

- Домен: RUBACKUP
- Пароль пользователя в контроллере: P@ssw0rd

Для выпуска пары токенов:

1. Перейдите по адресу `https://<app_host>:<app_port>/api/v1/` (Рисунок 14):

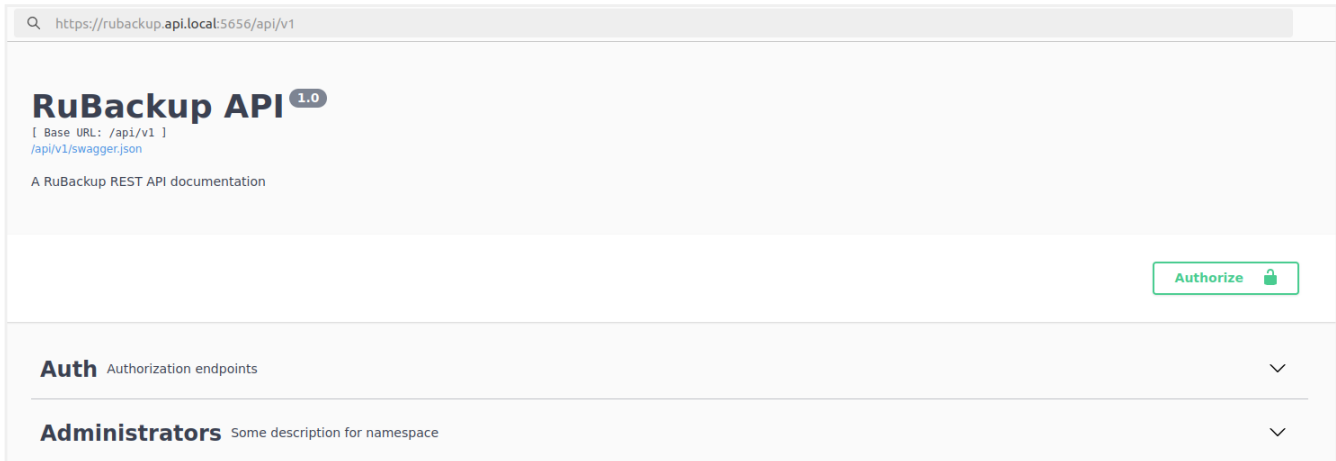


Рисунок 14. Внешний вид Swagger

1. Перейдите на вкладку «Auth» и выберите эндпоинт `/auth/login`.
2. Нажмите кнопку «Try it out», заполните payload актуальными данными и нажмите «Execute» (Рисунок 15):

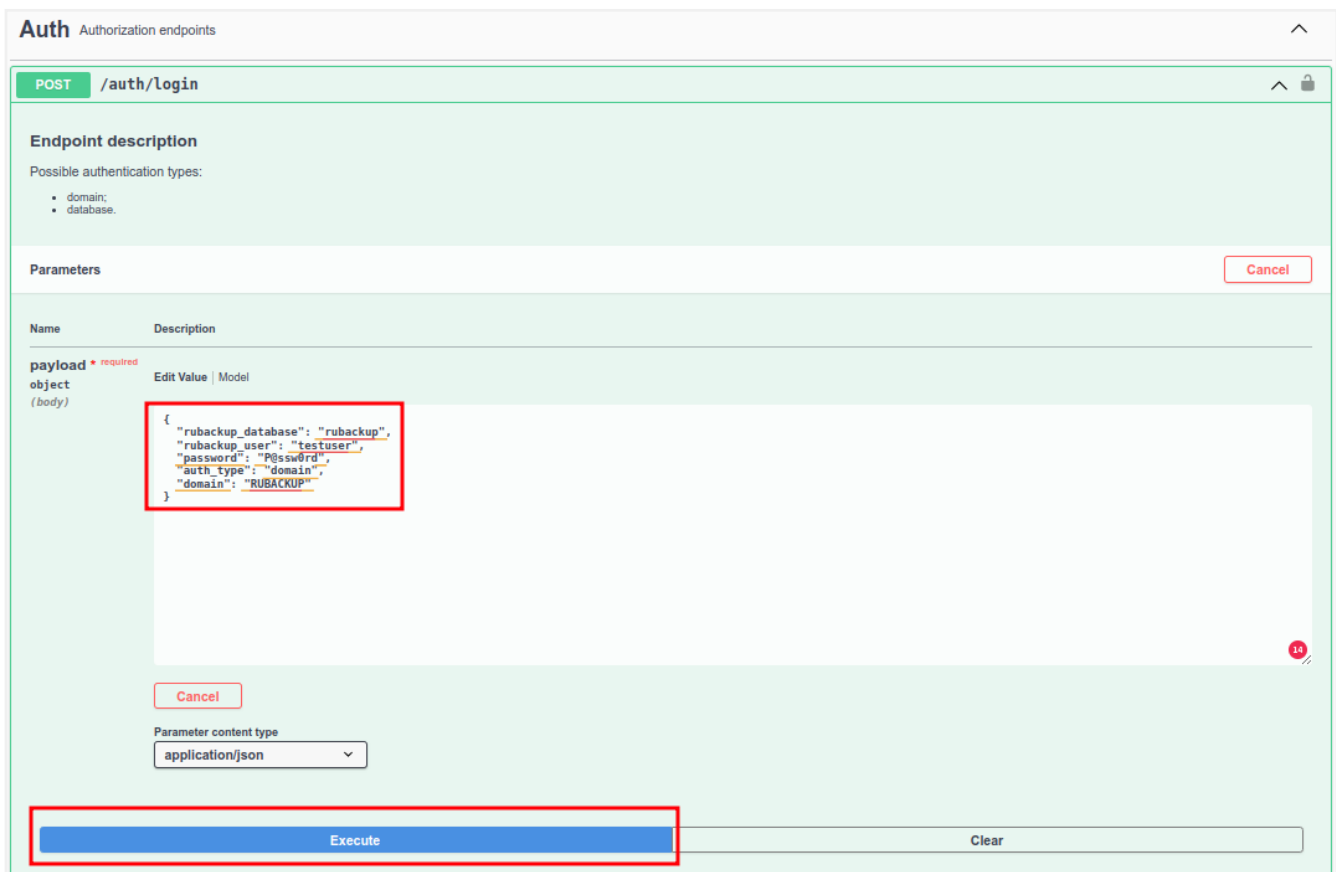


Рисунок 15. Заполнение payload авторизационными данными

В результате получены `access_token`, `refresh_token` и `csrf_token`, а также сопутствующая информация о пользователе, прошедшем авторизацию (Рисунок 16). Сервис автоматически разместит `access_token` и `refresh_token` в cookie-файлах.

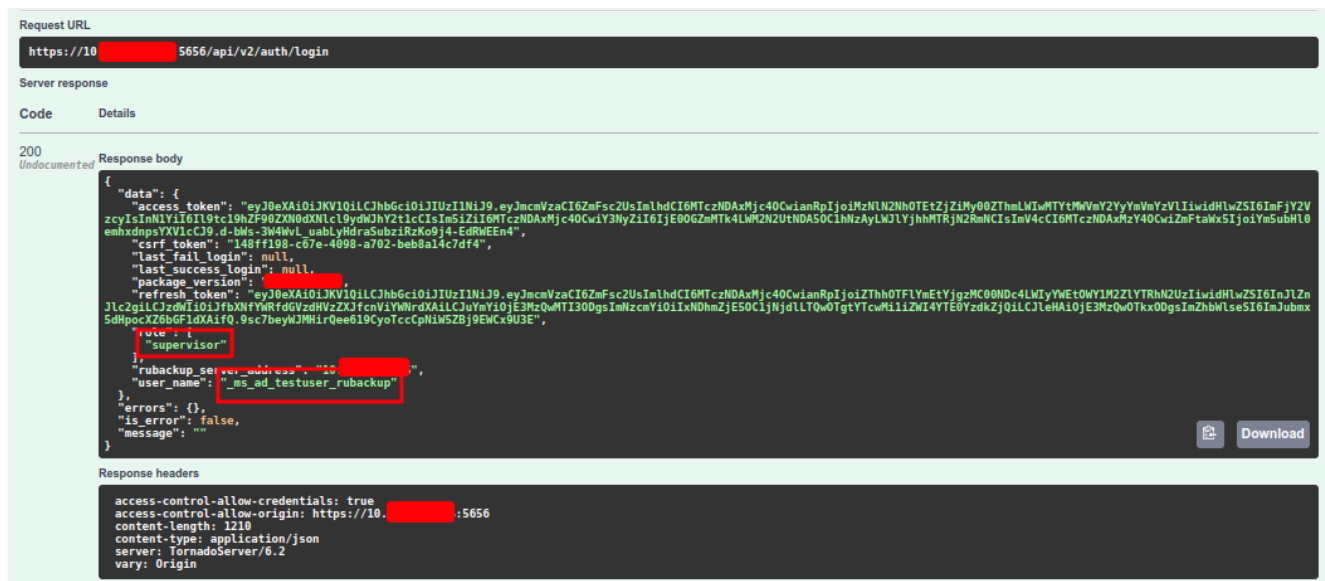


Рисунок 16. Ответ сервера

В параметре `role` будет возвращена роль пользователя, которая соответствует роли группы, в которой находится пользователь. Роль группы определяется через ассоциацию групп и ролей. Подробнее о создании ассоциации — в «Интеграция RuBackup со средствами управления доменом Microsoft Active Directory».

6.5.2. Выпуск `access_token`, `refresh_token` и `csrf_token` в командной строке

Перед авторизацией убедитесь, что все условия из раздела Предварительные условия для аутентификации через контроллер домена выполнены.

Для примера будут использованы следующие данные MS AD:

- Группа контроллера домена: `test_group`
- Пользователь контроллера домена : `testuser`
- Роль группы: Супервайзер
- Домен: `RUBACKUP`
- Пароль пользователя в контроллере: `P@ssw0rd`

Чтобы произвести выпуск пары токенов через терминал, необходимо отправить POST-запрос с помощью консольной утилиты `curl` или любым другим удобным способом. В данном примере используется `curl`:

```
curl -k -X POST 'https://api.rubackup.local:5656/api/v1/auth/login' \ ❶ ❷ ❸
--header 'Content-Type: application/json' \ ❹
--data '{
```

```
"rubackup_database": "rubackup", ⑤
"rubackup_user": "testuser", ⑥
"password": "P@ssw0rd", ⑦
"auth_type": "domain", ⑧
"domain": "RUBACKUP" ⑨
}'
```

- ① `https://api.rubackup.local` — адрес, где запущен `rubackup_api`
- ② `5656` - порт, который будет использоваться как часть адреса сервера API
- ③ `/api/v1/auth/login` — путь до запроса на авторизацию
- ④ `--header <argument>` - значения, передаваемые в заголовках
- ⑤ `rubackup_database` — имя служебной базы данных
- ⑥ `rubackup_user` — имя суперпользователя
- ⑦ `password` — пароль суперпользователя
- ⑧ `auth_type` — тип аутентификации (база данных или контроллер домена)
- ⑨ `domain` — домен для авторизации через контроллер домена

Результат вернётся в формате JSON:

```
{
  "data": {
    "access_token":
      "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJmcmVzaCI6ZmFsc2UsIm1hdCI6MTczNDExNDMzNCwianRpIjoizWRjYzY0NzItYTdhOC00MmQ5LWlXNmMtMWE4NzQ2ZTI5NWNkIiwidHlwZSI6ImFjY2VzcyIsInN1YiI6Ii9tc19hZF90ZXN0dXNlc19ydWJhY2t1cCI6Im5iZiI6MTczNDExNDMzNCwiY3NyZiI6IjE0OGZmMTk4LWMM2N2U0tNDA5OC1hNzAyLWJlYjhhMTRjN2RmNCIsImV4cCI6MTczNDExNTIzNCwiZmFtaWx5IjoiaHh0Z3hkcWVobmxcncWdidCJ9.5YU2yxGZshimv6kqqgvP-t6ynn timerVxDhlGurcXFHg0s",
    "csrf_token": "148ff198-c67e-4098-a702-beb8a14c7df4",
    "last_fail_login": null,
    "last_success_login": "2024-12-12T17:13:08.305226",
    "package_version": "<Версия пакетов>",
    "refresh_token":
      "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJmcmVzaCI6ZmFsc2UsIm1hdCI6MTczNDExNDMzNCwianRpIjoiaHh0Z3hkcWVobmxcncWdidCJ9.5YU2yxGZshimv6kqqgvP-t6ynn timerVxDhlGurcXFHg0s",
    "role": [
      "supervisor"
    ],
  },
}
```

```

    "rubbackup_server_address": "10.177.xxx.xxx",
    "user_name": "_ms_ad_testuser_rubbackup"
  },
  "errors": {},
  "is_error": false,
  "message": ""
}

```

Если необходимо получить еще и `access_token` и `refresh_token` из cookies, то в команде `curl` следует указать опцию `--cookie-jar -`:

```

curl -k -X --cookie-jar - POST
'https://api.rubbackup.local:5656/api/v1/auth/login' \ ❶
--header 'Content-Type: application/json' \
--data '{
  "rubbackup_database": "rubbackup",
  "rubbackup_user": "testuser",
  "password": "P@ssw0rd",
  "auth_type": "domain",
  "domain": "RUBACKUP"
}'

```

❶ Добавлена опция `--cookie-jar -`

С этой опцией к выводу добавится следующая информация:

```

#HttpOnly.rubbackup.local FALSE / TRUE 0 refresh_token_cookie
eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJmcmVzaCI6ZmFsc2UsIm1hdCI6MTcxNTE2MDUz
NywianRpIjoIYzM4MTE0N2YtNDI0MS00Y2NjLTlhZjYtNDcwODAxMWFmOTRjIiwidHlwZSI6InJlZ
nJlc2giLCJzdWIiOiJydWJhY2t1cCIscIm5iZiI6MTcxNTE2MDUzNywiY3NyZiI6IjRjOGMzNDU3LT
QwMDUtNGM2Yi05NjFjLTBjNTA1OTY3MwUwNiIsImV4cCI6MTcxNTI0NjkzNywiZmFtaWx5IjoieXR
qemR4dG1xd212bHlyayJ9.g0aFkoob7jAwZ5Bv6Fzb0Po3_Q6y-vBpASqXXk19tyw__
#HttpOnly_.rubbackup.local FALSE / TRUE 0 access_token_cookie
eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJmcmVzaCI6ZmFsc2UsIm1hdCI6MTcxNTE2MDUz
NywianRpIjoI0DhmNDI0N2QtNjJlMy00ZjFjLTk4NzgtNDY4OTM2YjRlNTJmIiwidHlwZSI6ImFjY
2VzcyIsInN1YiI6InJlYmFja3VwIiwibmJmIjojNzE1MTYwNTM3LCJjc3JmIjoIbG4YzM0NTctND
AwNS00YzZiLTk2MwMtMGM1MDU5NjcXZTA2IiwiaXhwIjojNzE1MTYxNDM3LCJmYW1pbHkiOiJ5dGp
6ZHh0bXF3bXZseXJrIn0.nj-UbudgyYqGmopRpK9du9gPaZcn_j0_5yhp0A5DLw8

