



Система управления  
ИТ-инфраструктурой

# Зодиак.АйТиЭм

**Руководство администратора**

Версия 3.35

## ОГЛАВЛЕНИЕ

1.	Общее Описание системы управления «Зодиак.АйТиЭм».....	6
1.1	Список используемых терминов и сокращений.....	6
1.2	Требования к системе.....	7
1.2.1	Серверная компонента.....	7
1.2.2	СУБД PostgreSQL.....	7
1.2.3	Управляемые компьютеры.....	7
2.	Начало работы.....	8
2.1	Вход в систему «Зодиак.АйТиЭм».....	8
2.2	Обзор интерфейса и навигация.....	9
2.2.1	Общая компоновка.....	9
2.2.2	Управление логическими окнами.....	10
2.2.3	Табличное представление данных.....	10
2.2.4	Конструктор фильтра.....	12
2.2.5	Конструктор расписания.....	14
3.	Режимы веб-консоли управления.....	16
3.1	Компьютеры.....	16
3.1.1	Карточка компьютера.....	17
3.1.2	Просмотр назначений.....	20
3.2	Ассоциации.....	20
3.2.1	Каталог ассоциаций.....	20
3.2.2	Создание ассоциации.....	21
3.2.3	Связывание групп и директив.....	22
3.2.4	Глобальное связывание директив.....	23
3.3	Группы.....	25
3.3.1	Каталог групп.....	25
3.3.2	Создание группы.....	25
3.3.3	Авто-группы.....	28
3.4	Директивы.....	29
3.4.1	Создание директивы.....	29
3.4.2	Элемент директивы типа скрипт.....	30
3.4.3	Элемент директивы типа пакеты.....	33
3.4.4	Элемент директивы типа репозитории.....	36
3.4.5	Элемент директивы «Конфигурация».....	37

3.4.6	Элемент директивы «Развертывание» .....	39
3.4.7	Элемент директивы «Сканирование сети».....	44
3.4.8	Элемент директивы «Инвентаризация WMI» .....	46
3.4.9	Элемент директивы «Инвентаризация пользователей» .....	48
3.4.10	Элемент директивы «Инвентаризация ПО» .....	49
3.4.11	Просмотр статистики работы скрипта .....	50
3.4.12	Просмотр результатов работы скрипта.....	52
3.4.13	Директивы точки обслуживания.....	53
3.5	Выборки.....	56
3.5.1	Каталог выборок.....	56
3.5.2	Настройка свойств выборки.....	57
3.5.3	Просмотр выборки .....	62
3.6	Представления.....	63
3.6.1	Каталог представлений.....	63
3.6.2	Создание представления .....	63
3.6.3	Настройка представления.....	64
3.7	Пользователи .....	70
3.7.1	Каталог пользователей .....	70
3.8	Подключения.....	73
3.8.1	Каталог подключений.....	73
3.8.2	Создание нового подключения.....	73
3.9	Роли.....	76
3.9.1	Каталог ролей .....	76
3.9.2	Создание новой роли.....	77
3.10	Планировщик задач.....	79
3.10.1	Каталог задач .....	79
3.10.2	Создание новой задачи .....	80
3.11	Версии агентов .....	81
3.11.1	Общий вид окна «Версии агентов».....	81
3.11.2	Статистика версий.....	82
3.11.3	Обновление «по воздуху» (OTA) .....	83
3.12	Границы.....	85
3.12.1	Каталог границ.....	85
3.13	Лог изменений .....	90
3.13.1	Каталог лога изменений.....	90

3.14	Серверы .....	91
3.14.1	Каталог серверов .....	91
3.15	Соседние устройства .....	92
3.15.1	Список соседних устройств.....	92
3.15.2	Подключение к соседним устройствам.....	93
3.16	Пакеты развертывания.....	93
3.16.1	Создание пакета развертывания.....	93
3.16.2	Формирование структуры пакета развертывания.....	94
3.16.3	Настройка инсталляции пакета.....	95
3.16.4	Настройка обнаружения установленного пакета.....	97
3.16.5	Настройка проверки требований для пакета.....	98
3.16.6	Финализация пакета развертывания.....	100
3.17	Каталог ПО.....	101
3.17.1	Карточка ПО .....	103
3.17.2	Настройки каталога ПО.....	110
3.18	Отчеты ПО .....	111
3.18.1	Создание и редактирование отчета.....	112
3.18.2	Расчет отчета .....	118
3.18.3	Просмотр всех результатов расчета.....	119
3.18.4	Просмотр конкретного результата расчета.....	120
3.19	Лицензия .....	122
4.	Помощник Администратора.....	123
4.1	Общая схема работы.....	123
4.2	Установка приложения Помощника.....	125
4.3	Начальная настройка приложения Помощника .....	128
4.4	Общие принципы размещения функционала Помощника в веб-консоли .....	133
4.5	Конфигурация приложения Помощника.....	137
4.6	Журналы приложения Помощника .....	140
4.7	Изменение настроек связи приложения Помощника и веб-консоли.....	142
4.7.1	Ручная модификация файла настроек связи .....	144
4.7.2	Консольные команды изменения настроек связи .....	145
4.8	Установка дополнительных плагинов.....	146
4.8.1	Установка посредством консольной команды.....	147
4.8.2	Установка посредством файловой ассоциации.....	150
4.9	Изоляция плагинов .....	151

4.9.1	Политика размещения плагинов в пулах.....	152
4.9.2	Ручное размещение плагинов в пулах.....	152
4.10	Справка по плагинам.....	155
4.11	Конфигурация плагинов .....	157
4.12	Журналы плагинов.....	160
4.13	Создание плагинов .....	161
4.13.1	Размещение проекта плагина .....	162
4.13.2	Структура проекта плагина .....	164
4.13.3	Работа с проектом плагина в редакторе Visual Studio Code .....	166
4.13.4	Редактирование основных файлов плагина .....	168
4.13.5	Расширенные сценарии реализации плагинов .....	177
5.	Режимы управляемых компьютеров.....	187
5.1	Пользовательский интерфейс агента.....	187
5.1.1	Компаньон агента.....	187
5.1.2	Общий вид пользовательского интерфейса.....	188
5.1.3	Изменение цветовой палитры интерфейса .....	188
5.1.4	Центр уведомлений .....	189
5.1.5	Приостановка уведомлений.....	190
5.1.6	Магазин приложений.....	191
5.2	Утилиты командной строки .....	192
5.2.1	Утилита нотификации zonot.....	192
5.2.2	Утилита инвентаризации zocol .....	192
6.	Примеры использования системы «Зодиак.АйТиЭм» .....	193
6.1	Инвентаризация.....	193
6.1.1	Базовая инвентаризация .....	193
6.1.2	Настраиваемая инвентаризация.....	193
6.1.3	Файловая инвентаризация под Windows .....	195
6.1.4	Файловая инвентаризация под Linux.....	197
6.2	Управление конфигурацией.....	198
6.2.1	Выполнение скриптов PowerShell под Windows .....	198
6.2.2	Выполнение скриптов Shell под Linux.....	199
6.3	Обнаружение соседних устройств.....	200
6.4	Установка ПО .....	200
6.5	Применение точек обслуживания .....	201

# 1. ОБЩЕЕ ОПИСАНИЕ СИСТЕМЫ УПРАВЛЕНИЯ «ЗОДИАК.АЙТИЭМ»

**Система управления «Зодиак.АйТиЭм»** — это простая и гибкая система для автоматизации управления ИТ-инфраструктурой предприятия. Система управления «Зодиак.АйТиЭм» позволяет организовать решение задач следующих видов:

1. Инвентаризация серверов и рабочих станций под управлением ОС семейства Microsoft Windows и ОС семейства Linux. В том числе:
  - а. Базовая инвентаризация с фиксированным набором параметров, периодически выполняемая агентами системы «Зодиак.АйТиЭм» сразу после их установки на сервере или рабочей станции.
  - б. Настраиваемая инвентаризация с произвольным набором параметров, выполняемая агентами по заданному расписанию с помощью заданных администратором системы «Зодиак.АйТиЭм» скриптов.
2. Управление конфигурацией посредством исполнения скриптов, модифицирующих конфигурацию управляемых компьютеров.
3. Обнаружение устройств, соседних по отношению к управляемым компьютерам, а также подключение к управляемым компьютерам и их соседним устройствам из веб-консоли администратора системы «Зодиак.АйТиЭм».
4. Установка ПО на управляемые компьютеры.

## 1.1 Список используемых терминов и сокращений

Таблица 1 Список используемых терминов и сокращений

Термин	Описание
Система «Зодиак.АйТиЭм», Зодиак.АйТиЭм	Система управления ИТ-инфраструктурой Зодиак.АйТиЭм.
Управляемый компьютер, УК	Физический или виртуальный сервер или рабочая станция, на котором установлен агент системы «Зодиак.АйТиЭм».
Точка обслуживания, ТО	Физический или виртуальный сервер или рабочая станция, на котором установлен агент системы «Зодиак.АйТиЭм» в режиме «Точка обслуживания». Используется для оптимизации передачи контента большого объема.
База данных, БД	База данных PostgreSQL. Может использоваться сторонний кластер СУБД PostgreSQL. Хранит информацию о настройках системы, а также результаты сбора и анализа данных.
Виртуальная машина, VM	Программа, которая эмулирует реальный (физический) компьютер со всеми его компонентами (жесткий диск, DVD-ROM, BIOS, сетевые адаптеры и т.д.).

## 1.2 Требования к системе

### 1.2.1 Серверная компонента

Серверная часть системы «Зодиак» может быть установлена для работы под управлением одной из следующих операционных систем **в минимальной установке**:

- Windows Server 2012 R2
- Windows Server 2016
- Windows Server 2019
- Windows Server 2022
- ALT Linux 9
- ALT Linux 10
- РЕД ОС 7
- Astra Linux Common Edition 2.12, релиз «Орел»  
Astra Linux Special Edition 1.6, релиз «Смоленск»  
Astra Linux Special Edition 1.7, уровни «Орел», «Воронеж», «Смоленск»

**Минимальные** системные требования для серверов:

- 2 CPU;
- 1 Гбайт (GB) RAM;
- 100 Гбайт (GB) дискового пространства.

### 1.2.2 СУБД PostgreSQL

Для установки серверной части системы «Зодиак» необходимо создать экземпляр СУБД PostgreSQL **не ниже версии 12**.

Данный единственный экземпляр СУБД предназначен для использования всеми компонентами серверной части системы «Зодиак».

### 1.2.3 Управляемые компьютеры

**Минимальные** требования к ОС для установки агентов:

- Windows 7
- Windows Server 2008 R2 или Windows Server 2012 R2
- Enterprise Linux 7 (RHEL and CentOS)
- Debian 8 или Ubuntu 14.04

## 2. НАЧАЛО РАБОТЫ

### 2.1 Вход в систему «Зодиак.АйТиЭм»

1. В адресной строке браузера введите IP-адрес или DNS-имя системы, в которой установлена система «Зодиак.АйТиЭм», как показано на рисунке 2.1.

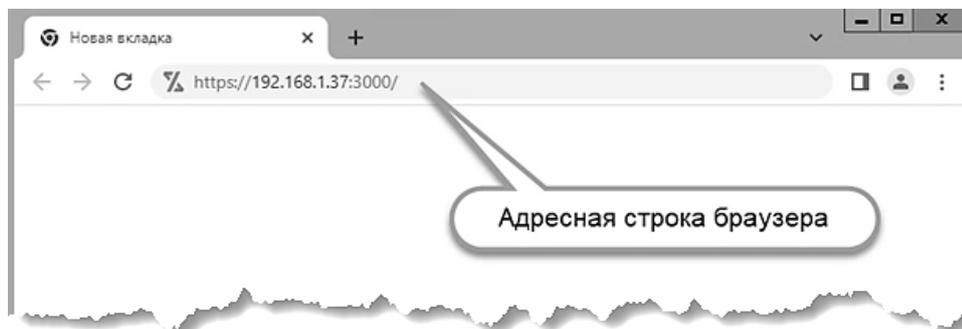


Рисунок 2.1 Пример адресной строки в браузере Chrome

2. В окне аутентификации сервера авторизации введите логин и пароль, который использовался при разворачивании сервера авторизации и нажмите «Sign In».

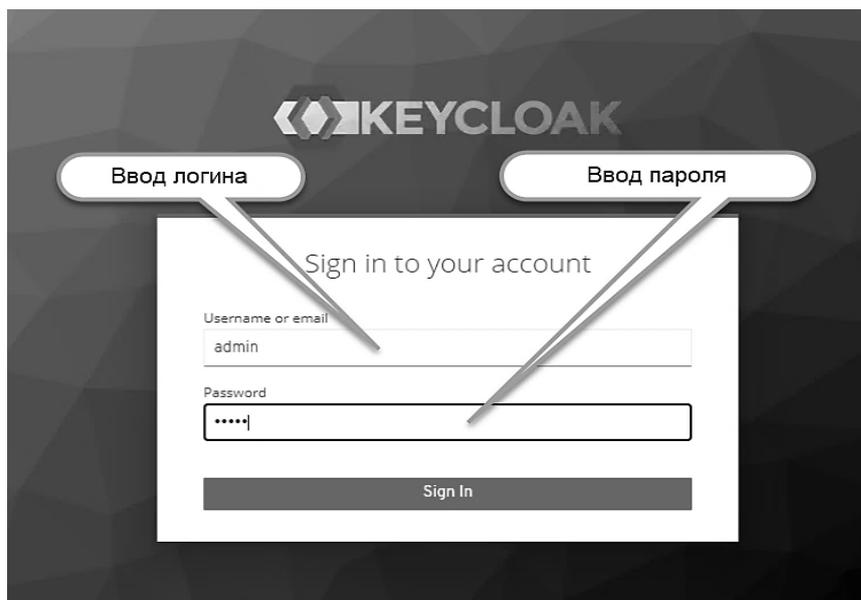


Рисунок 2.2 Вход в систему через окно аутентификации SSO-провайдера KeyCloak

---

#### Примечание

Система «Зодиак.АйТиЭм» ориентирована на работу с внешним SSO-провайдером (OpenID Connect). В зависимости от используемого SSO-провайдера внешний вид страницы входа может различаться.

---

## 2.2 Обзор интерфейса и навигация

### 2.2.1 Общая компоновка

Взаимодействие пользователя с системой осуществляется посредством графического пользовательского интерфейса, представляющего собой одностраничное веб-приложение.

В соответствии с концепцией одностраничных веб-приложений (SPA), изменение внешнего вида пользовательского интерфейса осуществляется в рамках единой веб-страницы, без выполнения переходов.

Общая компоновка пользовательского интерфейса системы «Зодиак.АйТиЭм» представлена на рисунке 2.3.

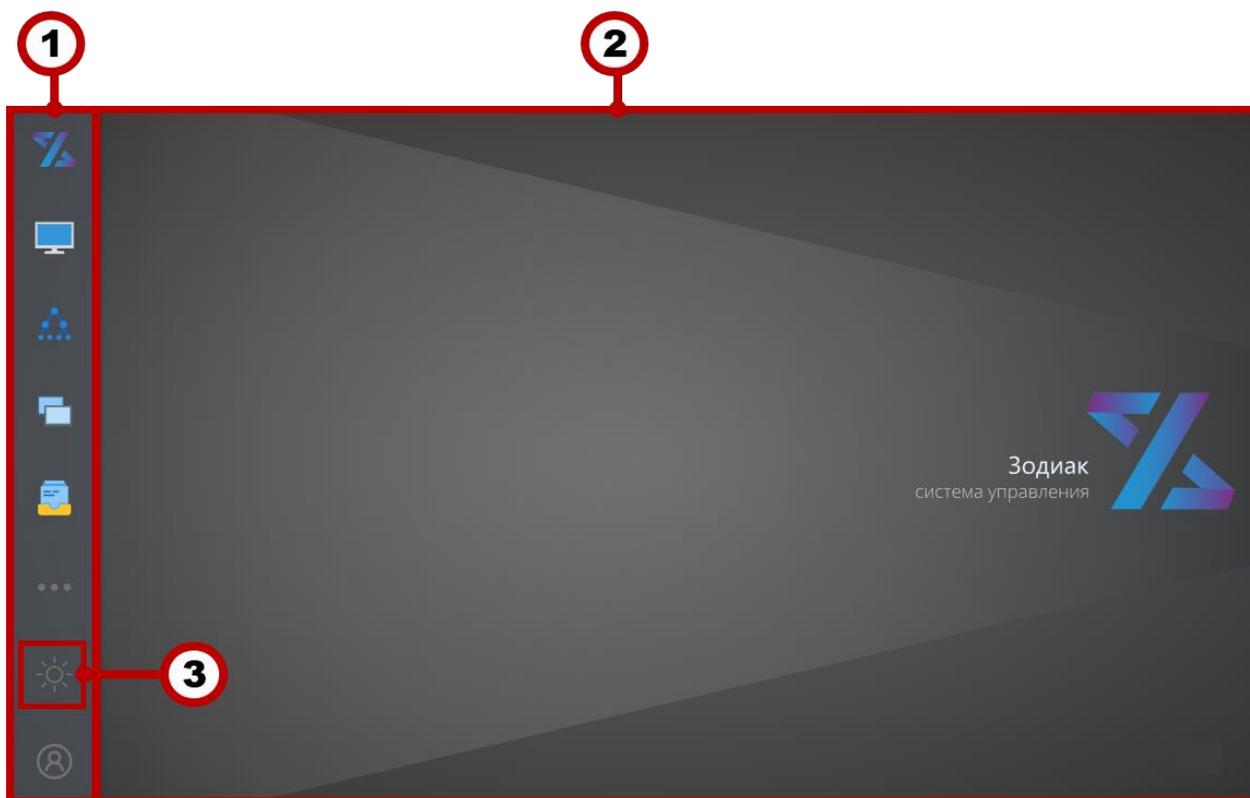


Рисунок 2.3 Общая компоновка пользовательского интерфейса системы «Зодиак.АйТиЭм»

1. **Боковая панель** – предназначена для группировки пунктов меню, отвечающих за переключение режимов пользовательского интерфейса.
2. **Основная рабочая область системы** – предназначена для отображения логических окон, представляющих собой основной способ группировки отображаемого содержимого и осуществления взаимодействия с системой.
3. Кнопка переключения **цветовой палитры** – предназначена для переключения между темной и светлой темой.

## Примечание

Система поддерживает светлую и темную тему. Далее в руководстве будут использоваться скриншоты светлой темы для улучшения читаемости.

### 2.2.2 Управление логическими окнами

Работа пользователя с системой осуществляется посредством **логических окон** – управления их составом и состоянием, взаимодействия с их содержимым. При открытии новых окон их группировка в рамках рабочей области системы осуществляется без взаимного перекрытия, **в порядке открытия, слева направо**.

При превышении совокупным размером логических окон размера рабочей области системы часть логических окон скрывается. Доступ к скрытым логическим окнам осуществляется посредством полосы прокрутки рабочей области как показано на рисунке 2.4.

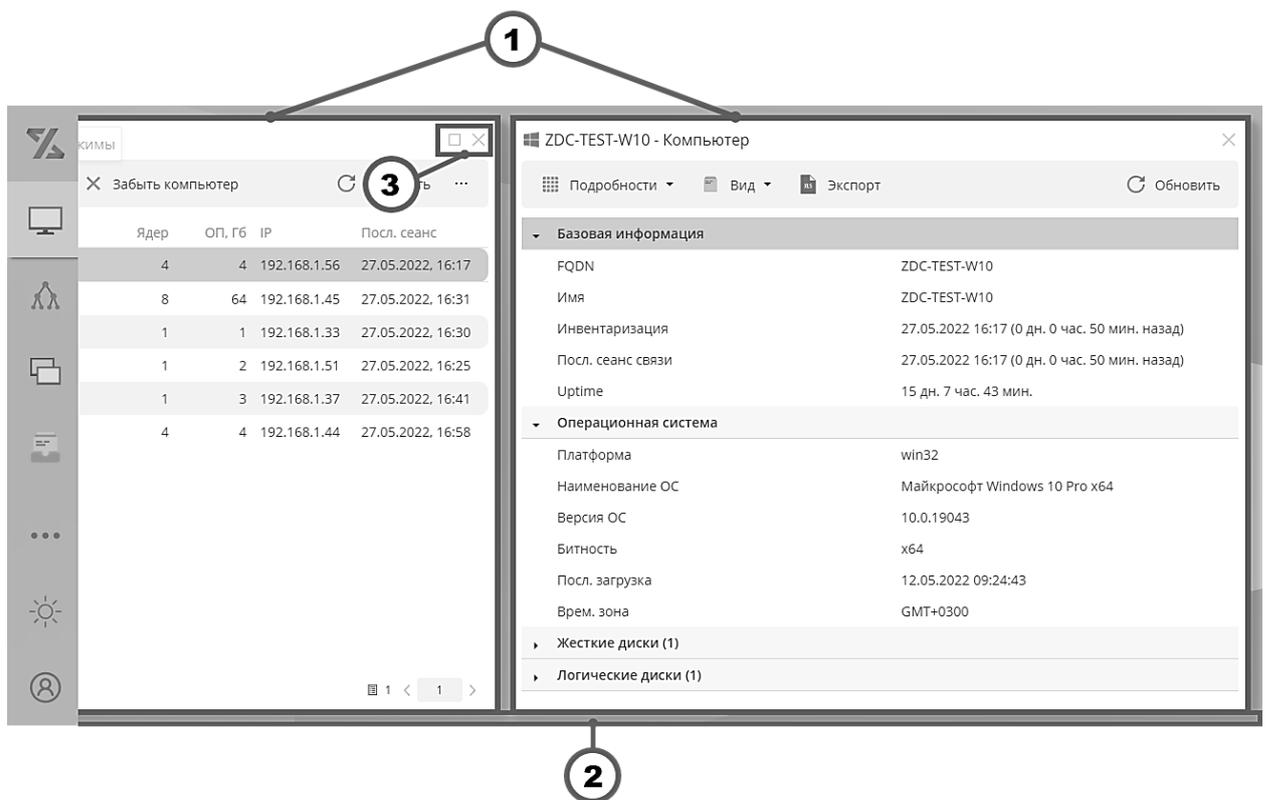


Рисунок 2.4 Пример расположения логических окон в основной рабочей области

1. Логические окна, открытые в основной рабочей области.
2. Полоса прокрутки рабочей области.
3. Блок кнопок управления состоянием окна.

### 2.2.3 Табличное представление данных

Общий вид табличного представления представлен на рисунке 2.5 .

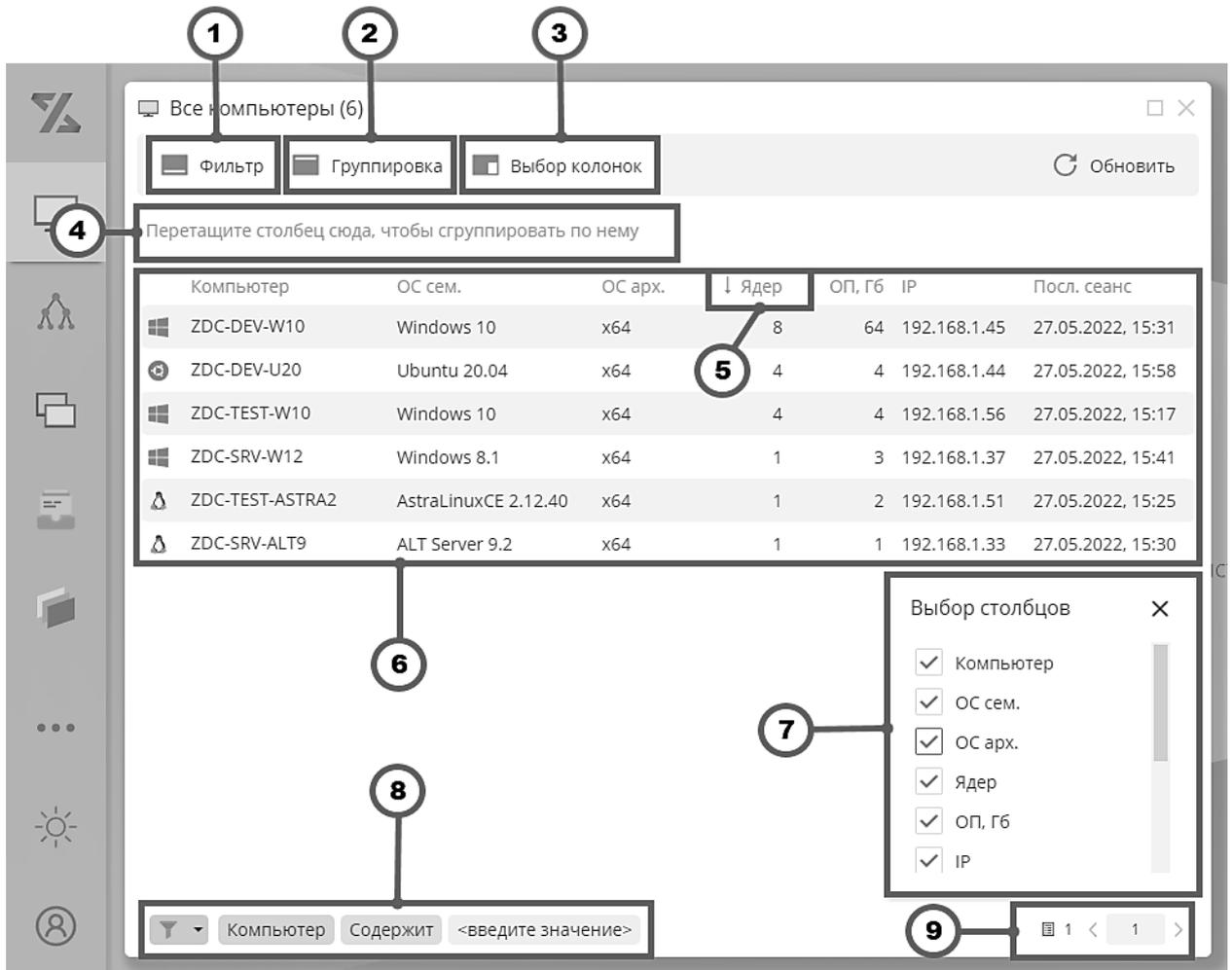


Рисунок 2.5 Общий вид табличного представления

1. Кнопка **«Фильтр»** – предназначена для включения/выключения видимости фильтра, находящегося в нижней части окна (8).
2. Кнопка **«Группировка»** – предназначена для включения/выключения видимости области группировки, находящегося над таблицей с данными (4).
3. Кнопка **«Выбор колонок»** – предназначена для включения/выключения видимости окна выбора колонок (7).
4. Область **группировки** – перетаскивание заголовка столбца из таблицы на эту область приводит к группировке таблицы по этому столбцу.
5. Пример заголовка с включенной **сортировкой** – клик по заголовку столбца переключает направление сортировки. Клик + «Ctrl» отменяет сортировку.
6. **Данные** таблицы.
7. **Окно «Выбор столбцов»** – предназначено для включения/выключения видимости столбцов в представлении.
8. **Фильтр** – предназначен для фильтрации списка свойств по произвольным критериям. Использование фильтра описано в разделе 2.2.4.

9. **Пейджер** – предназначен для переключения между страницами табличного представления и отображения номера текущей страницы.

#### 2.2.4 Конструктор фильтра

Конструктор фильтра позволяет пользователю строить сложные критерии отбора с неограниченным числом условий в комбинации с логическими операциями.

Для вызова конструктора фильтра табличного представления включите видимость фильтра переключателем **«Фильтр»** и нажмите кнопку как показано на рисунке 2.6.

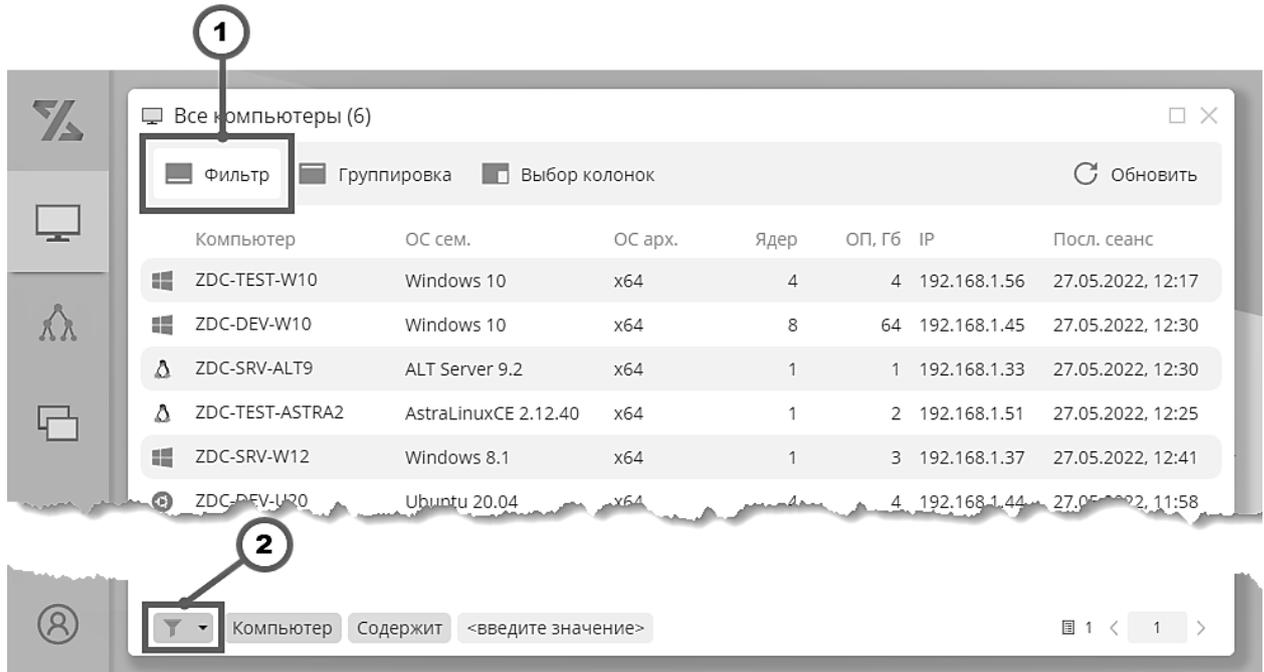


Рисунок 2.6 Вызов окна конструктора фильтра

1. Кнопка **«Фильтр»** – предназначена для включения/выключения видимости фильтра, находящегося в нижней части окна.
2. Кнопка вызова **конструктора фильтра** – вызывает окно, содержащее редактор древовидного списка критериев.