```

Глава 7. Отправка запросов

7.1. Отправка запросов в браузере

Для отправки запроса на создание срочной резервной копии в браузере необходимо выполнить следующие действия:

Перейдите по адресу `https://<app_host>:<app_port>/api/v1/` (Рисунок 17):

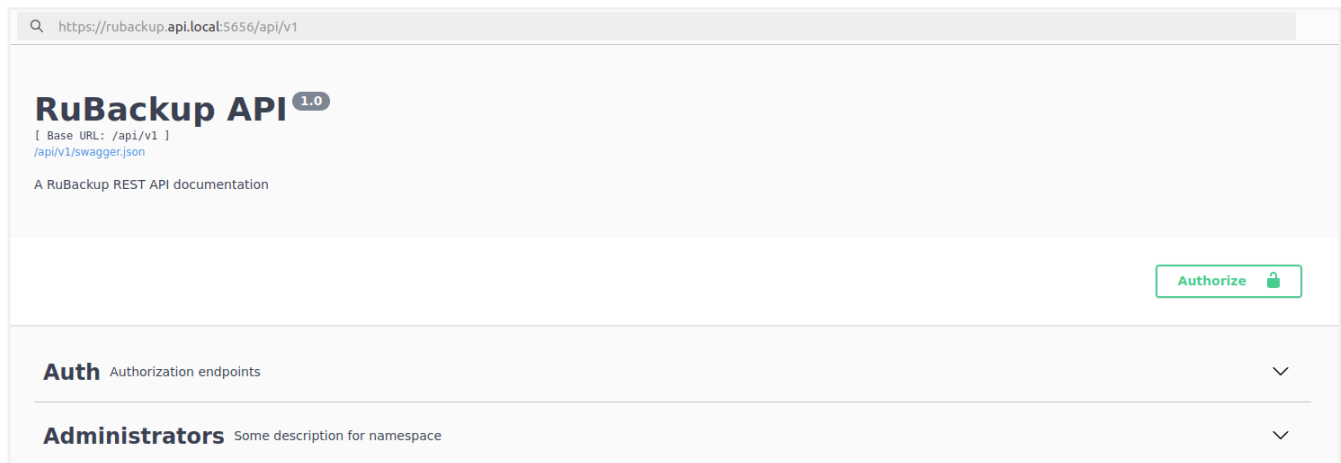


Рисунок 17. Внешний вид интерактивной документации Swagger

Перейдите к разделу Task queue и выберите метод POST (Рисунок 18):

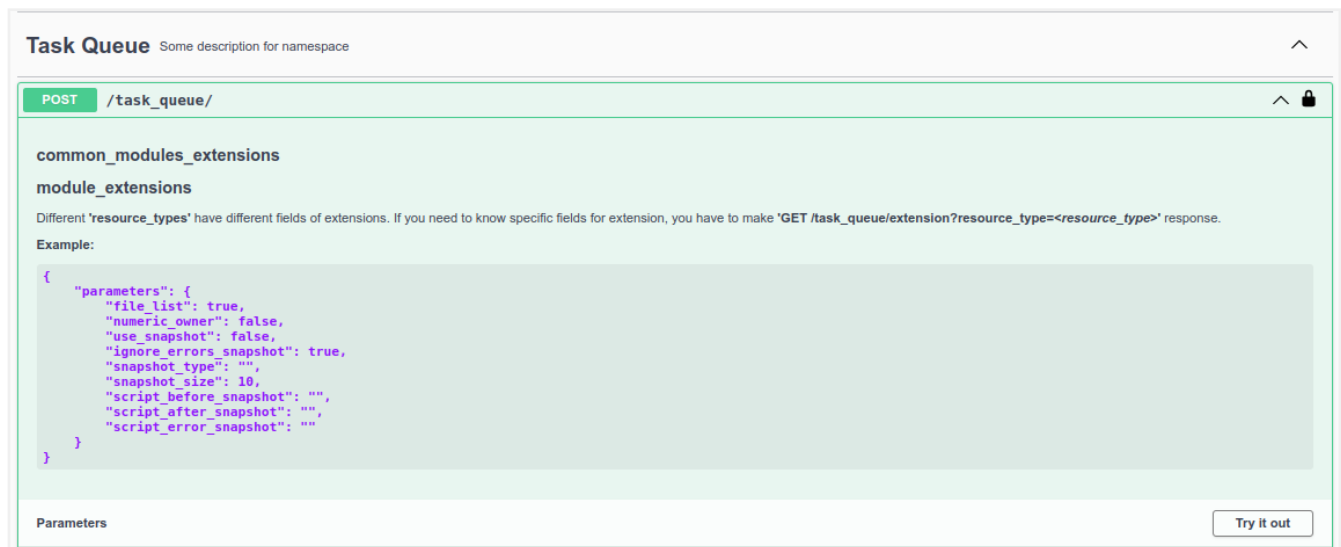


Рисунок 18. Метод `/task_queue`

Нажмите кнопку «Try it out». Заполните обязательные поля тела запроса, например (для создания копии файлового ресурса):

```
{
  "data": {
    "task": {
```

```

"resource_type": "File system",
"resource": "<имя ресурса>",
"client": "<client hostname> (<client HW ID>)",
"backup_type": "full",
"pool": "<имя пула>",
"crypto": "nocrypt",
"priority": 100,
"storage_duration": "1 Years",
"archiving": false
}
}
}

```

Нажмите «Execute» (Рисунок 19):

The screenshot shows a REST client interface with a 'Parameters' tab. A table lists parameters with columns 'Name' and 'Description'. The 'payload' parameter is highlighted in red and marked as 'required'. Its value is a JSON object: `{ "data": { "task": { "resource_type": "File system", "resource": "<имя ресурса>", "client": "<client hostname> (<client HW ID>)", "backup_type": "full", "pool": "<имя пула>", "crypto": "nocrypt", "priority": 100, "storage_duration": "1 Years", "archiving": false } } }`. Below the table, the 'Parameter content type' is set to 'application/json'. At the bottom, the 'Execute' button is highlighted with a red rectangle, and a 'Clear' button is visible to its right. The 'Responses' section at the bottom shows the 'Response content type' set to 'application/json'.

Рисунок 19. Кнопка **Execute**

В результате должен быть получен статус-код 201 CREATED и ответ с информацией о том, что задача добавлена в очередь (Рисунок 20):

The screenshot shows the 'Code' tab of the REST client. The status code is 201. The 'Response body' is a JSON object: `{ "is_error": false, "message": "Resource with name 'Task Queue' was created successfully! TASK WAS ADDED TO QUEUE", "data": {}, "errors": {} }`. The 'Response headers' are: `access-control-allow-credentials: true, access-control-allow-origin: https://10.0.0.1:5656, content-length: 162, content-type: application/json, server: TornadoServer/6.2, vary: Origin`. A 'Download' button is visible next to the response body.

Рисунок 20. Успешный ответ сервера

Для отслеживания статуса выполнения задачи можно использовать [метод GET](#)

/task_queue.

7.2. Отправка запросов в терминале

Для того, чтобы отправить запрос на сервер, используя терминал, необходимо отправить запрос с помощью консольной утилиты `curl` или любым другим удобным способом. В данном примере используется `curl`. В запросе необходимо передать полученные ранее любым удобным способом `csrf-token` и `access-token`.

Для того, чтобы отправить запрос на создание срочной резервной копии файлового ресурса, необходимо выполнить:

```
curl --location 'https://<app_host>:<app_port>/api/v1/task_queue/' \
--header 'Content-Type: application/json' \
--header 'X-CSRF-TOKEN: <полученный при авторизации csrf-token>' \
--header 'Authorization: Bearer <полученный при авторизации access-token>' \
--data '{
  "data": {
    "task": {
      "client": "<client hostname> (<client HW ID>)",
      "resource": "<имя ресурса>",
      "resource_type": "File system",
      "backup_type": "full",
      "pool": "<имя пула>",
      "crypto": "nocrypt",
      "priority": 100,
      "storage_duration": "1 Years",
      "archiving": false
    }
  }
}'
```

После отправки данного запроса должен быть получен ответ с информацией о том, что задача добавлена в очередь:

```
{
  "is_error": false,
  "message": "Resource with name 'Task Queue' was created successfully! TASK WAS ADDED TO QUEUE: <номер в очереди>",
  "data": {},
  "errors": {}
}
```

Глава 8. Описание методов

RuBackup REST API содержит методы для взаимодействия с сервером резервного копирования. Пользователь может обращаться к каждому методу либо через браузер, используя визуальное представление спецификации, либо через консольную утилиту `curl`.

В каждом из методов каждого эндпоинта должен передаваться CSRF токен.

В части запросов GET в query-параметрах могут передаваться:

- номер страницы списка
- лимит отображения списка (необязательный параметр)
- параметр, по которому будет отсортирован список
- направление сортировки
- фильтр (необязательный параметр). Фильтр указывается в параметре `filterdata`. Формат, в котором можно указать фильтр (пример, зависит от эндпоинта):
 - `poolname="test"`
 - `poolname~"te"`
 - `lastonline<"2024-08-19T19:20:28.614838+03:00"`
 - `lastonline>"2024-08-19T19:20:28.614838+03:00"`

8.1. Авторизация

8.1.1. Получение токенов и прохождение авторизации (POST `/auth/login`)

`access_token` действителен 15 минут, `refresh_token` 24 часа, `csrf_token` - до перезагрузки сервиса `rest-api`

До 2.3 включительно

```
{
  "rubakup_database": "rubakup", ❶
  "rubakup_user": "rubakup", ❶
  "password": "12345" ❶
}
```

❶ Обязательное поле

Начиная с 2.4

```
{
```

```

"rubakup_database": "rubakup",
"rubakup_user": "rubakup",
"password": "12345",
"auth_type": "database", ①
"domain": "rubakup" ②
}

```

① `auth_type` — тип аутентификации (база данных или контроллер домена)

② `domain` — домен для авторизации через контроллер домена

200 OK

Успешная авторизация, получены токены

8.1.2. Деактивация имеющихся токенов (DELETE `/auth/logout`)

После выполнения запроса access и refresh токены более не действуют.

200 OK

Успешная деактивация токенов

401 UNAUTHORIZED

Verification failed

8.1.3. Обновление access токена после его истечения (POST `/auth/refresh`)

Срок действия токена доступа — 15 минут.

200 OK

Успешное обновление токена

401 UNAUTHORIZED

Один или оба токена не указаны

8.1.4. Получение списка прав пользователя (GET `/auth/permissions`)

200 OK

Успешно получен список прав текущего пользователя

8.2. Пользователи

8.2.1. Администраторы

Получение списка администраторов (GET `/administrators`)

200 OK

Получен список администраторов

Удаление списка ролей администратор (DELETE /administrators)

```
{
  "ids": [ ①
    1
  ]
}
```

① Обязательное поле

200 OK

Роль администратора удалена

404 NOT_FOUND

Роль администратора не найдена

Назначение пользователю роли администратора (POST /administrators)

```
{
  "data": {
    "client_group": "Some group", ①
    "administrator": "username" ①
  }
}
```

① Обязательное поле

201 CREATED

Роль администратора добавлена пользователю

Получение информации об одном администраторе (GET /administrators/{id})

Тела запроса нет. В query-параметрах передается id администратора, информацию о котором необходимо получить

200 OK

Получена информация об администраторе

Удаление роли администратора. (DELETE /administrators/{id})

Тела запроса нет. В query-параметрах передается id администратора, которого необходимо удалить

200 OK

Роль администратора удалена

404 NOT_FOUND

Роль администратора не найдена

8.2.2. Супервайзеры

Удаление списка пользователей роли супервайзера (DELETE /supervisors)

```
{
  "ids": [ ❶
    1
  ]
}
```

❶ Обязательное поле**200 OK**

Роль супервайзера удалена

404 NOT_FOUND

Роль супервайзера не найдена

Получение списка супервайзеров (GET /supervisors)**200 OK**

Получен список супервайзеров

Назначение пользователю роли супервайзера (POST /supervisors)

```
{
  "data": {
    "supervisor": "Nobody" ❶
  }
}
```

❶ Обязательное поле**201 CREATED**

Роль супервайзера добавлена пользователю

Удаление пользователю роли супервайзера (DELETE /supervisors/{id})

Тела запроса нет. В query-параметрах передается id супервайзера, информацию о котором необходимо получить

200 OK

Получена информация об супервайзере

Получение информации о супервайзере (GET /supervisors/{id})

Тела запроса нет. В query-параметрах передается id супервайзера, которого необходимо удалить

200 OK

Роль супервайзера удалена

404 NOT_FOUND

Роль супервайзера не найдена

8.2.3. Сопровождающие

Удаление списка пользователей роли сопровождающего (DELETE /maintainers)

```
{
  "ids": [ ❶
    1
  ]
}
```

❶ Обязательное поле

200 OK

Роль сопровождающего удалена

404 NOT_FOUND

Роль сопровождающего не найдена

Получение списка сопровождающих (GET /maintainers)**200 OK**

Получен список сопровождающих

Назначение пользователю роли сопровождающего (POST /maintainers)

```
{
  "data": {
    "media_server": "hostname", ❶
    "maintainer": "username" ❶
  }
}
```

1 Обязательное поле**201 CREATED**

Роль администратора добавлена пользователю

Удаление пользователю роли сопровождающего (DELETE /maintainers/{id})

Тела запроса нет. В query-параметрах передается id сопровождающего, информацию о котором необходимо удалить

200 OK

Роль сопровождающего удалена

404 NOT_FOUND

Роль сопровождающего не найдена

Получение информации о сопровождающем (GET /maintainers/{id})

Тела запроса нет. В query-параметрах передается id сопровождающего, информацию о котором необходимо получить

200 OK

Получена информация о сопровождающем

8.2.4. Аудиторы**Назначение пользователю роли аудитора (POST /auditors)**

```
{
  "data": {
    "auditor": "username" 1
  }
}
```

1 Обязательное поле**201 CREATED**

роль аудитора добавлена пользователю

Получение информации об аудиторе (GET /auditors/{id})**200 OK**

получена информация об аудиторе

Удаление аудитора (DELETE /auditors/{id})

Тела запроса нет. В query-параметрах передается id аудитора, которого необходимо удалить

200 OK

роль аудитора удалена

404 NOT_FOUND

роль аудитора не найдена

Получение списка аудиторов (GET /auditors)**200 OK**

получен список аудиторов

Удаление списка аудиторов (DELETE /auditors)

```
{
  "ids": [ ①
    1
  ]
}
```

① Обязательное поле

200 OK

роль аудитора удалена

404 NOT_FOUND

роль аудитора не найдена

8.2.5. Пользователи**Удаление списка пользователей (DELETE /users)**

```
{
  "ids": [ ①
    1
  ]
}
```