Общий вид конструктора фильтра приведен на рисунке 2.7.

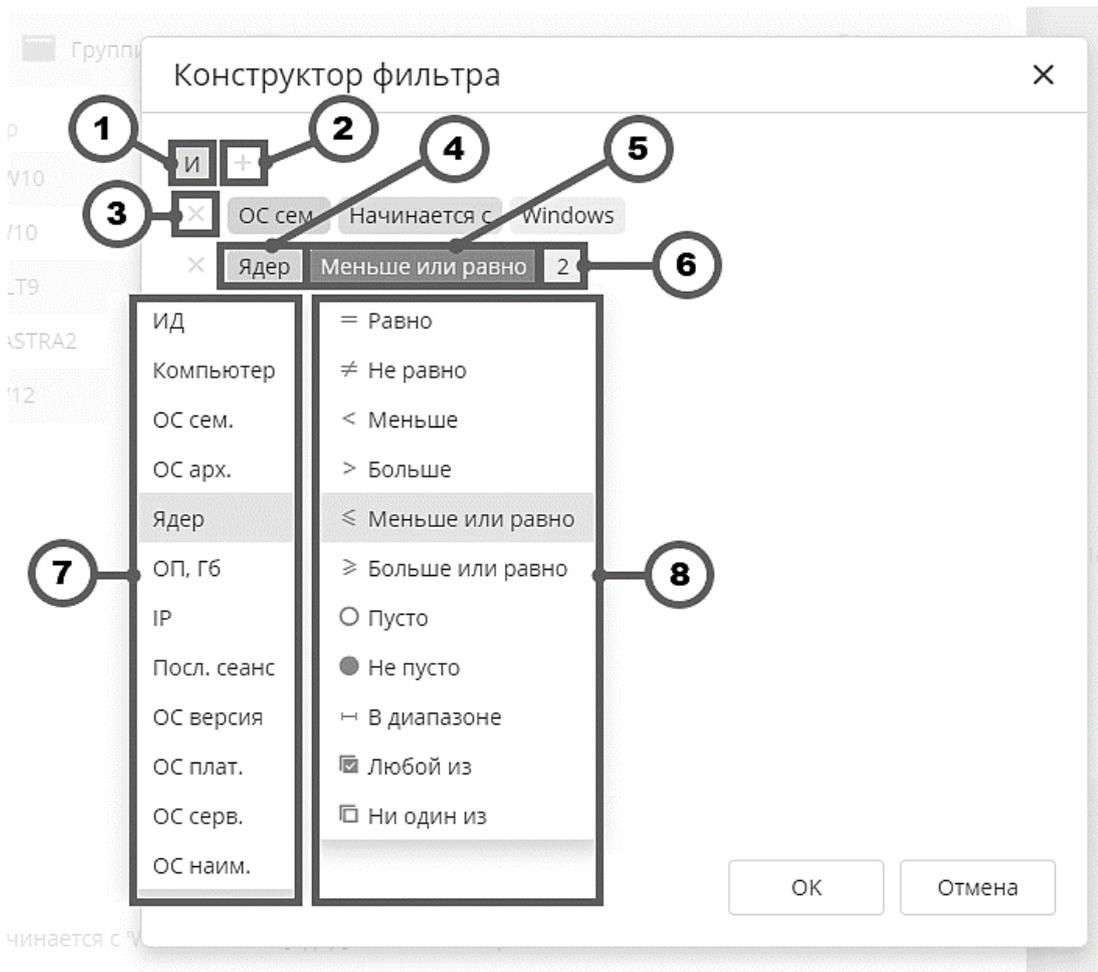


Рисунок 2.7 Окно конструктора фильтра

1. Кнопка выбора **логической операции**.
2. Кнопка **добавления** критерия отбора для логической операции.
3. Кнопка **удаления** критерия отбора.
4. **Окно выбора поля** для критерия отбора.
5. **Окно выбора условия** для критерия отбора.
6. Окно **значения** поля в условии.
7. Выпадающий **список выбора поля** – строится из набора столбцов таблицы. В нашем примере – это столбцы списка компьютеров.
8. Выпадающий **список выбора условия** – зависит от типа значений поля. В нашем примере – это условия для целых чисел (количество ядер процессора).

Табличное представление с примененным фильтром представлено на рисунке 2.8.



Рисунок 2.8 Табличное представление с примененными критериями отбора

1. Кнопка с названием-расшифровкой примененных критериев отбора – вызывает окно редактора фильтра с **текущими** критериями фильтрации.
2. Кнопка «Очистить» – сбрасывает примененные критерии отбора и возвращает табличное представление в исходное состояние.

### 2.2.5 Конструктор расписания

Конструктор расписания используется на формах, где требуется задавать регулярные моменты времени. Общий вид конструктора расписания приведен на рисунке 2.9.

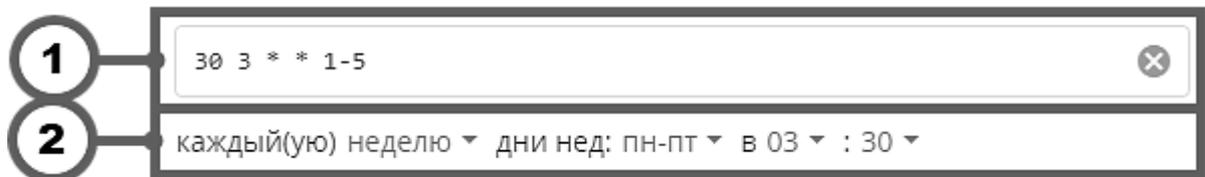


Рисунок 2.9 Общий вид конструктора расписания.

1. Поле ввода маски в формате **cron**. Ввод текста синхронизирован со строкой расшифровки.
2. Строка расшифровки маски **cron** с выпадающими списками выбора значений месяцев, дней недели, часов, минут и т.п. Выбор значений синхронизирован с полем ввода маски.

Расшифровка позиций маски **cron**:

```
* * * * *
| | | | |
| | | | ----- день недели (0-7) (воскресенье = 0 или 7)
| | | ----- месяц (1-12)
| | ----- день (1-31)
| ----- час (0-23)
----- минута (0-59)
```

Примеры задания расписания:

```
* * * * * - каждую минуту
*/5 * * * * - каждые 5 минут
0 0-23 * * * - каждый час в 0 минут
5 0 * * * - каждый день в 0 часов 5 минут
30 3 * * 1-5 - каждую неделю с понедельника по пятницу в 3:30
15 14 1 * * - 1 числа каждого месяца в 14 часов 15 минут
15 10,13 * * 1,4 - в понедельник и четверг в 10:15 и 13:15
```

## 3. РЕЖИМЫ ВЕБ-КОНСОЛИ УПРАВЛЕНИЯ

### 3.1 Компьютеры

#### **i** Примечание

Режим «**Компьютеры**» предназначен для использования супер-администратором и/или для машин, которые находятся в рамках федеральных администраторов. Для остальных машин должен быть доступен только режим «**Группы**». При инсталляции данный режим должен быть включен только у тех ролей, которым нужно выдать доступ ко всем машинам.

При активизации режима «Компьютеры» открывается логическое окно «Все компьютеры», общий вид которого приведен на рисунке 3.1.

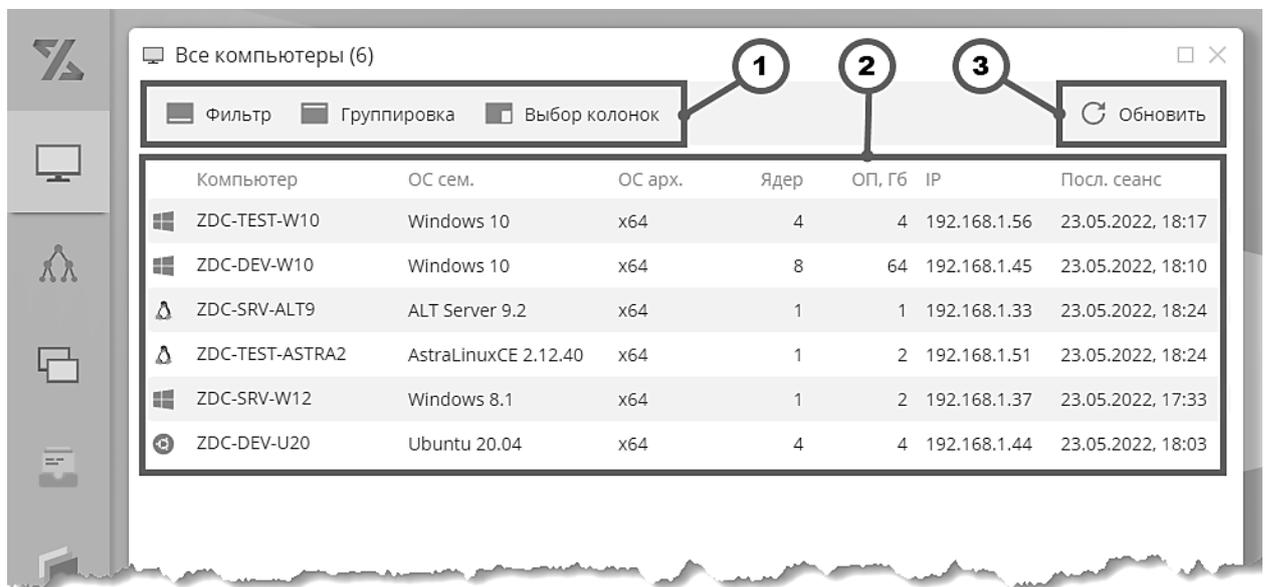


Рисунок 3.1 Общий вид окна «Все компьютеры» ниже

1. **Панель инструментов** управления табличным представлением данных (более подробное описание – в разделе 2.2.3).
2. **Список компьютеров** – отображает данные базовой инвентаризации компьютеров, полученные от установленных на них агентов.

#### **i** Примечание

Именно в этом списке появляется компьютер до назначения ему определенной группы.

3. Кнопка «**Обновить**» – предназначена для обновления списка компьютеров актуальными данными, приходящими от агентов.

Выбор компьютера в списке делает доступными дополнительные действия для данных этого компьютера – через панель инструментов или контекстное меню (рисунок 3.2).

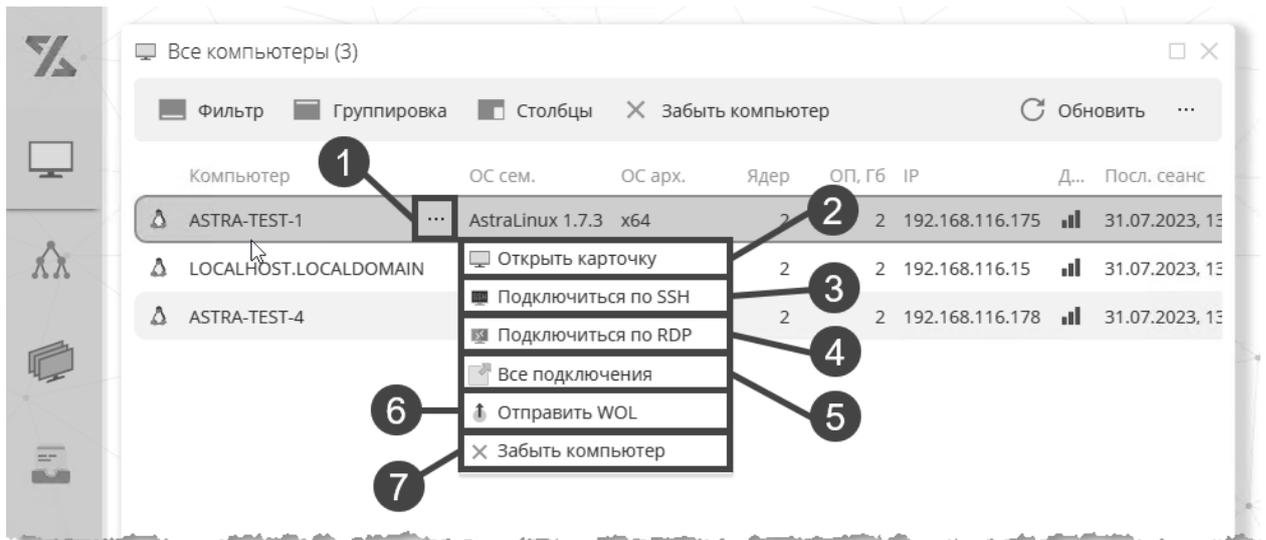


Рисунок 3.2 Вызов действий для выбранного компьютера через контекстное меню

1. Кнопка «...» – предназначена для вызова контекстного меню.
2. Кнопка «**Открыть карточку**» – предназначена для вызова логического окна карточки компьютера (раздел 3.1.1). Карточка компьютера вызывается также двойным кликом по строчке компьютера в списке.
3. Кнопка «**Подключиться по SSH**» - предназначена для вызова логического окна подключения по SSH (раздел 3.8).
4. Кнопка «**Подключиться по RDP**» - предназначена для вызова логического окна подключения по RDP (раздел 3.8).
5. Кнопка «**Все подключения**» - предназначена для вызова логического окна со списком возможных подключений к этому компьютеру.
6. Кнопка «**Отправить WOL**» - предназначена для отправки команды «Wake-on-LAN» данному компьютеру.
7. Кнопка «**Забудь компьютер**» – предназначена для удаления данных компьютера в случае, если данные утратили актуальность.

### 3.1.1 Карточка компьютера

Карточка компьютера по умолчанию открывается в режиме «**Сводка**» (рисунок 3.3).

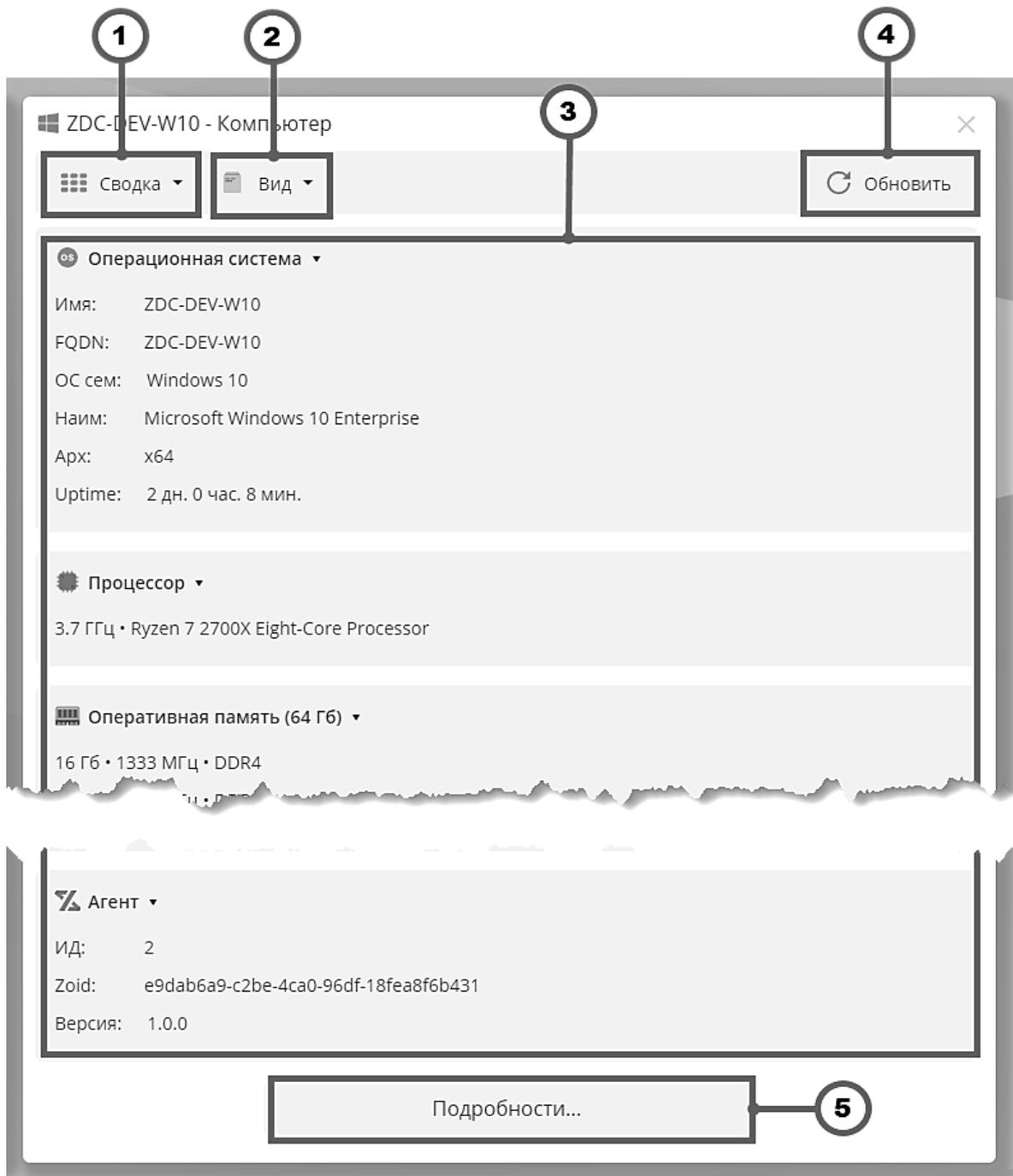


Рисунок 3.3 Карточка компьютера в режиме «Сводка»

1. Переключатель **режима карточки** – предназначен для переключения карточки компьютера между режимами «Сводка», «Подробности», «Группы», «Назначения».
2. Переключатель вида области данных – переключает панели области данных между состояниями «Развернуть все» и «Свернуть все».
3. **Область данных** – предназначена для отображения сводных данных базовой инвентаризации компьютера.
4. Кнопка **«Обновить»** – предназначена для обновления карточки компьютера актуальными данными, приходящими от агентов.

5. Кнопка «**Подробности**» – открывает карточку компьютера в режиме «**Подробности**». Карточка компьютера в режиме «**Подробности**» (рисунок 3.4) содержит все данные базовой инвентаризации компьютера.

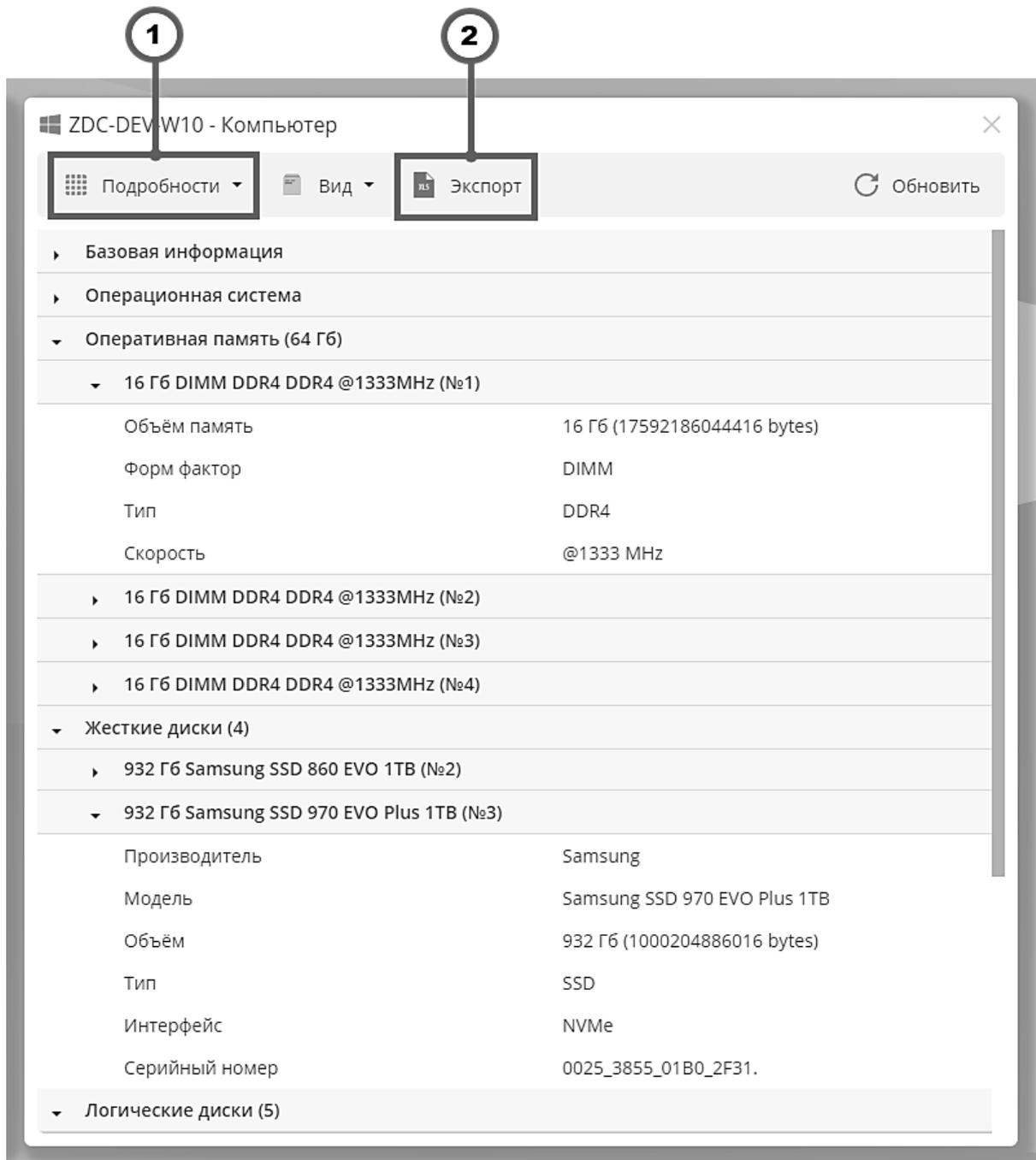


Рисунок 3.4 Карточка компьютера в режиме «**Подробности**»

1. Переключатель **режима карточки** – предназначен для переключения карточки компьютера между режимами «**Сводка**», «**Подробности**», «**Группы**», «**Назначения**».
2. Кнопка «**Экспорт**» – предназначена для сохранения данных в файле формата MS Excel.

### 3.1.2 Просмотр назначений

В режиме «**Назначения**» карточка компьютера позволяет просматривать информацию о назначенных на компьютерах директивах и использованных для этого ассоциациях.

Например, на рисунке 3.5 приведена ситуация, когда скрипт PowerShell с именем «Скрипт1», являющийся элементом директивы «Директива 1», назначен на компьютер «ZDC-TEST-W10» с помощью ассоциации «Ассоциация 1».

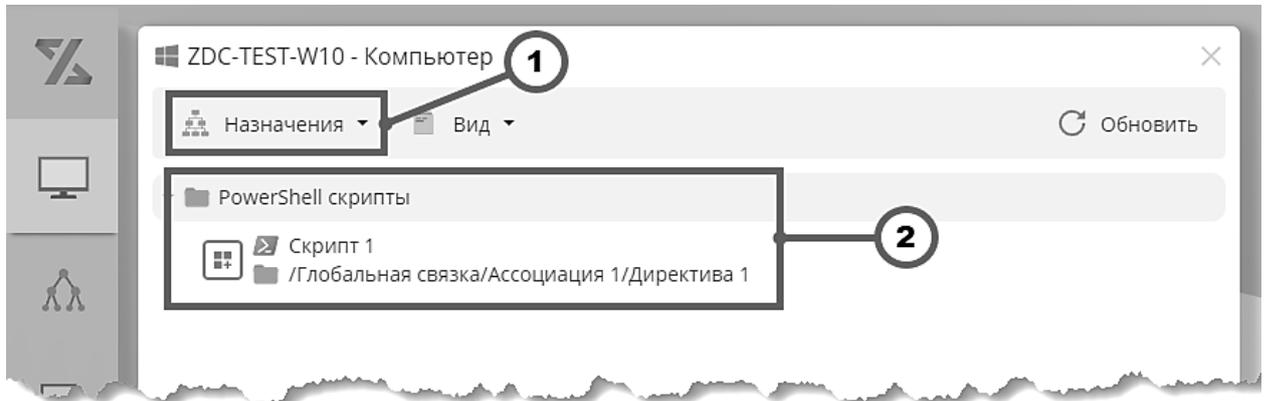


Рисунок 3.5 Просмотр назначений, примененных к компьютеру

1. Переключатель **режима карточки** – предназначен для переключения карточки компьютера между режимами «**Сводка**», «**Подробности**», «**Группы**», «**Назначения**».
2. **Панель назначения** – предназначена для отображения информации о назначенных на компьютер директивах.

---

#### Совет

Для быстрого перехода к назначенному на компьютер элементу директивы сделайте двойной клик по панели назначения – элемент директивы откроется в новом логическом окне.

---

## 3.2 Ассоциации

### 3.2.1 Каталог ассоциаций

Общий вид каталога ассоциаций представлен на рисунке 3.6.

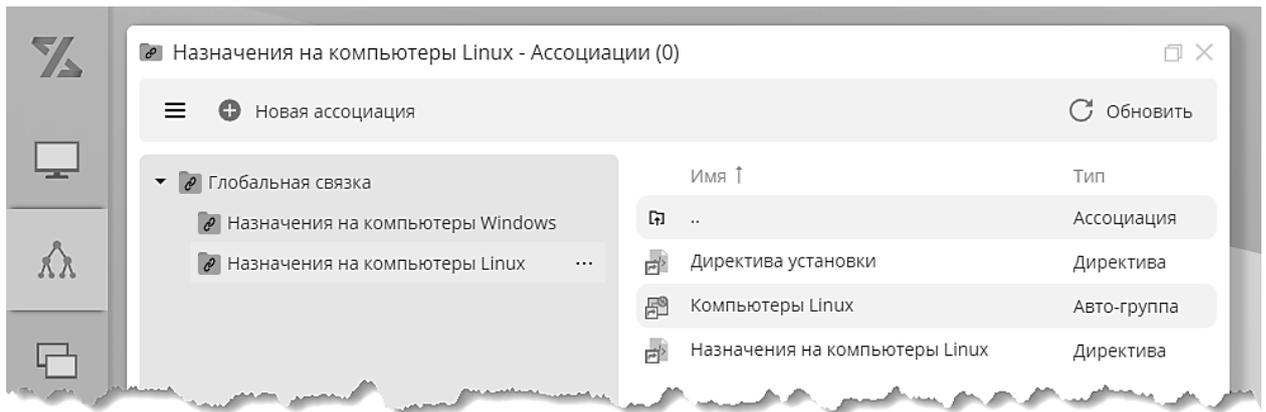


Рисунок 3.6 Общий вид каталога ассоциаций

### 3.2.2 Создание ассоциации

Для создания ассоциации перейдите в каталог ассоциаций и на панели инструментов вызовите окно создания ассоциации как показано на рисунке 3.7.

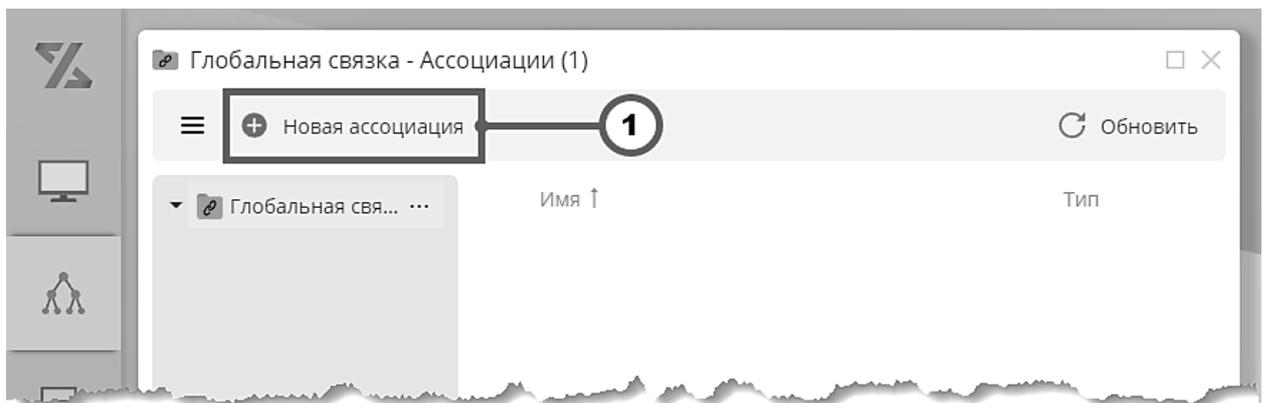


Рисунок 3.7 Вызов окна создания новой ассоциации

1. Кнопка «**Новая ассоциация**» – предназначена для вызова окна создания новой ассоциации.

В окне создания ассоциации введите имя ассоциации и нажмите «ОК» (рис. 3.8).

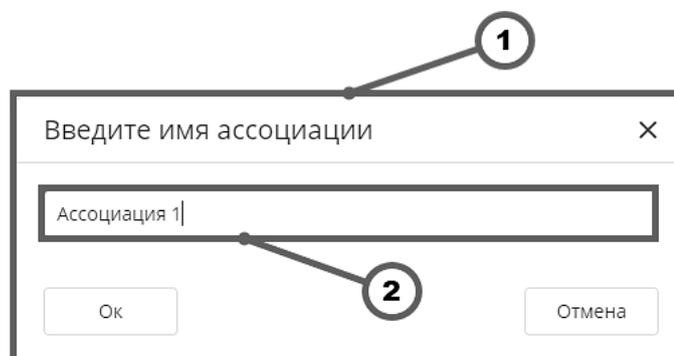


Рисунок 3.8 Окно создания ассоциации

1. Форма – предназначена для ввода полей новой ассоциации.
2. Поле ввода – предназначено для ввода имени новой ассоциации.