① Обязательное поле

200 OK

Список пользователей удален

404 NOT_FOUND

Пользователь не найден

Получение списка пользователей (GET /users)**200 OK**

Получен список пользователей

Добавление пользователя (POST /users)

Также в query-параметрах передаются:

- роль пользователя (администратор, супервайзер, сопровождающий)
- имя группы клиентов (для администраторов)
- хостнейм медиасервера (для сопровождающих)

```
{
  "data": {
    "group_id": "Nobody", ①
    "fullname": "",
    "address": "",
    "tel": "",
    "office": "",
    "email": "", ①
    "username": "", ①
    "user_password": "" ①
  }
}
```

① Обязательное поле

201 CREATED

Пользователь создан

Изменение пароля пользователя (PATCH /users/change_password)

```
{
  "new_password": "" ①
}
```

① Обязательное поле

200 OK

Пароль пользователя изменен

404 NOT_FOUND

Пользователь не найден

Редактирование пользователей (PATCH /users/{id})

Также в query-параметрах передается id пользователя, информацию о которой необходимо изменить

```
{
  "data": {
    "group_id": "Nobody",
    "fullname": "",
    "address": "",
    "tel": "",
    "office": "",
    "email": "",
    "new_password": ""
  }
}
```

200 OK

Параметры пользователя изменены

400 BAD_REQUEST

Некорректное тело запроса

404 NOT_FOUND

Пользователь не найден

Удаление пользователя (DELETE /users/{id})

Тела запроса нет. В query-параметрах передается id пользователя, которого необходимо удалить

200 OK

Пользователь удален

404 NOT_FOUND

Пользователь не найден

Получение информации о пользователе (GET /users/{id})

Тела запроса нет. В query-параметрах передается id пользователя, информацию о котором необходимо получить

200 OK

Информация о пользователе получена

404 NOT_FOUND

Пользователь не найден

Получение списка прав пользователя (GET /users/permissions)

200 OK

Получен список прав пользователей СРК с разными ролями

8.3. Сбор сведений

8.3.1. Лицензии

Получение списка типов лицензий (GET /license_types)

200 OK

Получен список типов лицензий

Получение информации о типе лицензии (GET /license_types/{id})

Тела запроса нет. В query-параметрах передается id типа лицензии, информацию о которой необходимо получить

200 OK

Получена информация о типе лицензии

404 NOT_FOUND

Тип лицензии не найден

8.3.2. Информация о сервере

Получение информации о сервере (GET /server_hello)

200 OK

Получена информация о версии rubackup сервера

8.3.3. Сведения о состоянии приложения

Получение информации о состоянии приложения (GET /server_state)

Возвращает список неавторизованных клиентов и медиасерверов, просроченных правил стратегий и глобального расписания, клиентских запросов на добавление и удаление правил

200 OK

Получена информация о текущем состоянии приложения

8.3.4. Типы резервного копирования

Получение списка доступных типов резервного копирования (GET /backup_types)

Тела запроса нет.

200 OK

Получен список типов резервного копирования

8.3.5. Типы сжатия

Получение списка доступных типов сжатия (GET /compression_type)

200 OK

Получен список типов сжатия

8.3.6. Алгоритмы защиты данных

Получение списка доступных криптоалгоритмов (GET /crypto)

200 OK

Получен список доступных криптоалгоритмов

8.3.7. Алгоритмы дедупликации

Получение списка доступных алгоритмов хеш-функций (GET /deduplication_hash_algorithm)

OK 200 - получен список алгоритмов хеш-функций

8.3.8. Поиск медиасерверов

Поиск клиента медиасервера по HWID сервера (GET /servers_hw_id_tmp)

200 OK

Получен список медиасерверов с информацией о них, их HW ID и лицензии

Получение информации о сервере и лицензии (GET /servers_hw_id_tmp/{id})

Тела запроса нет. В query-параметрах передается id медиасервера, информацию о котором необходимо получить

200 OK

Получена информация о сервере и лицензии

404 NOT_FOUND

Информация о медиасервере и лицензии не найдена

8.3.9. Доступные типы ресурсов

Получение списка доступных типов ресурсов (GET `/resource_types`)

200 OK

Получен список типов ресурсов

Получение информации о типе ресурса (GET `/resource_types/{id}`)

Тела запроса нет. В query-параметрах передается id типа ресурса, информацию о котором необходимо получить

200 OK

Получена информация о типе ресурса

404 NOT_FOUND

Тип ресурса не найден

8.3.10. Доступные операционные системы

Получение списка доступных операционных систем (GET `/os_types`)

200 OK

Получен список доступных типов операционных систем

8.3.11. Статусы верификации

Получение списка доступных статусов верификации (GET `/verification_status`)

200 OK

Получен список доступных статусов верификации

8.3.12. Статусы правил

Получение списка доступных статусов правил (GET `/status_of_rule`)

200 OK

Получен список существующих статусов правил

8.3.13. Ресурсы модулей

Получение списка доступных ресурсов модуля (POST `/module_resource_list`)

Например, для PostgreSQL Universal это версия PostgreSQL (например, PostgreSQL 16.3)

```
{
```

```

"data": {
  "client_hwid": "525a99154f3505a2", ①
  "resource_type": "LVM logical volume", ①
  "secret_method": 0
}

```

① Обязательное поле

200 OK

Получен список доступных ресурсов модуля

8.3.14. Метрики (статистика)

Получение статистики по уведомлениям (GET `/metrics/notifications`)

200 OK

Получена статистика уведомлений

Получение статистики по объектам репозитория (GET `/metrics/repository`)

200 OK

Получена статистика репозитория

Получение статистики по очереди задач облаков (GET `/metrics/s3_cloud_task_queue`)

200 OK

Получена статистика очереди задач облаков

Получение статистики по очереди задач (GET `/metrics/task_queue`)

200 OK

Получена статистика очереди задач

Получение статистики по очереди задач ленточных библиотек (GET `/metrics/tl_task_queue`)

200 OK

Получена статистика очереди задач ленточных библиотек

8.4. Глобальная конфигурация

8.4.1. Редактирование параметров глобальной конфигурации (PATCH `/global_configuration`)

```

{
  "data": {

```

```

"file_transfer_block_size": 16384,
"data_spred_into_pool": "sequentially",
"emergency_storage_local_catalog":
"/tmp/rubackup_emergency_storage_local_catalog",
"done_tasks_remove_period": 1440,
"error_tasks_remove_period": 10080,
"broken_tasks_remove_period": 10080,
"killed_tasks_remove_period": 10080,
"obsoleted_backup_notification_period": 1440,
"rule_validity_end_notification_period": 1440,
"strategy_validity_end_notification_period": 1440,
"rule_storage_capacity_reserve": 0,
"strategy_storage_capacity_reserve": 0,
"rule_storage_capacity_notification_period": 1440,
"strategy_storage_capacity_notification_period": 1440,
"service_window_start": "00:00:00",
"service_window_end": "23:59:59",
"pool_storage_capacity_reserve": 2000000000,
"tape_library_mount_point": "/opt/rubackup/mnt",
"suspended_task_restart_period": 1,
"unload_tape_cartridges_whes_media_server_starts": "yes",
"lufs_umount_timeout": 50,
"max_system_monitoring_records": 3600,
"monitoring_period": 1,
"digital_signature_public_key_obsolescence": 1440,
"rb_key_hash": "",
"verify_archive_after_creation": "no",
"kill_task_of_offline_client": "yes",
"create_new_task_if_client_offline": "create new task",
"auto_delete_archive_from_broken_chain": "no",
"consolidating_tape_library_tasks": "yes",
"clean_tape_drives_period": 30,
"auto_clean_tape_drives": "no",
"dedup_clean_unused_blocks": "yes",
"dedup_clean_period": 30,
"dedup_provide_common_hash_table_for_client": "yes",
"dedup_provide_common_hash_table": "yes",
"dedup_verify_only_meta_data": "yes",
"mandatory_storage_time": 0,
"clients_capacity_limits": false,
"global_schedule_capacity_limits": false,
"backup_strategies_capacity_limits": false,
"immutable_archives": true,
"delete_archives_with_zeroing": true,
"dont_delete_last_gs_rule_archive": true,

```

```

    "dont_delete_last_strategy_rule_archive": true,
    "wrong_auth_attempts_count": 5,
    "wrong_auth_block_period": 30,
    "repository_remove_chain": "yes",
    "filesystems_clean_period": 30,
    "show_previous_login_attempts_after_logging_in": "yes",
    "search_backup_in_cluster_group": true,
    "bandwidth_limit_advantage": "client",
    "bandwidth_client_limit": "minimum",
    "bandwidth_rule_limit": "minimum"
  }
}

```

200 OK

Параметры глобальной конфигурации изменены

400 BAD_REQUEST

Некорректное тело запроса

8.4.2. Получение параметров глобальной конфигурации (GET /global_configuration)

200 OK

Параметры глобальной конфигурации получены

8.4.3. Включение и выключение сервисного режима (PATCH /global_configuration/service_mode)

```

{
  "service_mode": false ①
}

```

① Обязательное поле

200 OK

Статус сервисного режима изменен

400 BAD_REQUEST

Некорректное тело запроса

8.5. Клиенты

8.5.1. Клиенты

Получение списка клиентов (GET /clients)**200 OK**

Получен список авторизованных клиентов

Удаление нескольких клиентов (DELETE /clients)

```
{
  "ids": [ ❶
    1
  ]
}
```

❶ Обязательное поле**200 OK**

Список авторизованных клиентов удален

404 NOT_FOUND

Авторизованный клиент не найден

Авторизация клиентов (POST /clients/authorize)

```
{
  "data": {
    "clients_ids": [ ❶
      1
    ]
  }
}
```

❶ Обязательное поле**201 CREATED**

Клиент авторизован

Получение дерева клиентов (POST /clients/tree)

```
{
  "data": {
    "clients": [
      {
        "client_group_id": 1, ❶
        "clients_limit": 20,
        "clients_page": 1 ❶
      }
    ]
  }
}
```

```

    ],
    "client_groups_page": 1,
    "client_groups_limit": 20,
    "filter_clients_hostname": "rubackup-client"
  }
}

```

1 Обязательное поле

200 OK

Получено дерево клиентов

404 NOT_FOUND

Авторизованный клиент не найден

Редактирование существующего клиента (PATCH /clients/{id})

Также в query-параметрах передается id клиента, информацию о котором необходимо изменить

```

{
  "data": {
    "group_id": "No group",
    "lsf_flag": false,
    "ls_pool": "Default",
    "lrl_flag": false,
    "storage_capacity": 10,
    "description": "Client description",
    "centralized_restore": false,
    "client_side_backup": false,
    "client_side_restore": false
  }
}

```

200 OK

Параметры клиента изменены

400 BAD_REQUEST

Некорректное тело запроса

404 NOT_FOUND

Авторизованный клиент не найден

Получение клиента (GET /clients/{id})

Тела запроса нет. В query-параметрах передается id клиента, информацию о кото-

ром необходимо получить

200 OK

Получена информация о клиенте

Удаление клиента (DELETE /clients/{id})

Тела запроса нет. В query-параметрах передается id клиента, информацию о котором необходимо получить

200 OK

Клиент удален (помещен в неавторизованные)

404 NOT_FOUND

Авторизованный клиент не найден

Получение доступных типов ресурсов для создания копии (GET /clients/{id}/resource_type)

Тела запроса нет. В query-параметрах передается id клиента, информацию о типах ресурсов которого необходимо получить

200 OK

Получена информация о типах ресурса для клиента

8.5.2. Группы клиентов

Получение списка групп клиентов (GET /client_groups)

200 OK

Получена информация о группах клиентов

Удаление списка групп клиентов (DELETE /client_groups)

```
{
  "ids": [ ①
    1
  ]
}
```

① Обязательное поле

200 OK

Список групп клиентов удален

404 NOT_FOUND

Группа клиентов не найдена

Добавление групп клиентов (POST /client_groups)

```
{
  "data": {
    "name": "group_name", ①
    "shared": false,
    "clustered": false,
    "description": "some description",
    "launch_attempts_limit": 1
  }
}
```

① Обязательное поле

201 CREATED

Группа клиентов создана

Изменение группы клиентов (PATCH /client_groups/{id})

Также в query-параметрах передается id группы клиентов, информацию о которой необходимо изменить

```
{
  "data": {
    "name": "group_name",
    "shared": false,
    "clustered": false,
    "description": "some description",
    "launch_attempts_limit": 1
  }
}
```

200 OK

Параметры группы клиентов изменены

400 BAD_REQUEST

Некорректное тело запроса

404 NOT_FOUND

Группа клиентов не найдена

Получение информации о группе клиентов (GET /client_groups/{id})

Тела запроса нет. В query-параметрах передается id группы клиентов, информацию о которой необходимо получить

200 OK

Получена информация о группе клиентов

Удаление группы клиентов (DELETE /client_groups/{id})

Тела запроса нет. В query-параметрах передается id группы клиентов, которую необходимо удалить

200 OK

Группа клиентов удалена

404 NOT_FOUND

Группа клиентов не найдена

8.5.3. Ограничения пропускной способности клиента

Получение списка ограничений пропускной способности клиента (GET /client_bandwidth)

200 OK

Получен список ограничений пропускной способности

Удаление списка ограничений пропускной способности клиента (DELETE /client_bandwidth)

```
{
  "ids": [ ❶
    1
  ]
}
```

❶ Обязательное поле

200 OK

Список ограничений пропускной способности удален

404 NOT_FOUND

Ограничение пропускной способности не найдено

Создание ограничения пропускной способности клиента (POST /client_bandwidth)

```
{
  "data": {
    "client": "rubackup-client (525a99154f3505a2)", ❶
    "backup_bandwidth": 0,
    "restore_bandwidth": 0,
    "window_start": "14:33:31",
    "window_end": "14:33:31"
  }
}
```

```

    }
  }

```

1 Обязательное поле

201 CREATED

Ограничение пропускной способности создано

Изменение ограничения пропускной способности клиента (PATCH /client_bandwidth/{id})

Также в query-параметрах передается id ограничения, информацию о котором необходимо изменить

```

{
  "data": {
    "client": "rubackup-client (525a99154f3505a2)",
    "backup_bandwidth": 0,
    "restore_bandwidth": 0,
    "window_start": "14:33:31",
    "window_end": "14:33:31",
    "id": 1,
    "client_hwid": "525a99154f3505a2"
  }
}

```

200 OK

Параметры ограничения пропускной способности изменены

400 BAD_REQUEST

Некорректное тело запроса

404 NOT_FOUND

Ограничение пропускной способности не найдено

Получение информации об ограничении пропускной способности клиента (GET /client_bandwidth/{id})