В результате появится логическое окно ассоциации без элементов. Добавление элементов ассоциации описано в разделе 3.2.3. Нажатие кнопки **«Сохранить»** ассоциации без элементов приведет к сохранению пустой ассоциации, элементы которой можно будет добавить позднее.

### 3.2.3 Связывание групп и директив

Для добавления элементов в ассоциацию, откройте окно ассоциации, например, используя контекстное меню в дереве ассоциаций как показано на рисунке 3.9.

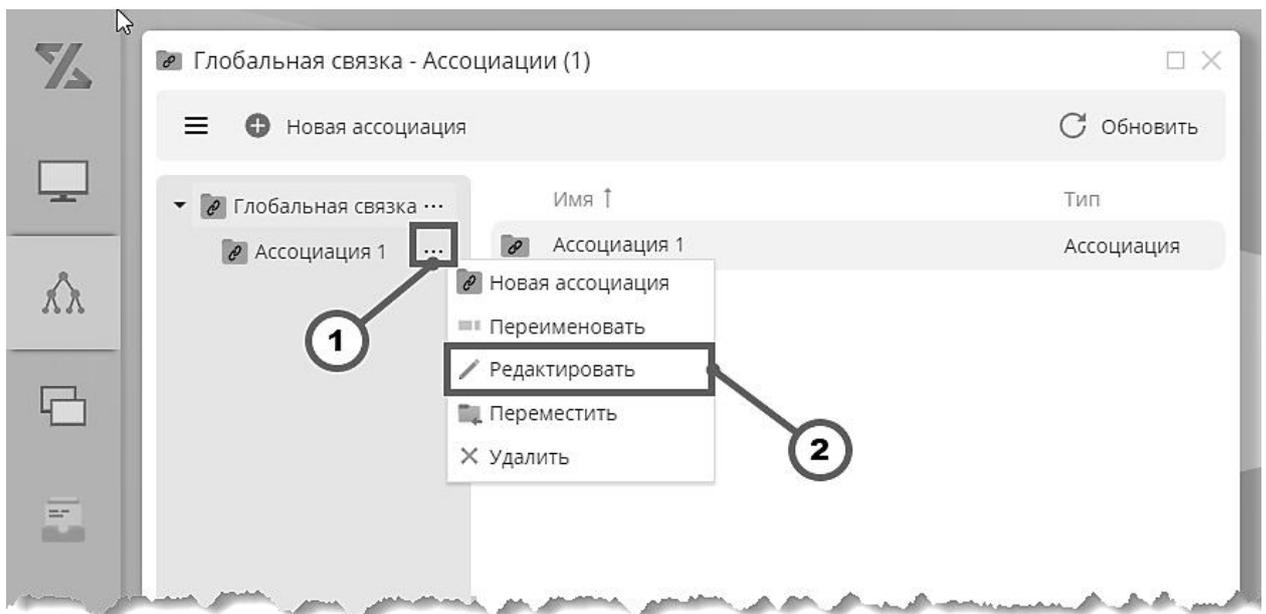


Рисунок 3.9 Вызов окна редактирования существующей ассоциации

1. Кнопка «...» – предназначена для вызова контекстного меню дерева ассоциаций.
2. Пункт контекстного **«Редактировать»** – предназначена для вызова логического окна ассоциации.

Отметьте группы и директивы, которые вы хотите связать (в нашем примере «Группа 1» и «Директива 2»), как показано на рисунке 3.10, и нажмите **«Сохранить»**.

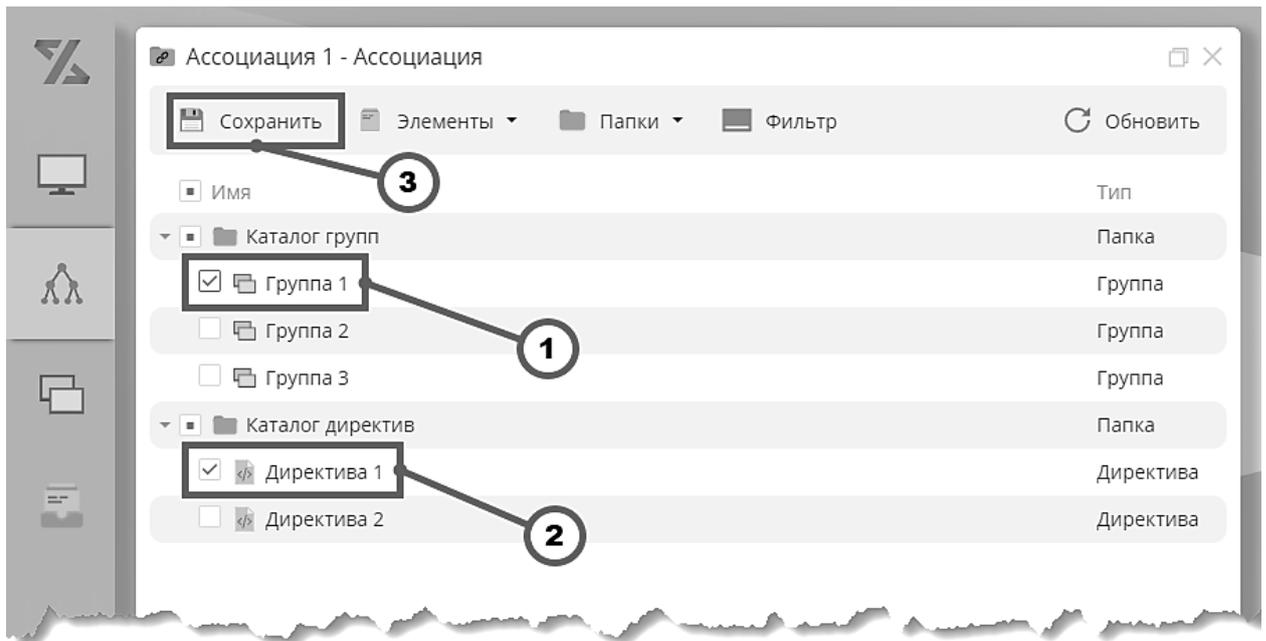


Рисунок 3.10 Добавление элементов ассоциации

1. Признак «Группа 1» – показывает, что группа «Группа 1» включена в ассоциацию.
2. Признак «Директива 1» – показывает, что директива «Директива 1» включена в ассоциацию.
3. Кнопка «Сохранить» – предназначена для сохранения всех изменений в структуре элементов ассоциации.

### 3.2.4 Глобальное связывание директив

Когда УК **в первый раз регистрируется** на сервере Зодиак.АйТиЭм, он еще не состоит ни в одной группе и даже, если он подпадает под критерии какой-либо автогруппы, он может быть включен в ее состав только через некоторое время. Таким образом, задачи из соответствующих директив будут переданы на УК только при **следующей синхронизации** УК с сервером.

В случае включения директив в **глобальную связку** задачи на УК будут переданы в момент **первой синхронизации**.



#### Совет

Использование глобальной связки оптимально на небольших контурах из нескольких машин для ускорения тестирования различных конфигураций.

Для добавления директивы в глобальную связку, откройте окно глобальной связки, например, используя контекстное меню в дереве ассоциаций как показано на рисунке 3.11.

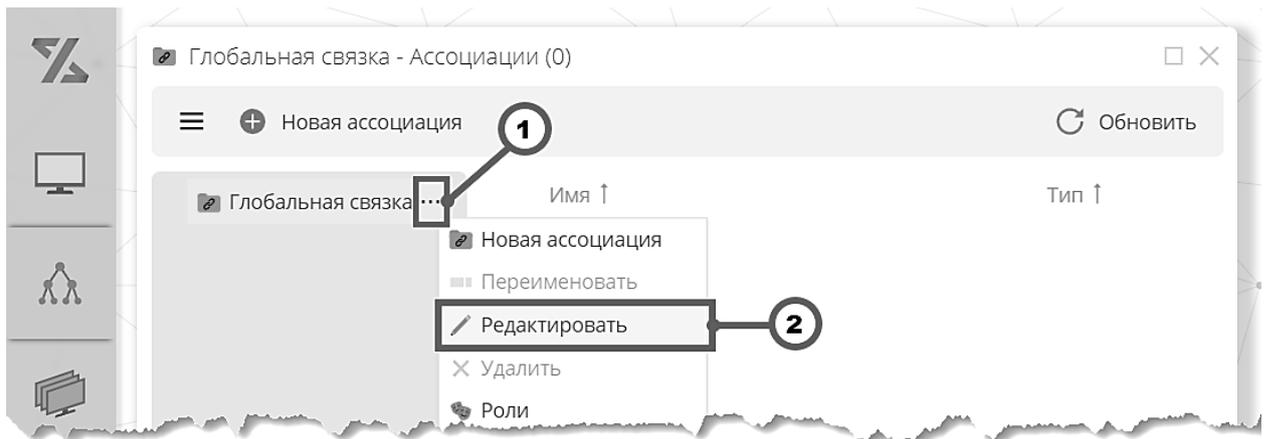


Рисунок 3.11 Вызов окна редактирования глобальной связки

1. Кнопка «...» – предназначена для вызова контекстного меню дерева ассоциаций.
2. Пункт контекстного «**Редактировать**» – предназначена для вызова логического окна ассоциации.

Отметьте директивы, которые вы хотите включить в глобальную связку (в нашем примере «Общая конфигурация»), как показано на рисунке 3.12, и нажмите «**Сохранить**».

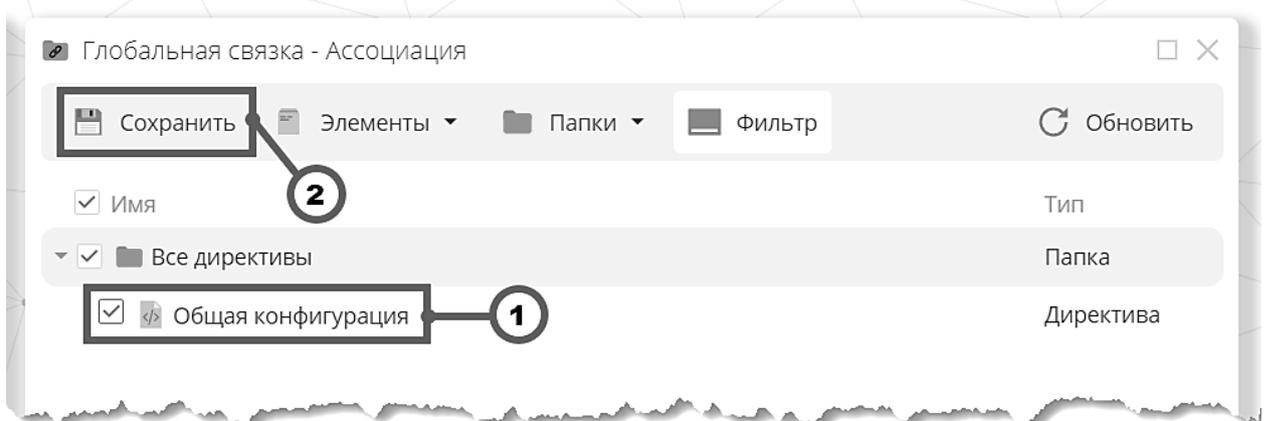


Рисунок 3.12 Добавление директивы в глобальную связку

1. Признак «**Общая конфигурация**» – показывает, что директива «Общая конфигурация» включена в глобальную связку.
2. Кнопка «**Сохранить**» – предназначена для сохранения всех изменений в структуре элементов глобальной связки.

Правильность назначения можно проверить способом, описанным в разделе 3.1.2. В нашем примере назначение на какой-либо из УК будет выглядеть, как на рисунке 6.1.

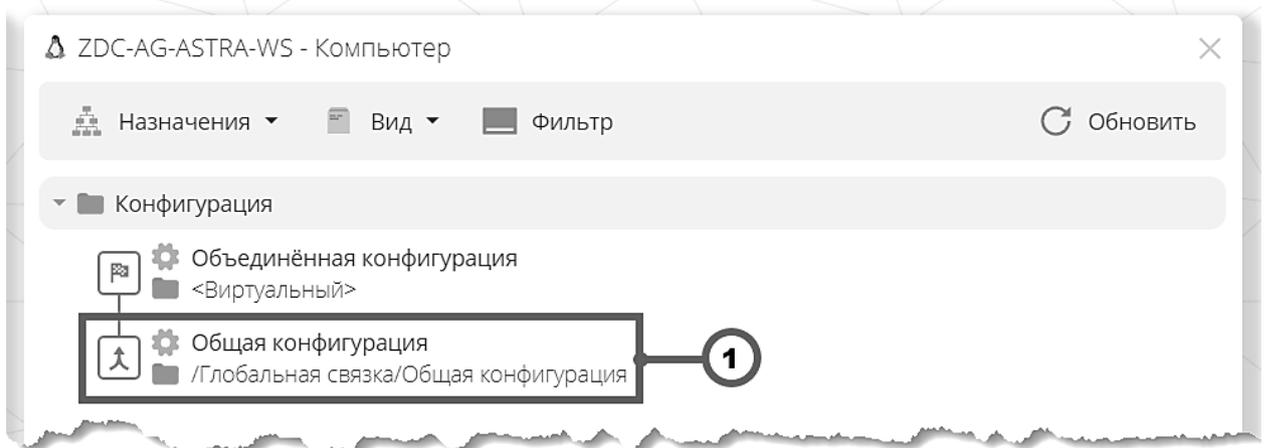


Рисунок 3.13 Просмотр назначений на УК, пришедших из глобальной связки

### ! Предупреждение

Директивы в глобальной связке применяются для всех агентов, установленных на УК организации, включая новых.

Для точек обслуживания требуется создавать специально предназначенные для ТО директивы – в том числе и в глобальной связке.

## 3.3 Группы

### 3.3.1 Каталог групп

Общий вид каталога групп представлен на рисунке 3.14.