Тела запроса нет. В query-параметрах передается id ограничения, информацию о котором необходимо получить

200 OK

Получена информация об ограничении пропускной способности

Удаление ограничения пропускной способности клиента (DELETE /client_bandwidth/{id})

Тела запроса нет. В query-параметрах передается id ограничения, которое необ-

ходимо удалить

200 OK

Ограничение пропускной способности удалено

404 NOT_FOUND

Ограничение пропускной способности не найдено

8.5.4. Неавторизованные клиенты

Удаление списка неавторизованных клиентов (DELETE /unauthorised_clients)

```
{
  "ids": [ ❶
    1
  ]
}
```

❶ Обязательное поле

200 OK

Список неавторизованных клиентов удален

404 NOT_FOUND

Неавторизованные клиенты не найдены

Получение списка неавторизованных клиентов (GET /unauthorised_clients)

200 OK

Получен список неавторизованных клиентов

Удаление неавторизованного клиента (DELETE /unauthorised_clients/{id})

Тело запроса отсутствует. В query-параметрах передается id неавторизованного клиента, которого необходимо удалить

200 OK

Неавторизованный клиент удален

404 NOT_FOUND

Неавторизованный клиент не найден

Получение информации о неавторизованном клиенте (GET /unauthorised_clients/{id})

Тело запроса отсутствует. В query-параметрах передается id неавторизованного клиента, информацию о котором необходимо получить

200 OK

Получена информация о неавторизованном клиенте

8.6. Хранилища

8.6.1. Пулы и их группы

Доступные типы пулов

Получение списка доступных типов пулов (GET `/destination_storage_types`)

200 OK

Получен список доступных типов пулов

Пулы

Удаление списка пулов (DELETE `/pool_list`)

```
{
  "ids": [ ❶
    1
  ]
}
```

❶ Обязательное поле

200 OK

Список пулов удален

404 NOT_FOUND

Пул не найден

Получение списка пулов (GET `/pool_list`)

200 OK

Получен список существующих пулов

Добавление пула (POST `/pool_list`)

Набор параметров, которые необходимо передать, зависит от типа пула. В случае, если переданы лишние параметры (которые не подходят создаваемому типу пула), то такие параметры игнорируются.

Параметр `compression_type` не нужно указывать для пулов типа Block device и Client defined.

Параметры `block_size`, `hash_length`, `algorithm` указываются только для пулов типа

Block device.

```
{
  "data": {
    "pool_name": "my_pool_name", ①
    "pool_type": "Block device", ①
    "media_server": "rubackup-server", ①
    "compression_type": "None",
    "description": "",
    "retention_period": "1 year",
    "ignore_free_space_check": false,
    "algorithm": "streebog",
    "block_size": 131072,
    "hash_length": 256,
    "meta_data_pool": "Default"
  }
}
```

① Обязательное поле

201 CREATED

Пул создан

Получение списка доступных для перемещения и копирования пулов (GET /pool_list/available_for_copy_or_move)

200 OK

Получен список доступных для перемещения и копирования пулов

Редактирование пула (PATCH /pool_list/{id})

Также в query-параметрах передается id пула, информацию о котором необходимо изменить

```
{
  "data": {
    "pool_name": "my_pool_name",
    "compression_type": "None",
    "description": "",
    "retention_period": "1 year",
    "ignore_free_space_check": false
  }
}
```

200 OK

Параметры пула изменены

400 BAD_REQUEST

Некорректное тело запроса

404 NOT_FOUND

Пул не найден

Удаление пула (DELETE /pool_list/{id})

Тела запроса нет. В query-параметрах передается id пула, который необходимо удалить

200 OK

Пул удален

404 NOT_FOUND

Пул не найден

Получение информации о пуле (GET /pool_list/{id})

Тела запроса нет. В query-параметрах передается id пула, информацию о котором необходимо получить

200 OK

Получена информация о пуле

Подмена пулов**Удаление списка подмены пулов (DELETE /pool_substitution)**

```
{
  "ids": [ ①
    1
  ]
}
```

① Обязательное поле

200 OK

Список настроенных подмен пулов удален

404 NOT_FOUND

Подмена пулов не найдена

Получение списка подмены пулов (GET /pool_substitution)**200 OK**

Получен список настроенных подмен пулов

Добавление подмены пулов (POST /pool_substitution)

```
{
  "data": {
    "pool": "Default", ①
    "substitution": "Default2" ①
  }
}
```

① Обязательное поле**201 CREATED**

Создано правило подмены пулов

Удаление подмены пулов (DELETE /pool_substitution/{id})

Тела запроса нет. В query-параметрах передается id правила подмены пулов, которое необходимо удалить

200 OK

Правило подмены пулов удалено

404 NOT_FOUND

Подмена пулов не найдена

Получение информации о подмене пулов (GET /pool_substitution/{id})

Тела запроса нет. В query-параметрах передается id правила подмены пулов, информацию о котором необходимо получить

200 OK

Получена информация о правило подмены пулов

Динамические группы пулов**Получение списка динамических групп пулов (GET /dynamic_pool_groups)****200 OK**

Получен список групп пулов

Удаление списка динамических групп пулов (DELETE /dynamic_pool_groups)

```
{
```

```

"ids": [ ❶
  1
]
}

```

❶ Обязательное поле

200 OK

Группы пулов удалены

404 NOT_FOUND

Группа пулов не найдена

Добавление динамической группы пулов (POST /dynamic_pool_groups)

```

{
  "data": {
    "name": "", ❶
    "max_pool_tasks": 10,
    "max_media_server_tasks": 10,
    "max_cpu_usage": 80,
    "max_in_block_operations": 0,
    "max_out_block_operations": 0,
    "calculation_period": 10,
    "description": ""
  }
}

```

❶ Обязательное поле

201 CREATED

Группа пулов создана

Изменение динамической группы пулов (PATCH /dynamic_pool_groups/{id})

Также в query-параметрах передается id группы пулов, информацию о которой необходимо изменить

```

{
  "data": {
    "name": "",
    "max_pool_tasks": 10,
    "max_media_server_tasks": 10,
    "max_cpu_usage": 80,
    "max_in_block_operations": 0,
    "max_out_block_operations": 0,
    "calculation_period": 10,

```

```

    "description": ""
  }
}

```

200 OK

Группа пулов изменена

400 BAD_REQUEST

Некорректное тело запроса

404 NOT_FOUND

Группа пулов не найдена

Получение информации о динамической группе пулов (GET /dynamic_pool_groups/{id})

Тела запроса нет. В query-параметрах передается id группы пулов, информацию о которой необходимо получить

200 OK

Получена информация о группе пулов

Удаление динамической группы пулов (DELETE /dynamic_pool_groups/{id})

Тела запроса нет. В query-параметрах передается id группы пулов, информацию о которой необходимо удалить

200 OK

Стратегия удалена

404 NOT_FOUND

Группа пулов не найдена

Пулы в динамических группах**Получение списка добавленных в динамическую группу пулов (GET /dynamic_pool_groups_records)****200 OK**

Получен список пулов динамической группы

Удаление списка добавленных в динамическую группу пулов (DELETE /dynamic_pool_groups_records)

```

{
  "ids": [ 1 ]
}

```

1 Обязательное поле**200 OK**

Список пулов динамической группы удален

404 NOT_FOUND

Пулов динамической группы не найдено

Добавление пула в динамическую группу пулов (POST /dynamic_pool_groups_records)

```
{
  "data": {
    "dynamic_pool_group": 1, ①
    "pool": "Default", ①
    "enabled": true
  }
}
```

1 Обязательное поле**201 CREATED**

Пул добавлен в группу

Получение информации о добавленном в динамическую группу пуле (GET /dynamic_pool_groups_records/{id})

Тела запроса нет. В query-параметрах передается id пула из группы пулов, информацию о котором необходимо получить

200 OK

Получена информация о пуле динамической группы

Удаление добавленного в динамическую группу пула (DELETE /dynamic_pool_groups_records/{id})

Тела запроса нет. В query-параметрах передается id пула, который необходимо удалить из группы пулов

200 OK

Пул удален из динамической группы

404 NOT_FOUND

Пул в динамической группе не найден

8.6.2. Локальные файловые хранилища**Удаление нескольких локальных файловых хранилищ (DELETE /storage_local_catalogs)**

```
{
```

```
"ids": [ ❶
  1
]
```

❶ Обязательное поле

200 OK

Список локальных файловых хранилищ удален

404 NOT_FOUND

Локальный файловый хранилище не найдено

Получение списка локальных файловых хранилищ (GET /storage_local_catalogs)

200 OK

Получен список локальных файловых хранилищ

Добавление локального файлового хранилища (POST /storage_local_catalogs)

```
{
  "data": {
    "path": "", ❶
    "pool": "Default",
    "description": ""
  }
}
```

❶ Обязательное поле

201 CREATED

Локальное файловое хранилище добавлено

Редактирование существующего локального файлового хранилища (PATCH /storage_local_catalogs/{id})

Также в query-параметрах передается id локального файлового хранилища, который необходимо изменить

```
{
  "data": {
    "description": ""
  }
}
```

200 OK

Параметры локального файлового хранилища изменены

400 BAD_REQUEST

Некорректное тело запроса

404 NOT_FOUND

Локальное файловое хранилище не найдено

Удаление локального файлового хранилища (DELETE /storage_local_catalogs/{id})

Также в query-параметрах передается id локального файлового хранилища, которое необходимо удалить

200 OK

Блочное устройство удалено

404 NOT_FOUND

Локальное файловое хранилище не найдено

Получение информации о локальном файловом хранилище (GET /storage_local_catalogs/{id})

Также в query-параметрах передается id локального файлового хранилища, информацию о котором необходимо получить

200 OK

Информация о локальном файловом хранилище получена

404 NOT_FOUND

Локальное файловое хранилище не найдено

8.6.3. Блочные устройства

Удаление списка блочных устройств (DELETE /storage_block_devices)

```
{
  "ids": [ ❶
    1
  ]
}
```

❶ Обязательное поле

200 OK

Список блочных устройств удален

404 NOT_FOUND

Блочное устройство не найдено

Получение списка блочных устройств (GET /storage_block_devices)**200 OK**

Получен список блочных устройств

Добавление блочного устройства (POST /storage_block_devices)

```
{
  "data": {
    "device": "", ①
    "pool": "Default", ①
    "overwrite_file_system": false, ①
    "description": ""
  }
}
```

① Обязательное поле

201 CREATED

Блочное устройство добавлено

Редактирование блочного устройства (PATCH /storage_block_devices/{id})

Также в query-параметрах передается id блочного устройства, которое необходимо изменить

```
{
  "data": {
    "description": "",
    "parallel_workers": 2
  }
}
```

200 OK

Параметры блочного устройства изменены

400 BAD_REQUEST

Некорректное тело запроса

404 NOT_FOUND

Блочное устройство не найдено

Удаление блочного устройства (DELETE /storage_block_devices/{id})

Также в query-параметрах передается id блочного устройства, которое необходимо удалить

200 OK

Блочное устройство удалено

404 NOT_FOUND

Блочное устройство не найдено

Получение информации о блочном устройстве (GET /storage_block_devices/{id})

Также в query-параметрах передается id блочного устройства, информацию о котором необходимо получить+

200 OK

Информация о блочном устройстве получена

404 NOT_FOUND

Блочное устройство не найдено

8.6.4. Ленточные хранилища

Ленточные библиотеки

Удаление списка ленточных библиотек (DELETE /tape_libraries)

```
{
  "ids": [ ①
    1
  ]
}
```

① Обязательное поле

200 OK

Список ленточных библиотек удален

404 NOT_FOUND

Ленточная библиотека не найдена

Получение списка ленточных библиотек (GET /tape_libraries)**200 OK**

Получен список ленточных библиотек

Добавление ленточной библиотеки (POST /tape_libraries)

```
{
  "data": {
    "pool": "tape_pool", ①
    "media_server": "rubackup-media", ①
    "description": "",
    "medium_changer_candidate_id": 1, ①
    "tape_drives_candidates_ids": [ ①
      1
    ]
  }
}
```

① Обязательное поле

201 CREATED

Ленточная библиотека добавлена

Удаление ленточной библиотеки (DELETE /tape_libraries/{id})

Тела запроса нет. В query-параметрах передается id ленточной библиотеки, информацию о которой необходимо получить

200 OK

Ленточная библиотека удалена

404 NOT_FOUND

Ленточная библиотека не найдена

Получение информации о ленточной библиотеке (GET /tape_libraries/{id})

200 OK

Информация о ленточной библиотеке получена

404 NOT_FOUND

Ленточная библиотека не найдена

Получение списка доступных для добавления роботов и приводов (GET /tape_libraries/candidates)

Тела запроса нет. В query-параметрах передается имя медиасервера, которому презентована библиотека, а также способ добавления

200 OK

Получен список доступных для добавления роботов и приводов

Получение информации о роботе и слотах (POST /tape_libraries/candidates/info)

```
{
  "data": {
    "media_server": "rubackup-media-server", ①
    "medium_changer_sg_path": "/dev/sg4"
  }
}
```

200 OK

Получена информация о роботе и слотах

Получение дерева ленточных библиотек (POST /tape_libraries/tree)

В списке отображаются: 1. Ленточные библиотеки, их роботы, их приводы. Список слотов должен быть пустым всегда. 2. Ленточные пулы, ассоциированные и неассоциированные с библиотекой

```
{
  "data": {
    "pools": {
      "limit": 50,
      "page": 1
    }
  }
}
```

200 OK

Получено дерево информация о роботе и слотах

Синхронизация данных ленточной библиотеки (POST /tape_libraries/{id}/sync)

Тела запроса нет. В query-параметрах передается id библиотеки, информацию о которой необходимо синхронизировать

200 OK

Синхронизация библиотеки прошла успешно

422 Unprocessable Entity - синхронизация завершена с ошибкой, библиотека содержит незарегистрированный картридж

Типы картриджей

Получение списка доступных типов картриджей (GET /tape_types)**200 OK**

Получен список доступных типов картриджей

Картриджи**Удаление списка картриджей (DELETE /tape_cartridges)**

```
{
  "ids": [
    1
  ]
}
```

200 OK

Список картриджей удален

404 NOT_FOUND

Картридж не найден

Получение списка доступных картриджей (GET /tape_cartridges)**200 OK**

Получен список доступных картриджей

Добавление ленточного картриджа (POST /tape_cartridges)

```
{
  "data": {
    "type": "LTO-9", ①
    "pool": "tape_pool", ①
    "volume_tag": "RB10001L9", ①
    "description": ""
  }
}
```

① Обязательное поле

201 CREATED

Ленточный картридж добавлен

Редактирование картриджа (PATCH /tape_cartridges/{id})

Также в query-параметрах передается id ленточного картриджа, информацию о которой необходимо изменить

```
{
  "data": {
    "volume_tag": "",
    "description": ""
  }
}
```

200 OK

Параметры картриджа изменены

400 BAD_REQUEST

Некорректное тело запроса

404 NOT_FOUND

Картридж не найден

Получение информации о ленточном картридже (GET /tape_cartridges/{id})

Тела запроса нет. В query-параметрах передается id картриджа, информацию о котором необходимо получить

200 OK

Получена информация о ленточном картридже

Операции с ленточными приводами**Получение списка ленточных приводов (GET /library_tape_drives)****200 OK**

Получен список приводов ленточной библиотеки

Получение информации о ленточном приводе (GET /library_tape_drives/{id})

Тела запроса нет. В query-параметрах передается id привода, информацию о которой необходимо получить

200 OK

Получена информация о приводе ленточной библиотеки

Добавление привода (POST /library_tape_drives)

```
{
  "data": {
    "ids": [ ①
      4,
      6,
```

```

    9
    ],
    "tape_library_id": 5 ①
  }
}

```

① Обязательное поле

201 CREATED

Новый привод добавлен к библиотеке

Удаление списка ленточных приводов (DELETE /library_tape_drives)

```

{
  "ids": [ ①
    2,
    6,
    9
  ]
}

```

① Обязательное поле

200 OK

Привод ленточной библиотеки удален

Получение списка ленточных приводов, доступных для добавления (GET /library_tape_drives/candidates)

Тела запроса нет. В query-параметрах передается способ добавления привода (ручной или автоматический), hostname медиасервера, на который презентована библиотека и sg_device_path робота библиотеки

200 OK

Информация о доступных для добавления приводах получена

Редактирование ленточного привода (PATCH /library_tape_drives/{id})

Также в query-параметрах передается id привода, информацию о котором необходимо изменить

```

{
  "data": {
    "scsi_path": "[6:0:0:0]",
    "device": "/dev/sg1",
    "sys_transfer_element": 3
  }
}

```

```
}
```

200 OK

Параметры привода изменены

400 BAD_REQUEST

Некорректное тело запроса

Очистка ленточного привода (PATCH /library_tape_drives/{id}/clean)

Тела запроса нет. В query-параметрах передается id привода, который необходимо очистить

200 OK

Задача на очистку привода создана

Картриджи**Получение списка слотов (GET /library_slots)****200 OK**

Получен список слотов ленточной библиотеки

Получение информации о слоте (GET /library_slots/{id})

Тела запроса нет. В query-параметрах передается id слота, информацию о которой необходимо получить

200 OK

Получена информация о слоте ленточной библиотеки

Проверка ltfs на картридже (PATCH /library_slots/check_ltfs)

```
{
  "data": [
    {
      "library_id": 5, ①
      "volume_tag": "ACS512L9" ①
    }
  ]
}
```

① Обязательное поле

200 OK

Создана задача на проверку наличия ltfs на картридже

400 BAD_REQUEST

Некорректное тело запроса

Стирание картриджа (PATCH /library_slots/erase_cartridge)

```
{
  "data": [
    {
      "library_id": 5, ❶
      "is_long_erase": false,
      "volume_tag": "ACS512L9" ❶
    }
  ]
}
```

❶ Обязательное поле

200 OK

Создана задача на стирание картриджа

400 BAD_REQUEST

Некорректное тело запроса

Экспорт картриджа (PATCH /library_slots/export_cartridge)

```
{
  "data": {
    "library_id": 5, ❶
    "volume_tag": "ACS512L9" ❶
  }
}
```

❶ Обязательное поле

200 OK

Создана задача на экспорт картриджа

400 BAD_REQUEST

Некорректное тело запроса

Импорт картриджа (PATCH /library_slots/import_cartridge)

```
{
  "data": {
    "library_id": 5, ❶
    "pool_name": "Tape libraries" ❶
  }
}
```

```
}  
}
```

1 Обязательное поле**200 OK**

Импорт картриджа прошел успешно

400 BAD_REQUEST

Некорректное тело запроса

Форматирование картриджа (PATCH /library_slots/format_cartridge)

```
{  
  "data": [  
    {  
      "library_id": 5, 1  
      "volume_tag": "ACS512L9" 1  
    }  
  ]  
}
```

1 Обязательное поле**200 OK**

Создана задача на форматирование картриджа

400 BAD_REQUEST

Некорректное тело запроса

Перемещение картриджа в другой слот (PATCH /library_slots/move_cartridge)

```
{  
  "data": {  
    "library_id": 5, 1  
    "volume_tag": "ACS512L9", 1  
    "slot_id": 6 1  
  }  
}
```

1 Обязательное поле**200 OK**

Создана задача на перемещение картриджа

400 BAD_REQUEST

Некорректное тело запроса

Роботы

Получение списка роботов (GET `/medium_changers`)

200 OK

Получен список роботов

Получение информации о роботе (GET `/medium_changers/{id}`)

Тела запроса нет. В query-параметрах передается id робота, информацию о которой необходимо получить

200 OK

Получена информация о роботе

404 NOT_FOUND

Робот не найден

Изменение робота (PATCH `/medium_changers/{id}`)

Также в query-параметрах передается id робота, информацию о котором необходимо изменить

```
{
  "data": {
    "scsi_path": "[8:0:0:0]",
    "device": "/dev/sg2"
  }
}
```

200 OK

Параметры робота изменены

400 BAD_REQUEST

Некорректное тело запроса

404 NOT_FOUND

Робот не найдена

8.6.5. Облачные хранилища

Облачное хранилище

Удаление списка облаков (DELETE `/s3_clouds`)

204 NO_CONTENT

удален список облаков

Получение списка доступных облаков (GET /s3_clouds)**200 OK**

получен список облаков

Добавление нового облака (POST /s3_clouds)

```
{
  "data": {
    "name": "my_s3_cloud", ①
    "pool": "cloud_pool", ①
    "access_key_id": "", ①
    "secret_key_access": "", ①
    "region": "ru-1",
    "endpoint_override": "https://192.168.222.197:9000",
    "proxy_host": "",
    "proxy_port": "",
    "proxy_username": "",
    "proxy_password": "",
    "bucket_name": "rubakup-server-5953bb5c71fc3bf8",
    "allow_auto_bucket_creation": true,
    "transfer_workers_number": 1
  }
}
```

① Обязательное поле**201 CREATED**

облако создано

Получение списка доступных бакетов не добавленного в СРК облака (POST /s3_clouds/buckets)

```
{
  "data": {
    "access_key_id": "jkepw6ubWUJpS04vtmgm", ①
    "secret_key_access": "R0hj0Gn7MjJ3iTvBBxw9dI1ez9jPd1wDkMuIQ4FU", ①
    "region": "ru-1", ①
    "endpoint_override": "https://192.168.222.197:9000", ①
    "proxy_host": "",
    "proxy_port": "",
    "proxy_username": "",
    "proxy_password": ""
  }
}
```

① Обязательное поле

201 CREATED

облако создано

Удаление облака DELETE `/s3_clouds/{id}`

Тела запроса нет. В query-параметрах передается id облака, которое необходимо удалить

204 NO_CONTENT

удалено облако

Получение информации о существующем облаке (GET `/s3_clouds/{id}` **)**

Тела запроса нет. В query-параметрах передается id облака, информацию о котором необходимо получить

200 OK

получена информация об облаке

404 NOT_FOUND

облако не найдено

Редактирование параметров облака (PATCH `/s3_clouds/{id}` **)**

В query-параметрах передается id облака, информацию о котором необходимо изменить

```
{
  "data": {
    "allow_auto_bucket_creation": false,
    "bucket_name": ""
  }
}
```

200 OK

параметры облака изменены

404 NOT_FOUND

облако не найдено

Получение списка доступных бакетов существующего облака (GET `/s3_clouds/{id}/buckets` **)**

В query-параметрах передается id облака, информацию о доступных бакетах которого необходимо получить

200 OK

получена информация о доступных бакетах существующего облака

Статус задач облачного хранилища

Получение списка доступных статусов задач на облачные хранилища (GET /s3_clouds_task_status)

200 OK

получен список статусов очереди облаков

Задачи облачного хранилища

Получение задач очереди облаков (GET /s3_clouds_task_queue)

200 OK

Получен список задач в очереди задач облаков

Получение информации о задаче из очереди облаков (GET /s3_clouds_task_queue/{id})

200 OK

Получена информация о задаче в очереди задач облаков

404 NOT_FOUND

Задача не найдена

Удаление списка задач из очереди облаков (DELETE /s3_clouds_task_queue)

```
{
  "ids": [ ①
    1
  ]
}
```

① Обязательное поле

204 NO_CONTENT

Удален список задач из очереди задач облаков

Удаление ошибочных задач из очереди облаков (DELETE /s3_clouds_task_queue/errors_deletion)

204 NO_CONTENT

Удален список ошибочных задач из очереди задач облаков

Удаление задачи из очереди облаков (DELETE /s3_clouds_task_queue/{id})

204 NO_CONTENT

Удалена задача из очереди задач облаков

404 NOT_FOUND

Задача не найдена

8.6.6. Клиентские хранилища

Получение списка клиентских хранилищ (GET /client_defined_storages)

200 OK

Получен список клиентских хранилищ

Добавление клиентского хранилища (POST /client_defined_storages)

```
{
  "data": {
    "pool": "Default", ①
    "metadata_fs_pool": "Default2", ①
    "name": "", ①
    "description": ""
  }
}
```

① Обязательное поле

201 CREATED

Клиентское хранилище создано

Удаление списка клиентских хранилищ (DELETE /client_defined_storages)

```
{
  "ids": [ ①
    1
  ]
}
```

① Обязательное поле

200 OK

Список клиентских хранилищ удален

404 NOT_FOUND

Клиентское хранилище не найдено

Изменение клиентского хранилища (PATCH /client_defined_storages/{id})

Также в query-параметрах передается id клиентского хранилища, информацию о котором необходимо изменить

```
{
  "data": {
    "name": "",
    "metadata_fs_pool": "Default2",
    "description": ""
  }
}
```

200 OK

Параметры клиентского хранилища изменены

400 BAD_REQUEST

Некорректное тело запроса

404 NOT_FOUND

Клиентское хранилище не найдено

Получение информации о клиентском хранилище (GET /client_defined_storages/{id})

Тела запроса нет. В query-параметрах передается id клиентского хранилища, информацию о котором необходимо получить

200 OK

Получена информация об клиентском хранилище

404 NOT_FOUND

Клиентское хранилище не найдено

Удаление клиентского хранилища (DELETE /client_defined_storages/{id})

Тела запроса нет. В query-параметрах передается id клиентского хранилища, информацию о котором необходимо удалить

200 OK

Клиентское хранилище удалено

404 NOT_FOUND

Клиентское хранилище не найдено

8.6.7. Клиентские блочные устройства

Получение списка доступных блочных устройств указанного клиента (POST /list_client_block_devices)

```
{
```

```

"data": {
  "client_hwid": "525a99154f3505a2", ❶
  "path": "/dev/" ❶
}
}

```

❶ Обязательное поле

200 OK

Получен список доступных блочных устройств клиента

8.6.8. Клиентские файловые системы

Получение списка доступных директорий указанного клиента (POST `/list_client_filesystem`)

```

{
  "data": {
    "client_hwid": "525a99154f3505a2", ❶
    "path": "/" ❶
  }
}

```

❶ Обязательное поле

200 OK

Получен список доступных директорий клиента

8.6.9. Медиасерверы

Удаление нескольких медиасерверов (DELETE `/media_servers`)

```

{
  "ids": [ ❶
    1
  ]
}

```

❶ Обязательное поле

200 OK

Список медиасерверов удален

404 NOT_FOUND

Медиасервер не найден

Получение списка медиасерверов (GET /media_servers)**200 OK**

Получен список медиасерверов

Авторизация медиасервера (POST /media_servers/authorize)

```
{
  "data": {
    "hostname": "my_hostname" 1
  }
}
```

1 Обязательное поле**201 CREATED**

Медиасервер авторизован

Получение дерева медиасерверов (GET /media_servers/tree)

В списке отображаются медиасерверы, с пулами, медиасерверы, у которых нет пула. Также в дереве отображаются все пулы медиасерверов и ассоциированные с ними хранилища (директории, блочные устройства, ленточные картриджи) и пулы без ассоциированных устройств

200 OK

Получено дерево медиасерверов

Получение списка медиасерверов, имеющих пул типа ленточная библиотека (GET /media_servers/filtered_by_tape_library_pool)**200 OK**

Получено дерево медиасерверов

Редактирование медиасервера (PATCH /media_servers/{id})

Также в query-параметрах передается id медиасервера, информацию о котором необходимо изменить

```
{
  "data": {
    "description": "",
    "local_service_mode": "no"
  }
}
```

200 OK

Параметры медиасервера изменены

400 BAD_REQUEST

Некорректное тело запроса

404 NOT_FOUND

Медиасервер не найден

Удаление медиасервера (DELETE /media_servers/{id})

Тела запроса нет. В query-параметрах передается id медиасервера, который необходимо удалить

200 OK

Медиасервер удален

404 NOT_FOUND

Медиасервер не найден

Получение информации о медиасервере (GET /media_servers/{id})

Тела запроса нет. В query-параметрах передается id медиасервера, информацию о котором необходимо получить

200 OK

Получена информация о медиасервере

8.6.10. Неавторизованные медиасерверы**Удаление списка неавторизованных медиасерверов (DELETE /unauthorised_media_servers)**

```
{
  "ids": [ ❶
    1
  ]
}
```

❶ Обязательное поле

200 OK

Список неавторизованных медиасерверов удален

404 NOT_FOUND

Неавторизованные медиасерверы не найдены

Получение списка неавторизованных медиасерверов (GET /unauthorised_media_servers)**200 OK**

Получен список неавторизованных медиасерверов

Удаление неавторизованного медиасервера (DELETE /unauthorised_media_servers/{id})

Тело запроса отсутствует. В query-параметрах передается id неавторизованного медиасервера, которого необходимо удалить

200 OK

Неавторизованный медиасервер удален

404 NOT_FOUND

Неавторизованный медиасервер не найден

Получение информации о неавторизованном медиасервере (GET /unauthorised_media_servers/{id})

Тело запроса отсутствует. В query-параметрах передается id неавторизованного медиасервера, информацию о котором необходимо получить

200 OK

Получена информация о неавторизованном медиасервере

8.6.11. Репозиторий

Получение списка объектов репозитория (GET /repository)**200 OK**

Получен список сделанных резервных копий

Удаление списка резервных копий из репозитория (DELETE /repository)

```
{
  "ids": [ ①
    1
  ]
}
```

① Обязательное поле

200 OK

Список резервных копий удален

404 NOT_FOUND

Резервная копия не найдена

Получение списка зависимостей для нескольких цепочек (POST `repository/deletion_reference_chain`)

```
{
  "ids": [ ❶
    1
  ]
}
```