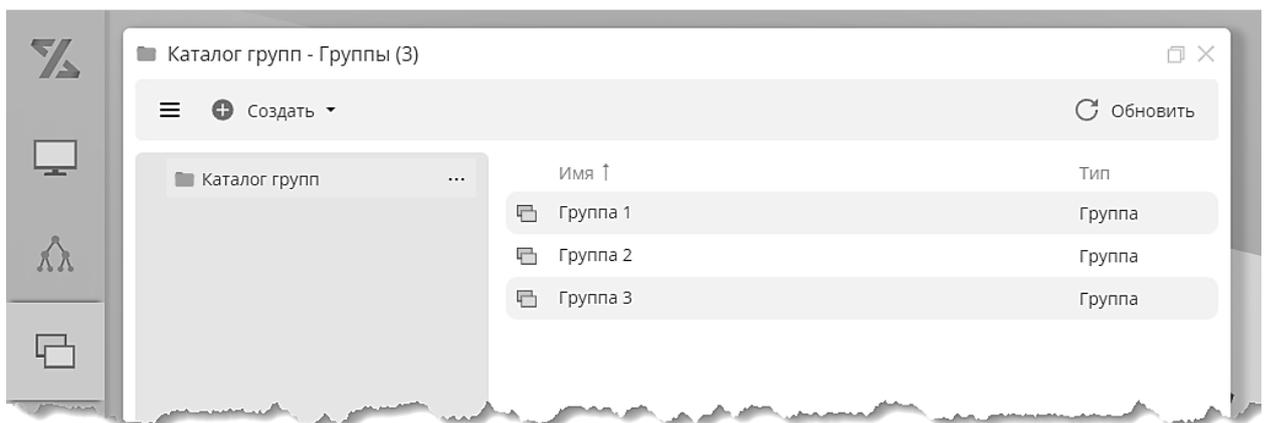


Рисунок 3.14 Общий вид каталога групп

### 3.3.2 Создание группы

Для создания группы перейдите в каталог групп и на панели инструментов вызовите окно создания группы как показано на рисунке 3.15.

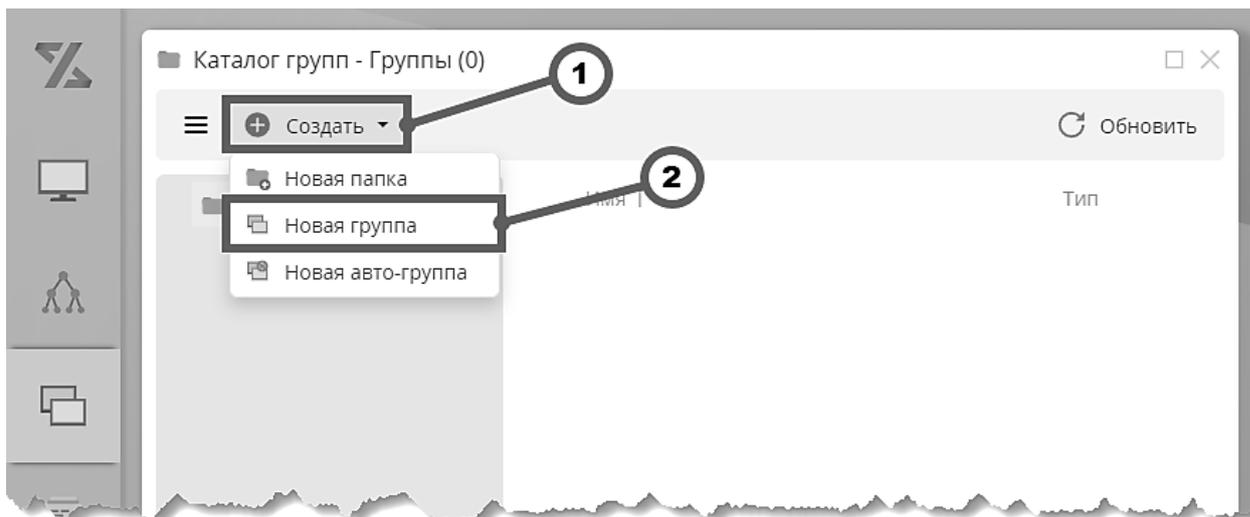


Рисунок 3.15 Вызов окна создания новой группы

1. Кнопка **«Создать»** – предназначена для вызова выпадающего списка выбора с командами управления каталогом.
  2. Кнопка **«Новая группа»** – предназначена для вызова окна создания группы.
- В окне создания группы введите имя группы и нажмите **«ОК»** (рис. 3.16).

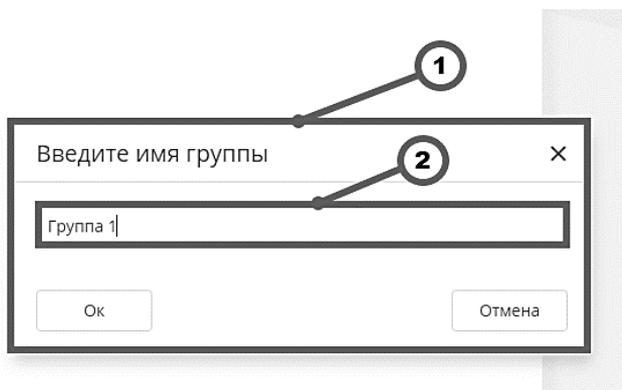


Рисунок 3.16 Окно создания группы

1. Форма – предназначена для ввода полей новой группы.
  2. Поле ввода – предназначено для ввода имени новой группы.
- Перейдите в режим редактирования как показано на рисунке 3.17.

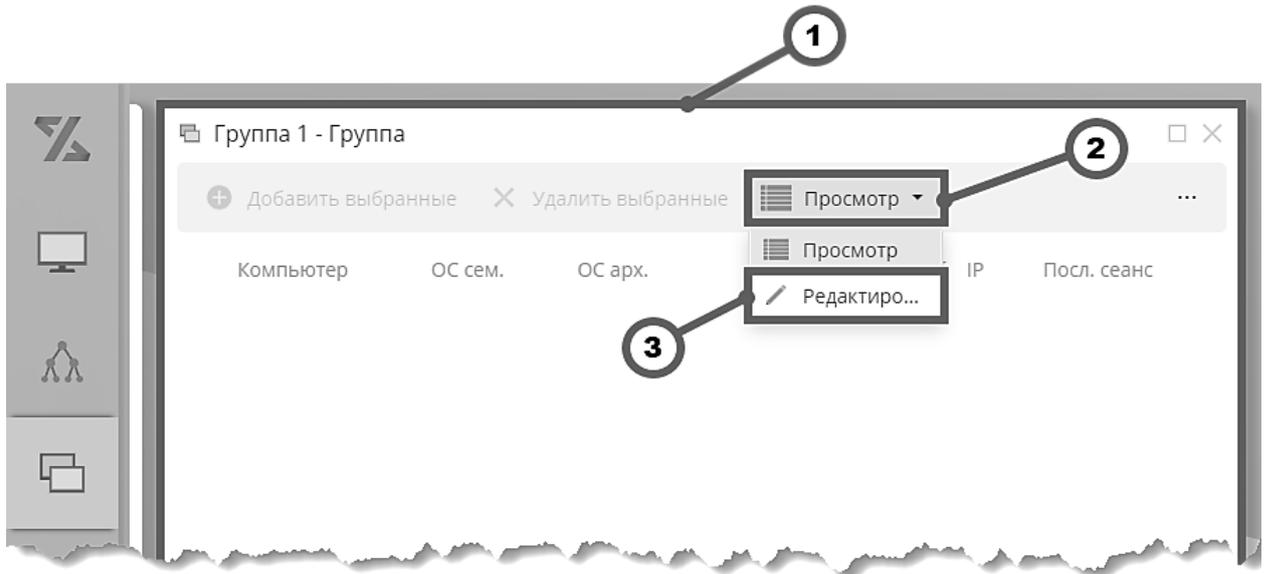


Рисунок 3.17 Включение режима редактирования группы

1. Логическое окно **«Группа1»** – окно для отображения или редактирования группы.
2. Выпадающий список режима окна группы – переключает режим окна группы между режимом **«Просмотр»** и **«Редактирование»**.
3. Кнопка **«Редактирование»** – переключает режим окна группы в режим редактирования.

Выберите компьютеры для добавления в группу, как показано на рисунке 3.18, и нажмите **«Добавить выбранные»**.

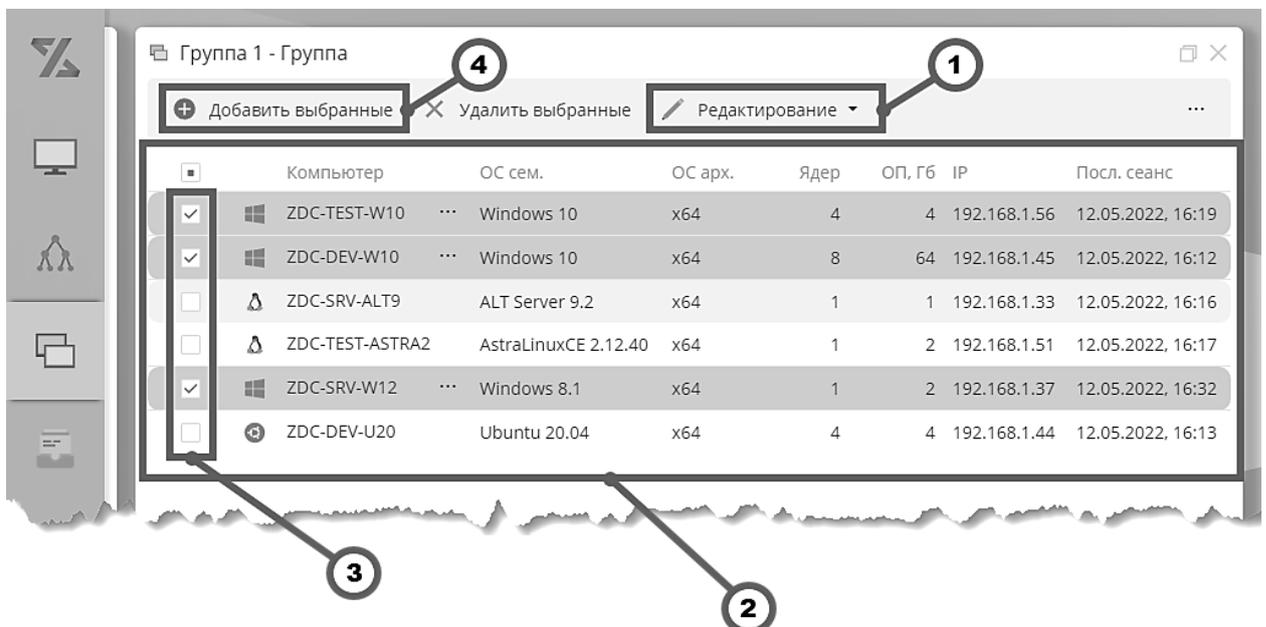


Рисунок 3.18 Добавление компьютеров в группу

1. Выпадающий список режима окна группы – переключает режим окна группы между режимом «**Просмотр**» и «**Редактирование**».

Перейдите в режим просмотра. Общий вид окна группы с добавленными компьютерами как представлен на рисунке 3.19.

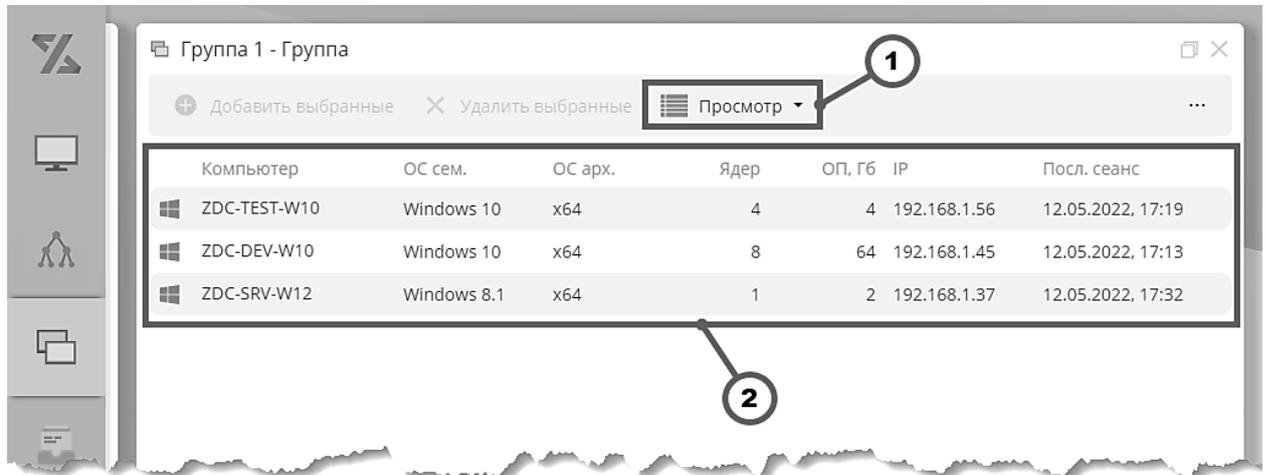


Рисунок 3.19 Включение режима просмотра группы

### 3.3.3 Авто-группы

Авто-группы компьютеров предназначены для **автоматического** поддержания своего состава в зависимости от заданных критериев.

1. Для создания авто-группы используйте кнопку «**Создать**» в панели инструментов каталога групп. В открывшемся подменю выберите пункт «**Новая авто-группа**».
2. В окне авто-группы выберите свойства и настройте критерии отбора компьютеров аналогично настройке представлений, описанной в разделе 3.5.2. Далее, нажмите «**Сохранить и обновить состав**». Примерный вид окна авто-группы в режиме настройки критериев показан на рисунке 3.20.
3. Используйте созданную авто-группу в ассоциациях для назначения директив на входящие в нее компьютеры.

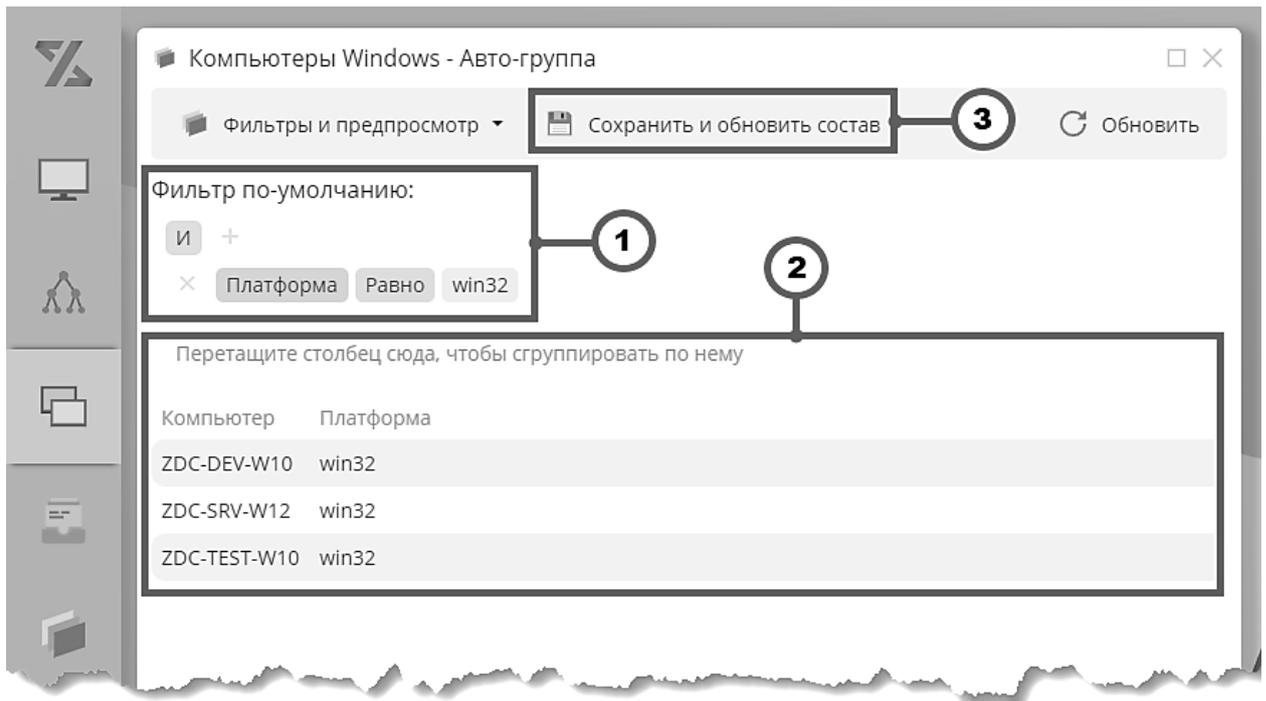


Рисунок 3.20 Окно авто-группы в режиме «Фильтры и предпросмотр»

1. **Фильтр авто-группы** – задает критерий отбора компьютеров, входящих в авто-группу. В нашем примере фильтр задает критерий, при котором в группу попадают компьютеры, для которых выполняется условие **Платформа = win32**. Использование конструктора фильтра описано в разделе 2.2.4.
2. **Область предварительного просмотра** – отображает компьютеры, которые на соответствуют выбранному критерию отбора.
3. Кнопка **«Сохранить и обновить состав»** – сохраняет настройки авто-группы и пересчитывает ее состав.