❶ Обязательное поле
200 OK

Получен список зависимостей для цепочки резервных копий

Получение дополнительных параметров восстановления (GET `/repository/extension`)

200 OK

Получен список дополнительных параметров восстановления

Получение информации об объекте репозитория (GET `/repository/{id}`)

200 OK

Получена информация о резервной копии

Удаление резервной копии из репозитория (DELETE `/repository/{id}`)

200 OK

Резервная копия удалена

404 NOT_FOUND

Резервная копия не найдена

Изменение периода хранения резервной копии (PATCH `/repository/{id}`)

```
{
  "data": {
    "store_until": "2024-09-07T22:06:09.801038"
  }
}
```

200 OK

Период хранения резервной копии изменен

400 BAD_REQUEST

Некорректное тело запроса

404 NOT_FOUND

Резервная копия не найдена

Копирование полной резервной копии (POST /repository/{id}/copying)

Тела запроса нет. В query-параметрах передается id резервной копии и названия пула, в который необходимо провести копирование

201 CREATED

Задача на копирование резервной копии создана

404 NOT_FOUND

Резервная копия не найдена

Перемещение полной резервной копии (POST /repository/{id}/moving)

Тела запроса нет. В query-параметрах передается id резервной копии и названия пула, в который необходимо провести перемещение

201 CREATED

Задача на перемещение резервной копии создана

404 NOT_FOUND

Резервная копия не найдена

Восстановление резервной копии (POST /repository/{id}/restoring)

Также в query-параметрах передается id резервной копии, которую необходимо восстановить. Для цепочек копий передается id последней разностной копии

```
{
  "data": {
    "restore_task": {
      "target_client": "rsafin-primary (55e05458152af100)", ①
      "destination_path": "/for_restore/", ①
      "data_deployment": false,
      "granular_restore_files": [
        ""
      ]
    },
    "module_extensions": {
      "parameters": {
        "new_name": ""
      }
    },
    "common_modules_extensions": {
```

```

    "parameters": {
      "worker_parallelism": 8,
      "memory_threshold": 0
    }
  }
}
}

```

1 Обязательное поле

201 CREATED

Задача на восстановление резервной копии создана

400 BAD_REQUEST

Некорректное тело запроса

404 NOT_FOUND

Резервная копия не найдена

Верификация резервной копии (POST /repository/{id}/verification)

Тела запроса нет. В query-параметрах передается id резервной копии, которую необходимо верифицировать

201 CREATED

Задача на верификацию резервной копии создана

404 NOT_FOUND

Резервная копия не найдена

8.6.12. Содержимое копии для гранулярного восстановления

Получение списка ресурсов, включенных в копию для гранулярного восстановления (GET /repository_record_file_list)

Тела запроса нет. В query-параметрах передается id резервной копии, список файлов которой необходимо получить

200 OK

Получен список ресурсов, включенных в копию для гранулярного восстановления

8.7. Стратегии

8.7.1. Стратегии резервного копирования

Изменение статуса стратегии (PATCH /backup_strategies)

```
{
  "data": [
    {
      "strategy_id": 1, ❶
      "status": "wait" ❶
    }
  ]
}
```

❶ Обязательное поле

200 OK

Статус стратегии изменен

400 BAD_REQUEST

Некорректное тело запроса

404 NOT_FOUND

Стратегия не найдена

Получение списка стратегий (GET /backup_strategies)

200 OK

Получен список стратегий

Удаление списка стратегий (DELETE /backup_strategies)

```
{
  "ids": [ ❶
    1
  ]
}
```

❶ Обязательное поле

200 OK

Стратегии удалены

404 NOT_FOUND

Стратегия не найдена

Создание стратегии (POST /backup_strategies)

```
{
```

```

"data": {
  "name": "my_backup_strategy", ①
  "status": "wait",
  "pool_id": "Default",
  "crypto": "nocrypt",
  "storage_capacity": 50,
  "description": "some text about backup strategy",
  "validity_start_period": "2024-08-13T14:33:31.359954",
  "validity_end_period": "2025-08-13T14:33:31.360035",
  "verify_flag": true,
  "verify_interval": "1 day",
  "auto_delete_obsoleted_copy_flag": false,
  "inform_when_obsoleted_copy": "Nobody",
  "client_delete_flag": true,
  "full_archive_enabled": false,
  "full_periodic_launch": "1 min",
  "full_storage_duration": "1 day",
  "full_min": 0,
  "full_hour": 0,
  "full_dom": 1,
  "full_mon": 1,
  "full_dow": 1,
  "full_move_copy_flag": false,
  "full_move_copy_while": "1 day",
  "full_move_copy_pool": "Default",
  "inc_archive_enabled": false,
  "inc_periodic_launch": "1 min",
  "inc_storage_duration": "1 day",
  "inc_min": 0,
  "inc_hour": 0,
  "inc_dom": 1,
  "inc_mon": 1,
  "inc_dow": 1,
  "inc_move_copy_flag": false,
  "inc_move_copy_pool": "Default",
  "inc_move_copy_while": "1 day",
  "diff_archive_enabled": false,
  "diff_periodic_launch": "1 min",
  "diff_storage_duration": "1 day",
  "diff_min": 0,
  "diff_hour": 0,
  "diff_dom": 1,
  "diff_mon": 1,
  "diff_dow": 1,
  "diff_move_copy_flag": false,

```

```

"diff_move_copy_pool": "Default",
"diff_move_copy_while": "1 day",
"notify_normal": "Nobody",
"notify_normal_cc": "email@http://domain.ru[domain.ru]",
"notify_exception": "Nobody",
"notify_exception_cc": "email@http://domain.ru[domain.ru]",
"notify_verify": "Nobody",
"notify_verify_cc": "email@http://domain.ru[domain.ru]",
"notify_strategy_validity": "Nobody",
"notify_strategy_validity_cc": "email@http://domain.ru[domain.ru]",
"notify_end_storage_capacity": "Nobody",
"notify_end_storage_capacity_cc": "email@http://domain.ru[domain.ru]"
}
}

```

1 Обязательное поле

201 CREATED

Стратегия создана

Изменение параметров стратегии (PATCH /backup_strategies/{id})

Также в query-параметрах передается id стратегии, информацию о которой необходимо изменить.

```

{
  "data": {
    "name": "my_backup_strategy",
    "status": "wait",
    "pool_id": "Default",
    "crypto": "nocrypt",
    "storage_capacity": 50,
    "description": "some text about backup strategy",
    "validity_start_period": "2024-08-13T14:33:31.359954",
    "validity_end_period": "2025-08-13T14:33:31.360035",
    "verify_flag": true,
    "verify_interval": "1 day",
    "auto_delete_obsoleted_copy_flag": false,
    "inform_when_obsoleted_copy": "Nobody",
    "client_delete_flag": true,
    "full_archive_enabled": false,
    "full_periodic_launch": "1 min",
    "full_storage_duration": "1 day",
    "full_min": 0,
    "full_hour": 0,
    "full_dom": 1,
  }
}

```

```

"full_mon": 1,
"full_dow": 1,
"full_move_copy_flag": false,
"full_move_copy_while": "1 day",
"full_move_copy_pool": "Default",
"inc_archive_enabled": false,
"inc_periodic_launch": "1 min",
"inc_storage_duration": "1 day",
"inc_min": 0,
"inc_hour": 0,
"inc_dom": 1,
"inc_mon": 1,
"inc_dow": 1,
"inc_move_copy_flag": false,
"inc_move_copy_pool": "Default",
"inc_move_copy_while": "1 day",
"diff_archive_enabled": false,
"diff_periodic_launch": "1 min",
"diff_storage_duration": "1 day",
"diff_min": 0,
"diff_hour": 0,
"diff_dom": 1,
"diff_mon": 1,
"diff_dow": 1,
"diff_move_copy_flag": false,
"diff_move_copy_pool": "Default",
"diff_move_copy_while": "1 day",
"notify_normal": "Nobody",
"notify_normal_cc": "email@http://domain.ru[domain.ru]",
"notify_exception": "Nobody",
"notify_exception_cc": "email@http://domain.ru[domain.ru]",
"notify_verify": "Nobody",
"notify_verify_cc": "email@http://domain.ru[domain.ru]",
"notify_strategy_validity": "Nobody",
"notify_strategy_validity_cc": "email@http://domain.ru[domain.ru]",
"notify_end_storage_capacity": "Nobody",
"notify_end_storage_capacity_cc": "email@http://domain.ru[domain.ru]"
}
}

```

200 OK

Стратегия изменена

400 BAD_REQUEST

Некорректное тело запроса

404 NOT_FOUND

Стратегия не найдена

Получение информации о стратегии (GET /backup_strategies/{id})

Тела запроса нет. В query-параметрах передается id стратегии, информацию о которой необходимо получить

200 OK

Получена информация о стратегии

Удалении одной стратегии (DELETE /backup_strategies/{id})

Тела запроса нет. В query-параметрах передается id стратегии, которую необходимо удалить

200 OK

Стратегия удалена

404 NOT_FOUND

Стратегия не найдена

8.7.2. Копирование в пул по стратегиям

Получение списка настроенного копирования резервных копий по стратегиям в пул (GET /backup_strategies_replication)**200 OK**

Получен список правил копирования

Удаление списка настроенного копирования резервных копий по стратегиям (DELETE /backup_strategies_replication)

```
{
  "ids": [ ❶
    1
  ]
}
```

❶ Обязательное поле

200 OK

Правило копирования удалено

404 NOT_FOUND

Правило копирования не найдено

Создание правила копирования резервных копий по стратегиям в пул (POST /backup_strategies_replication)

```
{
  "data": {
    "client": "rubackup-client(525a99154f3505a2)", ❶
    "backup_bandwidth": 0,
    "restore_bandwidth": 0,
    "window_start": "18:10:20",
    "window_end": "18:10:20"
  }
}
```

❶ Обязательное поле

201 CREATED

Правило копирования создано

Получение информации о копировании резервных копий по стратегии в пул (GET /backup_strategies_replication/{id})

Тела запроса нет. В query-параметрах передается id правила копирования, которое необходимо удалить

200 OK

Получена информация о правиле копирования

Удаление настроенного копирования резервных копий по стратегиям (DELETE backup_strategies_replication/{id})

Тела запроса нет. В query-параметрах передается id правила копирования, информацию о котором необходимо получить

200 OK

Правило копирования удалено

404 NOT_FOUND

Правило копирования не найдено

Изменение параметров правила копирования резервных копий по стратегии в пул (PATCH backup_strategies_replication/{id})

Также в query-параметрах передается id правила копирования, информацию о котором необходимо изменить

```
{
  "data": {
    "backup_bandwidth": 0,
    "restore_bandwidth": 0,
    "window_start": "18:10:20",
    "window_end": "18:10:20"
  }
}
```

200 OK

Правило копирования удалено

404 NOT_FOUND

Правило копирования не найдено

8.7.3. Правила стратегий

Удаление списка правил стратегии (DELETE /strategy_rules)

```
{
  "ids": [ ①
    1
  ]
}
```

① Обязательное поле

200 OK

Список правил стратегии удален

404 NOT_FOUND

Правило стратегии не найдено

Получение списка правил стратегий (GET /strategy_rules)

200 OK

Получен список правил стратегии

Добавление правила в стратегию (POST /strategy_rules)

```
{
  "data": {
    "strategy_rule": {
      "client_id": "rubackup-client (525a99154f3505a2)",

```

```

"strategy_id": "Default",
"resource_type": "File system", ①
"resource": "/home/", ①
"normal_execution_script": "",
"exception_execution_script": "",
"priority": 100,
"restore_script": ""
},
"module_extensions": {
  "parameters": {
    "file_list": true,
    "numeric_owner": false,
    "use_snapshot": false,
    "ignore_errors_snapshot": true,
    "snapshot_type": "",
    "snapshot_size": 10,
    "script_before_snapshot": "",
    "script_after_snapshot": "",
    "script_error_snapshot": ""
  }
},
"common_modules_extensions": {
  "parameters": {
    "worker_parallelism": 8,
    "enable_multithreading": false,
    "enable_flexible_dedup": false,
    "network_parallelism": 8,
    "memory_threshold": 0,
    "deny_memory_exceed": false
  }
}
}
}

```

① Обязательное поле

201 CREATED

Правило добавлено в стратегию

Получение дополнительных параметров правила стратегии (GET /strategy_rules/extension)

Тело запроса отсутствует. В query-параметрах передается название ресурса (resource_type) или название ресурса и id правила, информацию о дополнительных параметрах которых необходимо получить

200 OK

Информация о дополнительных параметрах получена

Изменение правила стратегии (PATCH /strategy_rules/{id})

Также в query-параметрах передается id правила, информацию о котором необходимо изменить

```
{
  "data": {
    "strategy_rule": {
      "client_id": "rubackup-client (525a99154f3505a2)",
      "strategy_id": "Default",
      "resource_type": "File system",
      "resource": "/home/",
      "normal_execution_script": "",
      "exception_execution_script": "",
      "priority": 100,
      "restore_script": ""
    },
    "module_extensions": {
      "parameters": {
        "file_list": true,
        "numeric_owner": false,
        "use_snapshot": false,
        "ignore_errors_snapshot": true,
        "snapshot_type": "",
        "snapshot_size": 10,
        "script_before_snapshot": "",
        "script_after_snapshot": "",
        "script_error_snapshot": ""
      }
    },
    "common_modules_extensions": {
      "parameters": {
        "worker_parallelism": 8,
        "enable_multithreading": false,
        "enable_flexible_dedup": false,
        "network_parallelism": 8,
        "memory_threshold": 0,
        "deny_memory_exceed": false
      }
    }
  }
}
```

```
}
```

200 OK

Параметры правила стратегии изменены

400 BAD_REQUEST

Некорректное тело запроса

404 NOT_FOUND

Правило стратегии не найдено

Удаление правила стратегии (DELETE /strategy_rules/{id})

Тело запроса отсутствует. В query-параметрах передается id правила, которое необходимо удалить

200 OK

Информация о правиле стратегии получена

404 NOT_FOUND

Правило стратегии не найдено

Получение информации о правиле стратегии (GET /strategy_rules/{id})

Тело запроса отсутствует. В query-параметрах передается id правила, информацию о котором необходимо получить

200 OK

Правило стратегии удалено

404 NOT_FOUND

Правило стратегии не найдено

8.7.4. Администраторы стратегий

Получение списка администраторов для стратегии (GET /strategy_administrators)**200 OK**

Получен список администраторов для стратегии

Получение информации об администраторе стратегии (GET /strategy_administrators/{id})

Тела запроса нет. В query-параметрах передается id администратора стратегии, информацию о котором необходимо получить