## 3.4 Директивы

### 3.4.1 Создание директивы

Для создания директивы перейдите в каталог директив и на панели инструментов вызовите окно создания директивы как показано на рисунке 3.21.



Рисунок 3.21 Вызов окна создания новой директивы

1. Кнопка **«Создать»** – предназначена для вызова выпадающего списка выбора команды создания элемента каталога директив.
  2. Кнопка **«Новая директива»** – предназначена для вызова окна создания директивы.
- В окне создания директивы введите имя директивы и нажмите «ОК» (рис. 3.22).

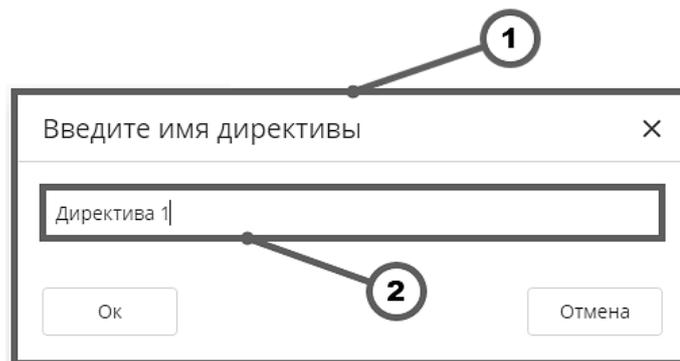


Рисунок 3.22 Окно создания директивы

1. Форма – предназначена для ввода полей новой директивы.
2. Поле ввода – предназначено для ввода имени новой директивы.

### 3.4.2 Элемент директивы типа скрипт

Для создания элемента директивы откройте окно директивы и на панели инструментов вызовите окно элемента директивы типа скрипт (в нашем примере - PowerShell) как показано на рисунке 3.23.

#### **Примечание**

Элементами директивы типа скрипт являются элементы «Javascript», «Coffescript», «PowerShell», «Shell».



Рисунок 3.23 Создание элемента директивы типа скрипт

1. Кнопка «Создать» – предназначена для вызова выпадающего списка выбора типа элемента директивы.
2. Кнопка «PowerShell» – предназначена для вызова окна элемента директивы типа скрипт PowerShell.

В окне элемента директивы введите текст скрипта как показано на рисунке 3.24.

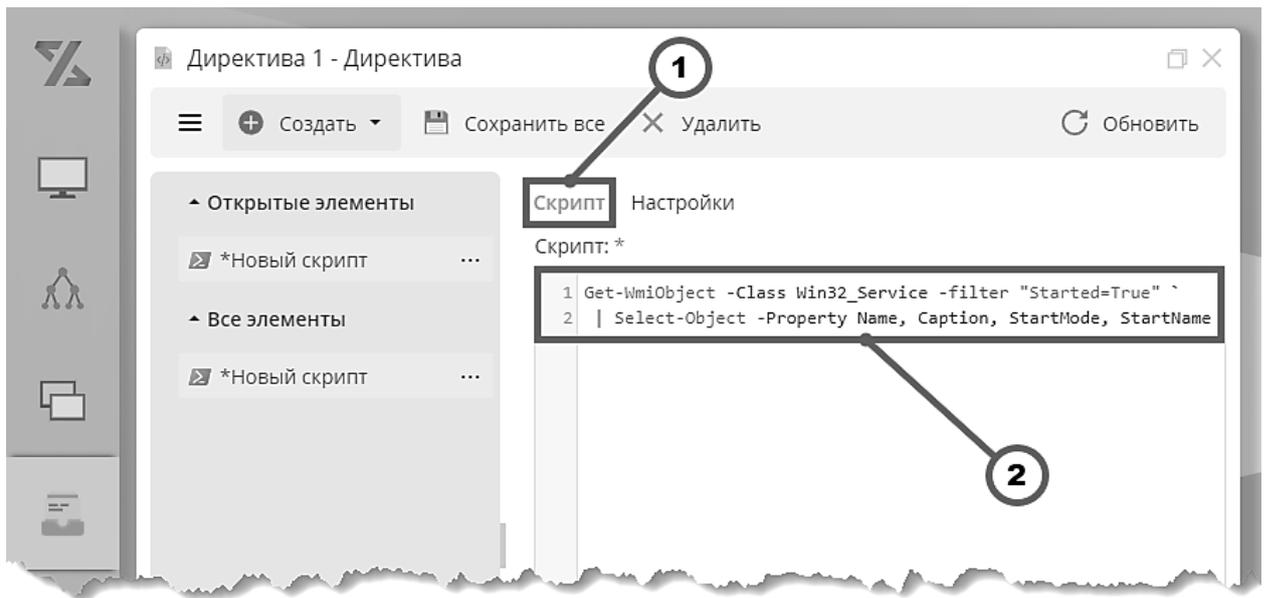


Рисунок 3.24 Ввод текста скрипта

1. Кнопка «Скрипт» – вызывает панель с полем ввода текста скрипта.
2. Поле ввода многострочного текста – предназначено для ввода текста скрипта.

Заполните форму настроек скрипта, как показано на рисунке 3.25, и нажмите кнопку «Сохранить все» на панели инструментов окна директивы.

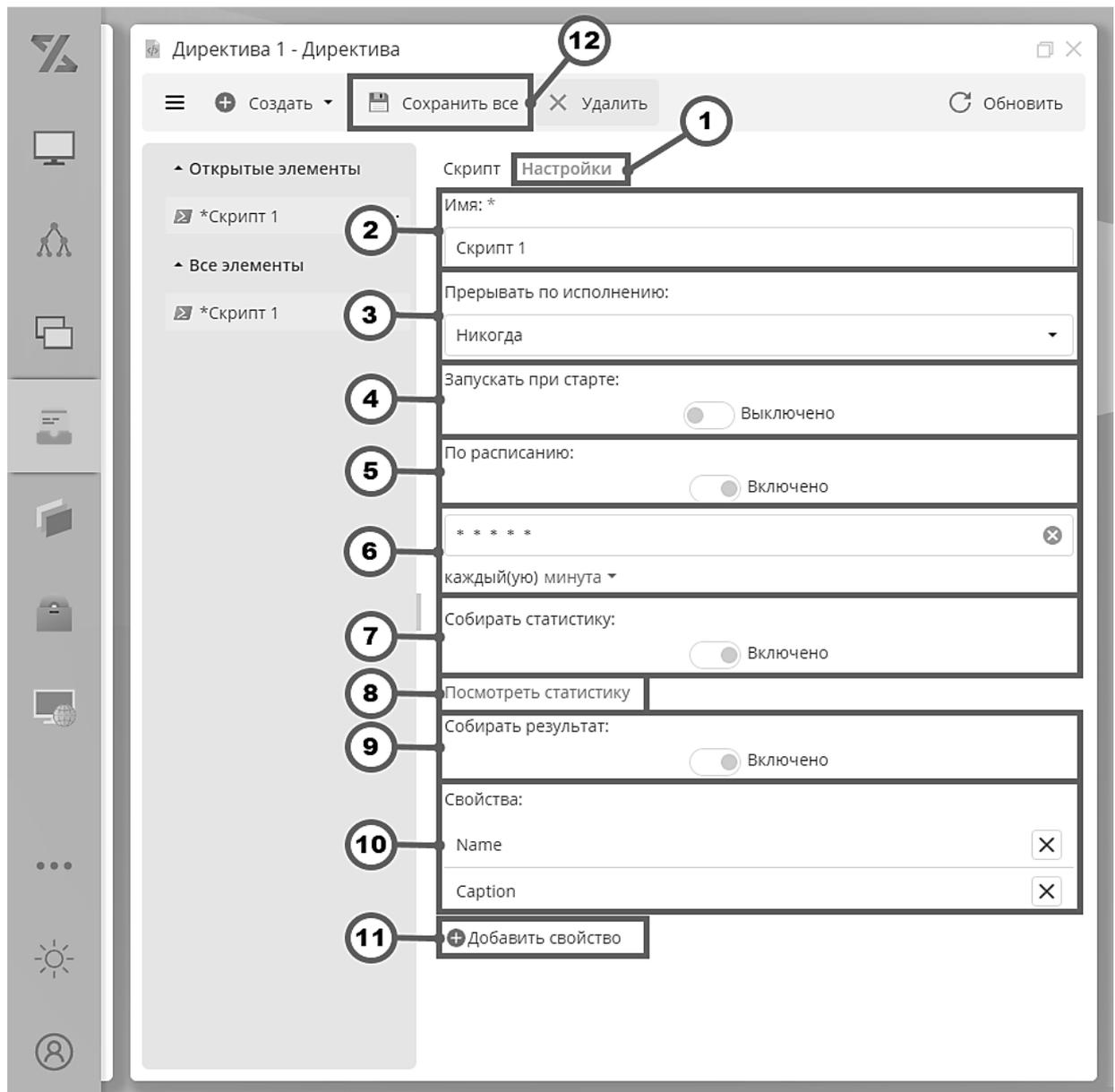


Рисунок 3.25 Заполнение формы настроек скрипта

1. Кнопка **«Настройки»** – вызывает панель с формой заполнения настроек скрипта.
2. Поле ввода **«Имя»** – предназначено для ввода наименования скрипта.
3. Выпадающий список **«Прерывать по исполнению»** – предназначен для настройки прерывания скрипта по исполнению.
4. Переключатель **«Запускать при старте»** – предназначен для переключения режима запуска скрипта при старте.
5. Переключатель **«По расписанию»** – предназначен для переключения режима запуска скрипта по расписанию.
6. Панель **настройки расписания** – предназначена для настройки моментов запуска скрипта. Доступна для ввода, если переключатель «По расписанию» находится в позиции «Включено». Подробно о работе с панелью в разделе 2.2.5.

7. Переключатель **«Собирать статистику»** – предназначен для переключения режима сбора статистики. См. также раздел о просмотре статистики 3.4.11.
8. Кнопка **«Посмотреть статистику»** – открывает логическое окно просмотра статистики работы скрипта. См. также раздел о просмотре статистики 3.4.11.
9. Переключатель **«Собирать результат»** – предназначен для переключения режима сбора результата работы скрипта. В позиции «Включено» активирует задание свойств результата, возвращаемого скриптом.
10. Список **«Свойства»** – предназначен для управления списком свойств. Свойства с указанными именами извлекаются агентом системы «Зодиак.АйТиЭм» из результата работы скрипта, возвращаются на сервер и сохраняются в БД. Просмотр возвращенных результатов обеспечивается механизмом представлений (раздел 3.4.8). Список свойств доступен при установке переключателя **«Собирать результат»** в позицию «Включено».
11. Кнопка **«Добавить свойство»** – предназначена для добавления свойств в список **«Свойства»**.
12. Кнопка **«Сохранить все»** – сохраняет все внесенные в окне директивы изменения.

### 3.4.3 Элемент директивы типа пакеты

Для создания элемента директивы откройте окно директивы и на панели инструментов вызовите окно элемента директивы типа пакеты (в нашем примере – **«Apt пакеты»**), как показано на рисунке 3.26.

---

#### **Примечание**

Элементами директивы типа пакеты являются **«Yum пакеты»** и **«Apt пакеты»**,

---

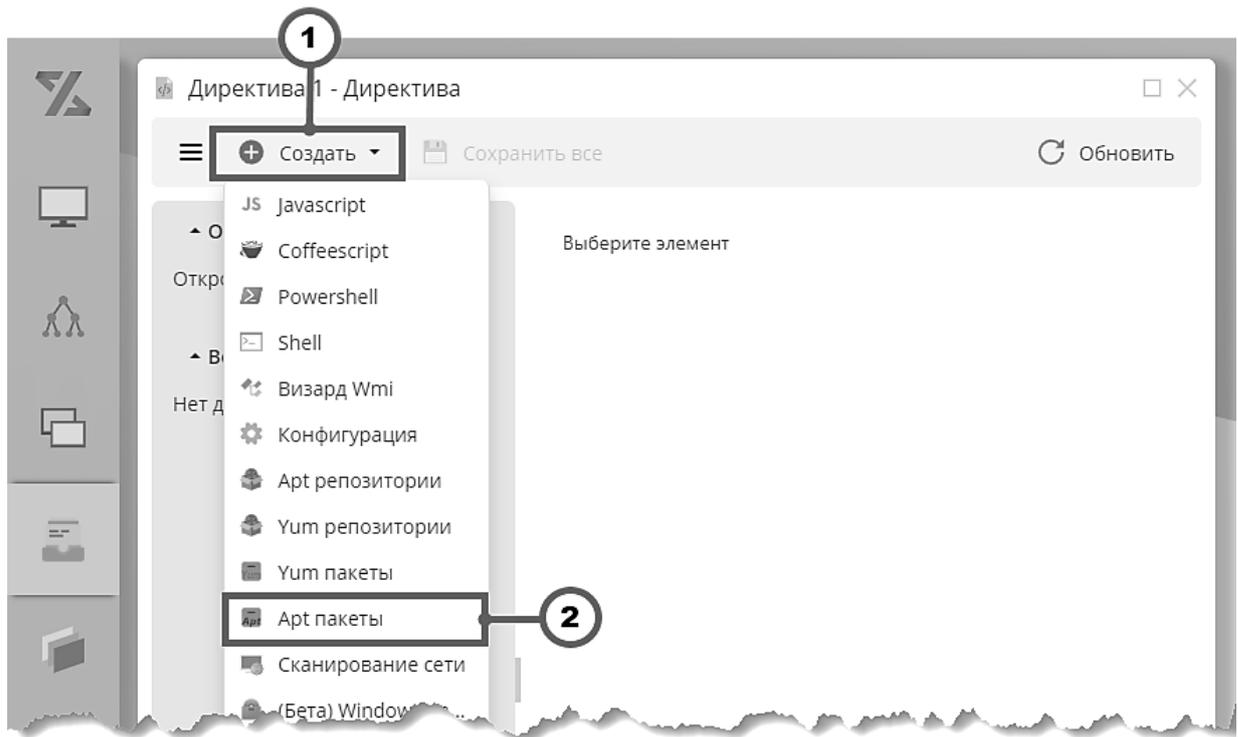


Рисунок 3.26 Создание элемента директивы типа пакеты

1. Кнопка **«Создать»** – предназначена для вызова выпадающего списка выбора типа элемента директивы.
2. Кнопка **«Арт пакеты»** – предназначена для вызова окна элемента директивы типа «Арт пакеты».