200 OK

Получена информация об администраторе для стратегии

404 NOT_FOUND

Информация об администраторе для стратегии не найдена

Назначение администраторов для стратегии (POST /strategy_administrators)

```
{
  "data": {
    "strategy": "Default", ①
    "administrator": "Nobody" ①
  }
}
```

① Обязательное поле

201 CREATED

Администратор для стратегии назначен

Удаление списка администраторов стратегии (DELETE /strategy_administrators)

```
{
  "ids": [ ①
    1
  ]
}
```

① Обязательное поле

200 OK

Список администраторов стратегии удален

404 NOT_FOUND

Администратор стратегии не найден

Удаление администратора стратегии (DELETE /strategy_administrators/{id})

Тела запроса нет. В query-параметрах передается id администратора стратегии, которого необходимо удалить

200 OK

Администратор стратегии удален

404 NOT_FOUND

Администратор стратегии не найден

8.8. Глобальное расписание

8.8.1. Правила глобального расписания

Изменение статуса правила глобального расписания (PATCH /global_schedule)

```
{
  "data": [
    {
      "id": 1, ①
      "status": "wait" ①
    }
  ]
}
```

① Обязательное поле

200 OK

Статус правила изменен

400 BAD_REQUEST

Некорректное тело запроса

404 NOT_FOUND

Правило глобального расписания не найдено

Получение списка правил глобального расписания (GET /global_schedule)

200 OK

Параметры правила глобального расписания получены

Удаление списка правил глобального расписания (DELETE /global_schedule)

```
{
  "ids": [ ①
    1
  ]
}
```

① Обязательное поле

200 OK

Правило глобального расписания удалено

Добавление правила глобального расписания (POST /global_schedule)

```
{
  "data": {
    "global_schedule": {
      "name": "global_schedule_name", ①
      "client_id": "rubackup-client (525a99154f3505a2)", ①
      "resource_type": "File system", ①
      "resource": "/home/", ①
      "backup_type": "full",
      "pool_id": "Default",
      "storage_capacity": 10,
      "priority": 100,
      "crypto": "nocrypt",
      "validity_start_period": "2024-09-02T15:40:01.208623",
      "validity_end_period": "2025-09-02T15:40:01.208636",
      "periodic_launch": "1 min",
      "min": 0,
      "hour": 0,
      "dom": 1,
      "mon": 1,
      "dow": 1,
      "verify_flag": true,
      "verify_interval": "1 day",
      "storage_duration": "1 day",
      "move_copy_flag": false,
      "move_copy_pool": "Default",
      "move_copy_while": "1 day",
      "auto_delete_obsoleted_copy_flag": false,
      "inform_when_obsoleted_copy": "Nobody",
      "client_delete_flag": true,
      "notify_normal": "Nobody",
      "notify_normal_cc": "email@http://domain.ru[domain.ru]",
      "notify_exception": "Nobody",
      "notify_exception_cc": "email@http://domain.ru[domain.ru]",
      "notify_rule_validity": "Nobody",
      "notify_rule_validity_cc": "email@http://domain.ru[domain.ru]",
      "notify_verify": "Nobody",
      "notify_verify_cc": "email@http://domain.ru[domain.ru]",
      "status": "wait",
      "notify_normal_script": "",
      "notify_exception_script": "",
      "notify_end_storage_capacity": "Nobody",
      "notify_end_storage_capacity_cc": "email@http://domain.ru[domain.ru]",
      "restore_script": ""
    }
  }
}
```

```

    },
    "module_extensions": {
      "parameters": {
        "file_list": true,
        "numeric_owner": false,
        "use_snapshot": false,
        "ignore_errors_snapshot": true,
        "snapshot_type": "",
        "snapshot_size": 10,
        "script_before_snapshot": "",
        "script_after_snapshot": "",
        "script_error_snapshot": ""
      }
    },
    "common_modules_extensions": {
      "parameters": {
        "worker_parallelism": 8,
        "enable_multithreading": false,
        "enable_flexible_dedup": false,
        "network_parallelism": 8,
        "memory_threshold": 0,
        "deny_memory_exceed": false
      }
    }
  }
}

```

1 Обязательное поле

201 CREATED

Правило глобального расписания создано

Получение дополнительных параметров глобального расписания по id правила (GET /global_schedule/extension)

Тело запроса отсутствует. В query-параметрах передается название ресурса (resource_type) или название ресурса и id правила, информацию о дополнительных параметрах которых необходимо получить

200 OK

Информация о дополнительных параметрах получена

Изменение правила глобального расписания (PATCH /global_schedule/{id})

Также в query-параметрах передается id правила, информацию о котором необходимо изменить

```

{
  "data": {
    "global_schedule": {
      "name": "global_schedule_name",
      "pool_id": "Default",
      "storage_capacity": 10,
      "priority": 100,
      "validity_start_period": "2024-09-02T15:40:01.208623",
      "validity_end_period": "2025-09-02T15:40:01.208636",
      "periodic_launch": "1 min",
      "min": 0,
      "hour": 0,
      "dom": 1,
      "mon": 1,
      "dow": 1,
      "verify_flag": true,
      "verify_interval": "1 day",
      "storage_duration": "1 day",
      "move_copy_flag": false,
      "move_copy_pool": "Default",
      "move_copy_while": "1 day",
      "auto_delete_obsoleted_copy_flag": false,
      "inform_when_obsoleted_copy": "Nobody",
      "client_delete_flag": true,
      "notify_normal": "Nobody",
      "notify_normal_cc": "email@http://domain.ru[domain.ru]",
      "notify_exception": "Nobody",
      "notify_exception_cc": "email@http://domain.ru[domain.ru]",
      "notify_rule_validity": "Nobody",
      "notify_rule_validity_cc": "email@http://domain.ru[domain.ru]",
      "notify_verify": "Nobody",
      "notify_verify_cc": "email@http://domain.ru[domain.ru]",
      "status": "wait",
      "notify_normal_script": "",
      "notify_exception_script": "",
      "notify_end_storage_capacity": "Nobody",
      "notify_end_storage_capacity_cc": "email@http://domain.ru[domain.ru]",
      "restore_script": ""
    },
    "module_extensions": {
      "parameters": {
        "file_list": true,
        "numeric_owner": false,
        "use_snapshot": false,

```

```

    "ignore_errors_snapshot": true,
    "snapshot_type": "",
    "snapshot_size": 10,
    "script_before_snapshot": "",
    "script_after_snapshot": "",
    "script_error_snapshot": ""
  },
  "common_modules_extensions": {
    "parameters": {
      "worker_parallelism": 8,
      "enable_multithreading": false,
      "enable_flexible_dedup": false,
      "network_parallelism": 8,
      "memory_threshold": 0,
      "deny_memory_exceed": false
    }
  }
}

```

200 OK

Параметры правила глобального расписания изменены

400 BAD_REQUEST

Некорректное тело запроса

404 NOT_FOUND

Правило глобального расписания не найдено

Получение информации о правиле глобального расписания (GET /global_schedule/{id})

Тело запроса отсутствует. В query-параметрах передается id правила, информацию о котором необходимо получить

200 OK

Информация о правиле глобального расписания получена

404 NOT_FOUND

Правило глобального расписания не найдено

Удаление правила глобального расписания (DELETE /global_schedule/{id})

Тело запроса отсутствует. В query-параметрах передается id правила, которое необходимо удалить

200 OK

Правило глобального расписания удалено

404 NOT_FOUND

Правило глобального расписания не найдено

Создание полной резервной копии по правилу глобального расписания (POST /global_schedule/{id}/execution)

Тело запроса отсутствует. В query-параметрах передается id правила, которое необходимо выполнить ("принудительно")

200 OK

Правило глобального расписания выполнено

404 NOT_FOUND

Правило глобального расписания не найдено

8.8.2. Репликация правил глобального расписания

Получение списка настроенных репликаций правил глобального расписания (GET /global_schedule_replication)**200 OK**

Получен список правил копирования

Настройка репликации правил глобального расписания (POST /global_schedule_replication)

```
{
  "data": {
    "rule": 1, ①
    "pool": "Default", ①
    "capacity": 0,
    "capacity_enabled": false,
    "period": "1mon",
    "period_enabled": false,
    "work_window_start": "00:00:00",
    "work_window_end": "00:00:00"
  }
}
```

① Обязательное поле

201 CREATED

правило копирования создано

Удаление списка настроенных репликаций правил глобального расписания (DELETE /global_schedule_replication)

```
{
  "ids": [ ❶
    1
  ]
}
```

❶ Обязательное поле

200 OK

правило копирования удалено

404 NOT_FOUND

правило копирования не найдена

Получение информации о настроенной репликации правила глобального расписания (GET /global_schedule_replication/{id})

Тела запроса нет. В query-параметрах передается id правила копирования, которое необходимо удалить

200 OK

получена информация о правиле копирования

Удаление настроенной репликации правила глобального расписания (DELETE /global_schedule_replication/{id})

Тела запроса нет. В query-параметрах передается id правила копирования, информацию о котором необходимо получить

200 OK

правило копирования удалено

404 NOT_FOUND

правило копирования не найдено

Изменение параметров репликации правил глобального расписания (PATCH /global_schedule_replication/{id})

Также в query-параметрах передается id правила копирования, информацию о котором необходимо изменить

```
{
  "data": {
    "capacity": 0,
    "capacity_enabled": false,
```

```

    "period": "1mon",
    "period_enabled": false,
    "work_window_start": "00:00:00",
    "work_window_end": "00:00:00"
  }
}

```

200 OK

правило копирования удалено

404 NOT_FOUND

правило копирования не найдена

8.8.3. Запросы на добавление правил глобального расписания

Получение списка запросов клиента на добавление правила глобального расписания (GET /new_rule_requests)

200 OK

Получен список запросов клиента на добавление правила глобального расписания

Одобрение запроса клиента на добавление правила глобального расписания (POST /new_rule_requests/approve)

```

{
  "ids": [ ①
    1
  ]
}

```

① Обязательное поле

200 NO_CONTENT

Запросы клиента на добавление правила глобального расписания одобрены

207 MULTI_STATUS

Часть из списка запросов клиента на добавление правила глобального расписания одобрена, часть нет

404 NOT_FOUND

Запросы клиента на добавление правила глобального расписания не найдены

Отклонение запроса клиента на добавление правила глобального расписания (DELETE /new_rule_requests/decline)

```
{
  "ids": [ ❶
    1
  ]
}
```

❶ Обязательное поле

200 NO_CONTENT

Запросы клиента на добавление правила глобального расписания отклонены

207 MULTI_STATUS

Часть из списка запросов клиента на добавление правила глобального расписания отклонена, часть нет

404 NOT_FOUND

Запросы клиента на добавление правила глобального расписания не найдены

Получение информации о запросе клиента на добавление правила глобального расписания (GET /new_rule_requests/{id})

Тела запроса нет. В query-параметрах передается id запроса, информацию о котором необходимо получить

200 OK

Получена информация о запросе клиента на добавление правила глобального расписания

8.8.4. Запросы на удаление правил глобального расписания

Получение списка запросов клиента на удаление правила глобального расписания (GET /delete_rule_requests)

200 OK

Получен запросов клиента на удаление правила глобального расписания

Одобрение запроса клиента на удаление правила глобального расписания (POST /delete_rule_requests/approve)

```
{
  "ids": [ ❶
    1
  ]
}
```

```
]
}
```

1 Обязательное поле

200 NO_CONTENT

Запросы клиента на удаление правила глобального расписания одобрены

207 MULTI_STATUS

Часть из списка запросов клиента на удаление правила глобального расписания одобрена, часть нет

404 NOT_FOUND

Запросы клиента на удаление правила глобального расписания не найдены

Отклонение запроса клиента на удаление правила глобального расписания (DELETE /delete_rule_requests/decline)

```
{
  "ids": [ 1
  ]
}
```

1 Обязательное поле

204 NO_CONTENT

Запросы клиента на удаление правила глобального расписания отклонены

207 MULTI_STATUS

Часть из списка запросов клиента на удаление правила глобального расписания отклонена, часть нет

404 NOT_FOUND

Запросы клиента на удаление правила глобального расписания не найдены

Получение информации о запросе клиента на удаление правила глобального расписания (GET /delete_rule_requests/{id})

Тела запроса нет. В query-параметрах передается id запроса, информацию о котором необходимо получить

200 OK

Получена информация о запросе клиента на удаление правила глобального расписания

8.8.5. Ограничения пропускной способности

Получение списка ограничений пропускной способности правила глобального расписания (GET /rule_bandwidth)

200 OK

получен список ограничений пропускной способности

Удаление списка ограничений пропускной способности правила глобального расписания (DELETE /rule_bandwidth)

```
{
  "ids": [ ①
    1
  ]
}
```

① Обязательное поле

200 OK

список ограничений пропускной способности удален

404 NOT_FOUND

ограничение пропускной способности не найдено

Ограничение пропускной способности правила глобального расписания (POST /rule_bandwidth)

```
{
  "data": {
    "rule": 1, ①
    "backup_bandwidth": 0,
    "restore_bandwidth": 0,
    "window_start": "00:00:00",
    "window_end": "00:00:00"
  }
}
```

① Обязательное поле

201 CREATED

ограничение пропускной способности создано

Изменение ограничения пропускной способности правила глобального расписания (PATCH /rule_bandwidth/{id})

Также в query-параметрах передается id ограничения, информацию о котором необходимо изменить

```
{
  "data": {
    "rule": 1,
    "backup_bandwidth": 0,
    "restore_bandwidth": 0,
    "window_start": "00:00:00",
    "window_end": "00:00:00",
    "id": 1
  }
}
```

200 OK

параметры ограничения пропускной способности изменены

400 BAD_REQUEST

некорректное тело запроса

404 NOT_FOUND

ограничение пропускной способности не найдено

Удаление ограничения пропускной способности правила глобального расписания (DELETE /rule_bandwidth/{id})

Тела запроса нет. В query-параметрах передается id ограничения, которое необходимо удалить

200 OK

ограничение пропускной способности удалено

404 NOT_FOUND

ограничение пропускной способности не найдено

Получение информации об ограничении пропускной способности правила глобального расписания (GET /rule_bandwidth/{id})

Тела запроса нет. В query-параметрах передается id ограничения, информацию о котором необходимо получить

200 OK

получена информация об ограничении пропускной способности

404 NOT_FOUND

ограничение пропускной способности не найдено

8.9. Задачи

8.9.1. Задачи

Удаление нескольких задач (DELETE /task_queue)

```
{
  "ids": [ ❶
    1
  ]
}
```

❶ Обязательное поле

200 OK

Список задач из очереди задач удален

404 NOT_FOUND

Задача не найдена в очереди

Получение очереди задач (GET /task_queue)

200 OK

Получен список задач из очереди задач

Создание срочной полной резервной копии (POST /task_queue)

```
{
  "data": {
    "task": {
      "resource_type": "File system", ❶
      "resource": "/home/", ❶
      "client": "rubcp-client (525a99154f3505a2)", ❶
      "backup_type": "full", ❶
      "pool": "Default", ❶
      "crypto": "nocrypt", ❶
      "priority": 100, ❶
      "storage_duration": "1 Years", ❶
      "archiving": false ❶
    },
    "module_extensions": {
      "parameters": {
        "file_list": true,
        "numeric_owner": false,
        "use_snapshot": false,

```

```

    "ignore_errors_snapshot": true,
    "snapshot_type": "",
    "snapshot_size": 10,
    "script_before_snapshot": "",
    "script_after_snapshot": "",
    "script_error_snapshot": ""
  },
  "common_modules_extensions": {
    "parameters": {
      "worker_parallelism": 8,
      "enable_multithreading": false,
      "enable_flexible_dedup": false,
      "network_parallelism": 8,
      "memory_threshold": 0,
      "deny_memory_exceed": false
    }
  }
}

```

1 Обязательное поле

201 CREATED

Создана задача на срочное РК

Получение дополнительных параметров созданной задачи (GET /task_queue/extension)

Тело запроса отсутствует. В query-параметрах передается название ресурса (resource_type) или название ресурса и id правила, информацию о дополнительных параметрах которых необходимо получить

200 OK

Информация о дополнительных параметрах получена

Удаление задачи (DELETE /task_queue/{id})

Тело запроса отсутствует. В query-параметрах передается id задачи, которую необходимо удалить

200 OK

Задача удалена из очереди

404 NOT_FOUND

Задача не найдена в очереди

Получение информации о задаче из очереди (GET /task_queue/{id})

Тело запроса отсутствует. В query-параметрах передается id задачи, информацию о которой необходимо получить

200 OK

Информация о задаче из очереди получена

404 NOT_FOUND

Задача не найдена в очереди

Удаление задачи с определенным статусом (DELETE /task_queue/{status})

Тело запроса отсутствует. В query-параметрах передается статус, задачи с которым необходимо удалить

Возможные статусы: `error`, `killed`, `obsoleted` (`obsoleted` - устаревшие, т.к. в статусе `done`).

200 OK

Задачи с выбранным статусом удалены из очереди

404 NOT_FOUND

Задач с выбранным статусом не найдена в очереди

8.9.2. Статусы задач**Получение списка возможных статусов задач (GET /task_status)****200 OK**

Получен список возможных статусов задач

8.9.3. Перезапуск задач**Перезапуск задачи (POST /restart_task)**

```
{
  "data": {
    "tasks_ids": [ ❶
      1
    ]
  }
}
```

❶ Обязательное поле

201 CREATED

Задача перезапущена

400 BAD_REQUEST

Некорректное тело запроса (задачу с таким статусом нельзя перезапустить)

8.9.4. Приостановка задач

Приостановка выполнения задачи (POST /put_task_on_pause)

```
{
  "data": {
    "task_id": 1 ①
  }
}
```

① Обязательное поле

200 OK

Задача приостановлена

400 BAD_REQUEST

Некорректное тело запроса (задачу с таким статусом нельзя приостановить)

8.9.5. Возобновление выполнения задач

Возобновление приостановленной задачи (POST /continue_task_working)

```
{
  "data": {
    "task_id": 1 ①
  }
}
```

① Обязательное поле

200 OK

Задача возобновлена

400 BAD_REQUEST

Некорректное тело запроса (задачу с таким статусом нельзя возобновить)

8.9.6. Принудительное завершение задач

Убийство задачи (POST `/kill_task`)

```
{
  "data": {
    "task_id": 1 ①
  }
}
```

① Обязательное поле

200 OK

Задача убита

400 BAD_REQUEST

Некорректное тело запроса (задачу с таким статусом нельзя убить)

8.9.7. Очередь задач ленточных библиотек

Получение списка задач из очереди ленточных библиотек (GET `/tl_task_queue`)

200 OK

Получен список задач из очереди задач ленточных библиотек

Получение информации о задаче из очереди ленточных библиотек (GET `/tl_task_queue/{id}`)

Тело запроса отсутствует. В query-параметрах передается id задачи из очереди задач ленточных библиотек, информацию о которой необходимо получить

200 OK

Получена информация о задаче из очереди задач ленточных библиотек

404 NOT_FOUND

Задача в очереди задач ленточных библиотек не найдена

8.10. Уведомления

8.10.1. Статус уведомлений

Получение списка статусов уведомлений (GET `/notifications_status`)

200 OK

Получен список существующих статусов для уведомлений

8.10.2. Уведомления

Получение списка уведомлений (GET /notifications)

200 OK

Получен список уведомлений

Получение информации об уведомлении (GET /notifications/{id})

Тела запроса нет. В query-параметрах передается id уведомления, информацию о котором необходимо получить

200 OK

Получена информация об уведомлении

8.10.3. Уведомления групп

Удаление списка групп для уведомлений (DELETE /user_groups)

```
{
  "ids": [ ①
    1
  ]
}
```

① Обязательное поле

200 OK

Список групп пользователей удален

404 NOT_FOUND

Группа пользователей не найдена

Получение списка групп для уведомлений (GET /user_groups)

200 OK

Список групп пользователей получен

Добавление группы уведомлений (POST /user_groups)

```
{
  "data": {
    "groupname": "my_group_name", ①
    "description": ""
  }
}
```

1 Обязательное поле**201 CREATED**

Создана группа пользователей

Изменение группы для уведомлений (PATCH /user_groups/{id})

Также в query-параметрах передается id группы пользователей, информацию о которой необходимо изменить

```
{
  "data": {
    "groupname": "my_group_name",
    "description": ""
  }
}
```

200 OK

Параметры группы пользователей изменены

400 BAD_REQUEST

Некорректное тело запроса

404 NOT_FOUND

Группа пользователей не найдена

Удаление группы для уведомлений (DELETE /user_groups/{id})

Тела запроса нет. В query-параметрах передается id группы пользователей, которую необходимо удалить

200 OK

Группа пользователей удалена

404 NOT_FOUND

Группа пользователей не найдена

Получение информации о группе уведомлений (GET /user_groups/{id})

Тела запроса нет. В query-параметрах передается id группы пользователей, информацию о которой необходимо получить

200 OK

Получена информация о группе пользователей

404 NOT_FOUND

Группа пользователей не найдена

8.11. Журналы

8.11.1. Журнал операций клиента

Получение журнала со списком операций клиентов (GET /clients_log)

200 OK

Получен список записей в журнале операций клиентов

Получение одной записи из журнала с операций клиентов (GET /clients_log/{id})

Тела запроса нет. В query-параметрах передается id операции клиента, информацию о которой необходимо получить

200 OK

Получена запись в журнале операций клиентов

404 NOT_FOUND

Запись в журнале операций клиентов не найдена

8.11.2. Журнал глобального расписания

Получение списка записей в журнале глобального расписания (GET /global_schedule_log)

200 OK

Получен список записей в журнале правил глобального расписания

Получение информации о записи в журнале глобального расписания (GET /global_schedule_log/{id})

Тела запроса нет. В query-параметрах передается id операции из журнала правил глобального расписания, информацию о которой необходимо получить

200 OK

Получена запись в журнале правил глобального расписания

404 NOT_FOUND

Запись в журнале правил глобального расписания не найдена

8.11.3. Журнал операций медиасерверов

Получение журнала со списком операций медиасерверов (GET /media_servers_log)

200 OK

Получен список записей в журнале операций медиасерверов

Получение журнала с операцией медиасервера (GET /media_servers_log/{id})

Тела запроса нет. В query-параметрах передается id операции медиасервера, информацию о которой необходимо получить

200 OK

Получена запись в журнале операций медиасерверов

404 NOT_FOUND

Запись в журнале операций медиасерверов не найдена

8.11.4. Журнал репозитория

Получение списка записей в журнале репозитория (GET /repository_log)

200 OK

Получен список записей в журнале репозитория

Получение информации о записи в журнале репозитория (GET /repository_log/{id})

Тела запроса нет. В query-параметрах передается id записи в журнале репозитория, информацию о которой необходимо получить

200 OK

Получена запись в журнале репозитория

404 NOT_FOUND

Запись в журнале репозитория не найдена

8.11.5. Журнал задач

Получение информации о записи в журнале задач (GET /task_log/{id})

Тела запроса нет. В query-параметрах передается id записи в журнале задач, информацию о которой необходимо получить

200 OK

Получена запись в журнале задач

404 NOT_FOUND

Запись в журнале задач не найдена

Получение списка записей в журнале задач (GET /task_log)**200 OK**

Получен список записей в журнале задач

8.11.6. Журнал операций сервера**Получение журнала серверных операций по задаче (GET /server_task_info)**

Аналогично действиям:

1. Перейти на вкладку очереди задач
2. Выбрать задачу в списке
3. Нажать "Журналы"
4. Выбрать "Серверные журналы"

Тела запроса нет. В query-параметрах передается id задачи из очереди задач, журнал которой необходимо получить

200 OK

Получен журнал серверных операций по задаче

8.11.7. Журнал клиентских операций**Получение журнала клиентских операций по задаче (GET /client_task_info)**

Тела запроса нет. В query-параметрах передается id задачи из очереди задач, журнал которой необходимо получить

Аналогично действиям:

1. Перейти на вкладку очереди задач
2. Выбрать задачу в списке
3. Нажать "Журналы"
4. Выбрать "Клиентские журналы"

200 OK

Получен журнал клиентских операций по задаче

8.11.8. Журнал мониторинга сервера**Получение списка записей в журнале мониторинга сервера (GET /rubakup_server_system_monitoring)****200 OK**

Получен список записей в журнале мониторинга сервера

Получение информации о записи в журнале мониторинга сервера (GET /rubakup_server_system_monitoring/{id})

Тела запроса нет. В query-параметрах передается id записи в журнале мониторинга сервера, информацию о которой необходимо получить

200 OK

Получена информация о записи в журнале мониторинга сервера

404 NOT_FOUND

Запись в журнале мониторинга сервера не найдена

8.11.9. Журнал событий информационной безопасности

Удаление списка записей из журналов событий ИБ (DELETE /security_audit_log)

```
{
  "ids": [ ①
    1
  ]
}
```

① Обязательное поле

204 NO_CONTENT

Список записей из журналов ИБ удален

Просмотр журналов событий ИБ (GET /security_audit_log)

200 OK

Список записей в журнале ИБ получен

Удаление записи из журнала событий ИБ (DELETE /security_audit_log/{id})

Тела запроса нет. В query-параметрах передается id записи, которую необходимо удалить

204 NO_CONTENT

Запись из журнала ИБ удалена

Просмотр записи из журнала событий ИБ (GET /security_audit_log/{id})

Тела запроса нет. В query-параметрах передается id записи, информацию о которой необходимо получить

200 OK

Информация о записи из журнала ИБ получена

8.12. Отчеты

8.12.1. Изменение статуса отчета (PATCH /reports)

```
{
  "data": [
    {
      "id": 1, ①
      "status": "wait" ①
    }
  ]
}
```

① Обязательное поле

200 OK

Статус отчета изменен

400 BAD_REQUEST

Некорректное тело запроса

404 NOT_FOUND

Отчет не найден

8.12.2. Удаление списка отчетов (DELETE /reports)

```
{
  "ids": [ ①
    1
  ]
}
```

① Обязательное поле

200 OK

Список отчетов удален

404 NOT_FOUND

Отчет не найден

8.12.3. Получение списка отчетов (GET /reports)

200 OK

Получен список отчетов

8.12.4. Создание отчета (POST /reports)

```
{
  "data": {
    "name": "my_report_name", ①
    "owner": "Nobody", ①
    "min": 0,
    "hour": 0,
    "dom": 1,
    "mon": 1,
    "dow": 1,
    "perpetually": true,
    "start_period": "2024-09-07T22:06:09.809052",
    "end_period": "2025-09-07T22:06:09.809067",
    "status": "wait",
    "notify": "Nobody",
    "notify_cc": "email@http://domain.ru[domain.ru]",
    "description": ""
  }
}
```

① Обязательное поле

201 CREATED

Отчет создан

8.12.5. Изменение отчета (PATCH /reports/{id})

Также в query-параметрах передается id отчета, информацию о котором необходимо изменить

```
{
  "data": {
    "perpetually": true,
    "start_period": "2024-09-07T22:06:09.809052",
    "end_period": "2025-09-07T22:06:09.809067",
    "status": "wait",
    "notify": "Nobody",
    "notify_cc": "email@http://domain.ru[domain.ru]",
    "description": ""
  }
}
```

200 OK

Параметры отчета изменены

400 BAD_REQUEST

Некорректное тело запроса

404 NOT_FOUND

Отчет не найден

8.12.6. Получение информации об отчете (DELETE /reports/{id})

Тела запроса нет. В query-параметрах передается id отчета, который необходимо удалить

200 OK

Отчет удален

404 NOT_FOUND

Отчет не найден

8.12.7. Удаление отчета (GET /reports/{id})

Тела запроса нет. В query-параметрах передается id отчета, информацию о котором необходимо получить

200 OK

Получена информация об отчете

404 NOT_FOUND

Отчет не найден

8.13. Microsoft Active Directory**8.13.1. Настройка LDAP****Удаление соединения (DELETE /ldap/configuration)****200 OK**

соединение с контроллером домена удалено

Настройка соединения с MS AD (POST /ldap/configuration)

```
{
  "data": {
    "domain_type": "MSAD", ①
    "protocol": "ldap", ①
    "address": "", ①
    "port": 389, ①
  }
}
```

```

"client_cert": "",
"root_cert": "",
"domain": "", ①
"login": "", ①
"password": "", ①
"user_search_base": "", ①
"group_search_base": "" ①
}
}

```

① Обязательное поле

201 CREATED

соединение с контроллером домена создано

Получение параметров настроенного соединения с MS AD (GET /ldap/configuration)

200 OK

получены параметры соединения с контроллером домена

8.13.2. Ассоциация групп MS AD и ролей RuBackup

Удаление ассоциации групп MS AD и ролей RuBackup (DELETE /ldap/assosiations)

```

{
  "ids": [ ①
    1
  ]
}

```

① Обязательное поле

200 OK

Ассоциация группы MS AD и ролей RuBackup удалена

404 NOT_FOUND

Ассоциация группы MS AD и ролей RuBackup не найдена

Получение списка ассоциаций групп MS AD и ролей RuBackup (POST /ldap/assosiations)

```

{
  "data": [
    {
      "domain_group": "Administrators", ①
      "rubakup_role": "auditor" ①
    }
  ]
}

```

```
]
}
```

1 Обязательное поле

201 CREATED

Ассоциация группы MS AD и ролей RuBackup создана

Ассоциация групп MS AD и ролей RuBackup (GET /ldap/assosiations)

200 OK

Получен список ассоциаций групп MS AD и ролей RuBackup