На панели **«Устанавливаемые»** вызовите контекстное меню и нажмите кнопку **«Новый»** для добавления арт-пакета, как показано на рисунке 3.27.

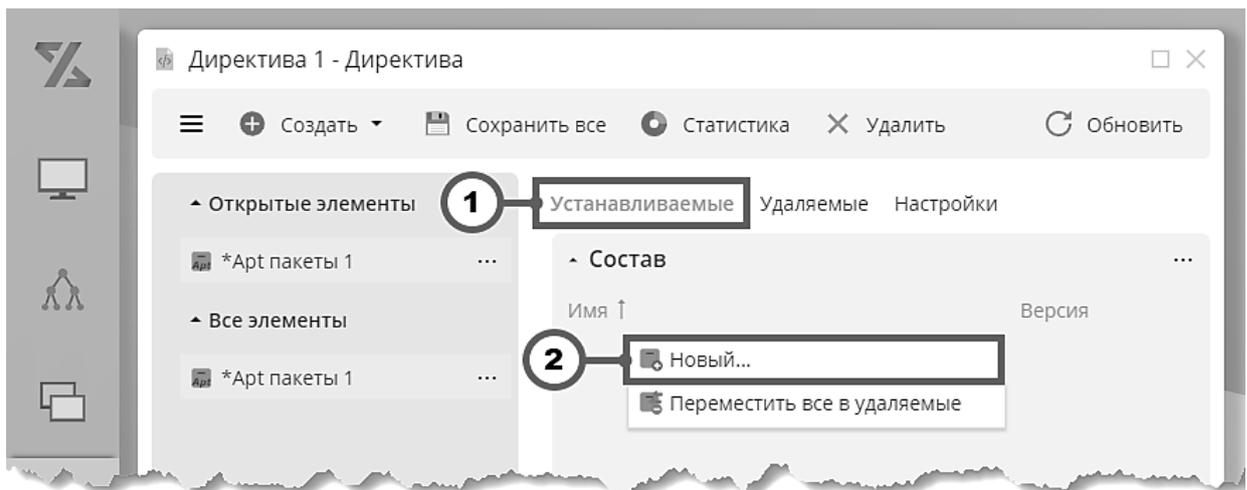


Рисунок 3.27 Добавление арт-пакета в состав директивы

1. Кнопка **«Устанавливаемые»** – вызывает панель со списком арт-пакетов, подлежащих установке.

2. Кнопка **«Новый»** – предназначена для добавления новой позиции в список арт-пакетов, подлежащих установке.

Введите имя пакета и, опционально, желаемую версию, как показано на рисунке 3.28.

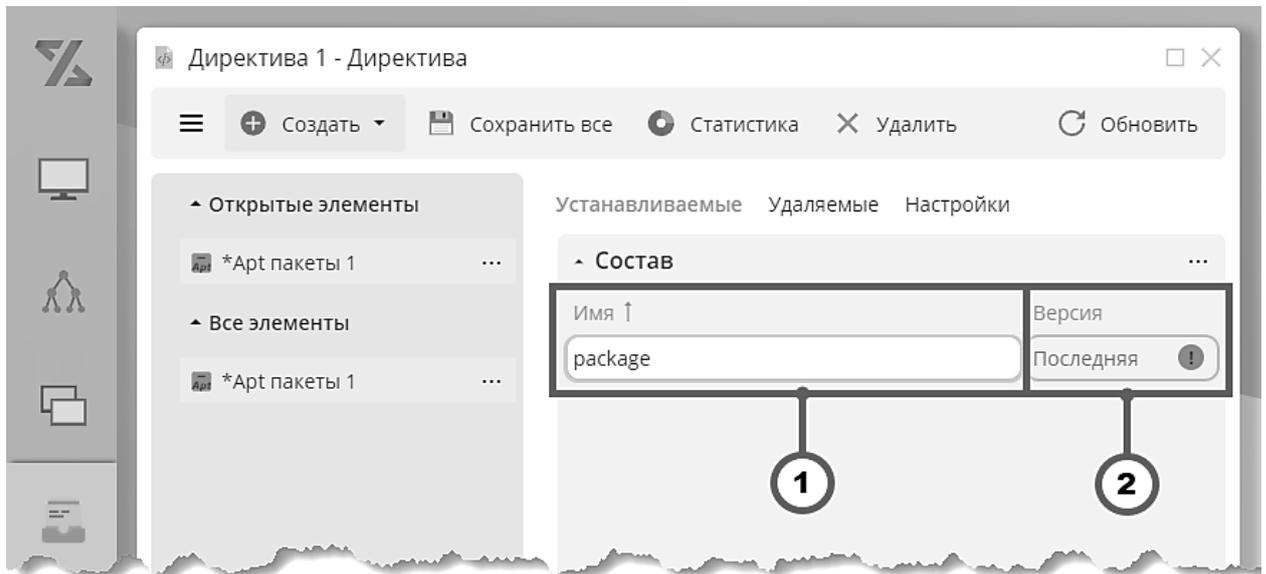


Рисунок 3.28 Задание имени и версии пакета

1. Поле ввода **«Имя»** – имя устанавливаемого пакета.
2. Поле ввода **«Версия»** – версия устанавливаемого пакета.

На панели **«Настройки»** задайте имя элемента директивы и нажмите **«Сохранить все»**, как показано на рисунке 3.29.

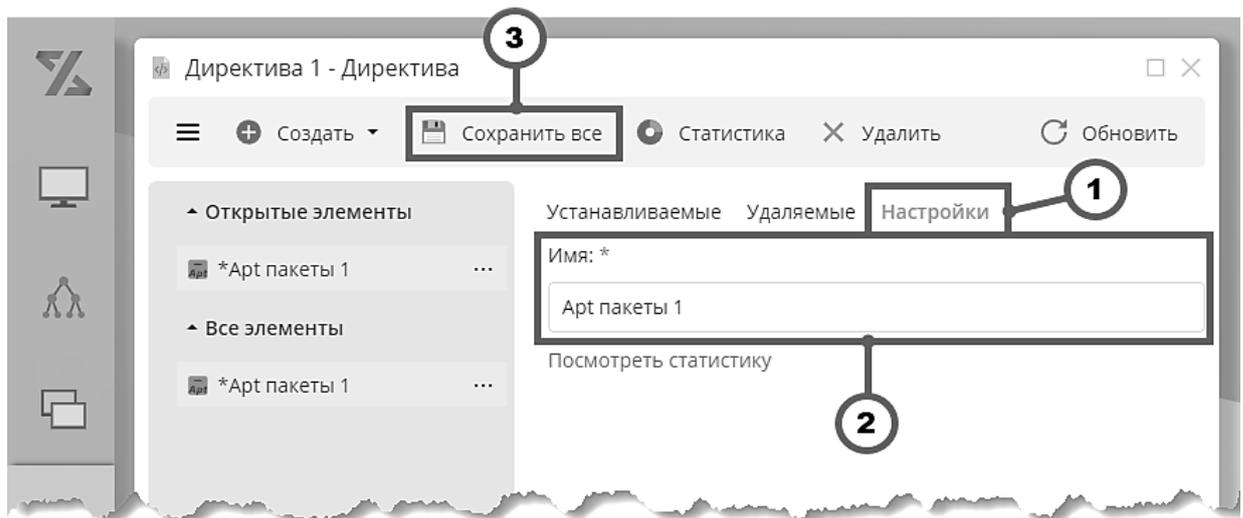


Рисунок 3.29 Настройка элемента директивы «Арт пакеты»

1. Кнопка **«Настройки»** – вызывает панель с формой настройки элемента директивы.
2. Поле ввода **«Имя»** – имя элемента директивы.

3. Кнопка **«Сохранить все»** – сохраняет все внесенные в окне директивы изменения

### 3.4.4 Элемент директивы типа репозитории

Для создания элемента директивы откройте окно директивы и на панели инструментов вызовите окно элемента директивы типа пакеты (в нашем примере – **«Apt репозитории»**), как показано на рисунке 3.30.

#### **Примечание**

Элементами директивы типа репозитории являются **«Apt репозитории»** и **«Yum репозитории»**

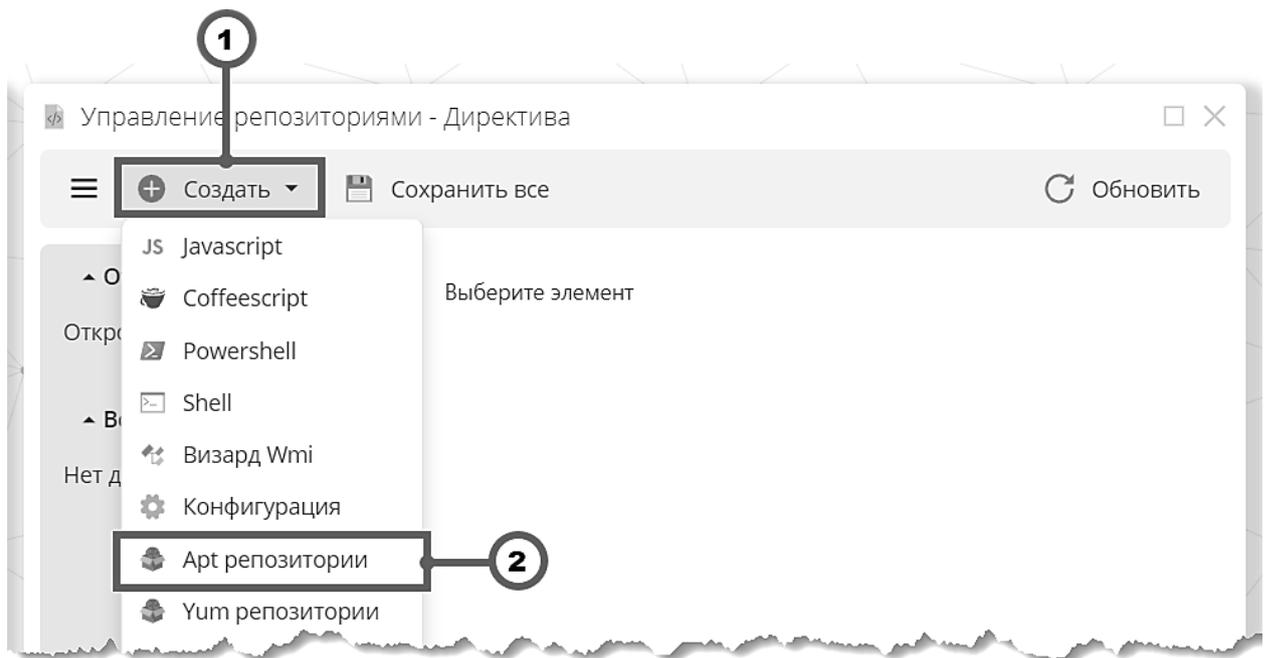


Рисунок 3.30 Создание элемента директивы типа репозитории

1. Кнопка **«Создать»** – предназначена для вызова выпадающего списка выбора типа элемента директивы.
2. Кнопка **«Apt репозитории»** – предназначена для вызова окна элемента директивы типа «Apt репозитории».

На панели **«Конфигурация»** задайте текст файла репозитория и нажмите **«Сохранить все»**, как показано на рисунке 3.31.

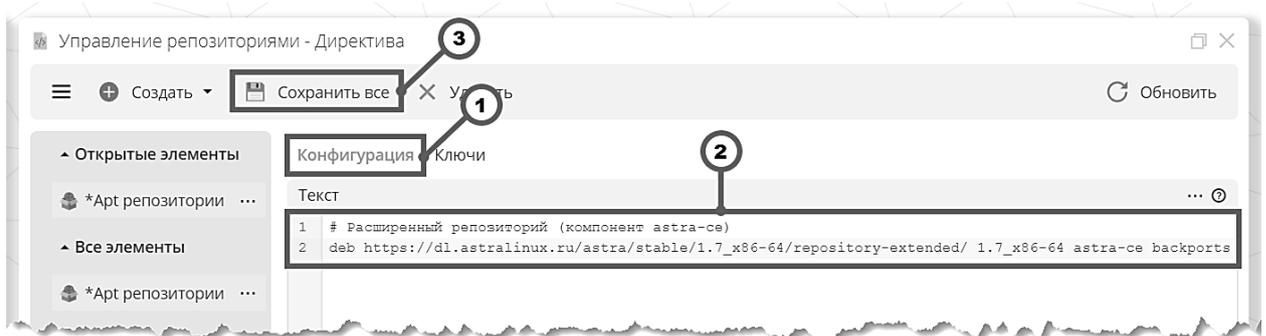


Рисунок 3.31 Настройка элемента директивы типа репозитории

1. Кнопка **«Конфигурация»** – вызывает панель с редактором текста файла репозитория.
2. Пример текста файла репозитория.
3. Кнопка **«Сохранить все»** – сохраняет все внесенные в окне директивы изменения

### **i** Примечание

Во избежание риска потерять какую-то ручную настройку локального администратора мы не перезаписываем стандартный файл `/etc/apt/sources.list`. Вместо этого создается отдельный файл `/etc/apt/sources.list.d/zodiac.list`.

В случае необходимости модифицировать стандартный файл репозитория, вы можете реализовать собственный сценарий управления посредством скрипта **shell**. Например, следующая команда добавляет запись в стандартный файл:

```
echo deb cdrom:[OS Astra Linux 1.7.0 1.7_x86-64 DVD ]/ \
1.7_x86-64 contrib main non-free \
>> /etc/apt/sources.list
```

### **!** Предупреждение

Файл репозитория `/etc/apt/sources.list.d/zodiac.list` не предназначен для модификации администратором, т.к. он регулярно приводится к состоянию, которое определяется сервером.

Если на части УК нужна какая-то отличающаяся конфигурация, следует создать отдельный файл репозитория или вносить изменения в стандартный файл (который мы не модифицируем).

## 3.4.5 Элемент директивы «Конфигурация»

Элемент директивы предназначен для переопределения параметров конфигурации агентов (а также ТО), заданных по умолчанию, а также в файлах конфигурации на управляемых компьютерах.

Таким образом обеспечивается **динамическая** конфигурация агентов, в отличие от **статической** конфигурации, которую, в свою очередь, можно разделить на **внешнюю** конфигурацию – файл `agent.ini` в каталоге данных агента, и **внутреннюю** конфигурацию – файл `agent.ini` в каталоге установки агента (расположения указаны в табл. 2).

Таблица 2 Расположение файлов статической конфигурации агентов и ТО на УК

Конфигурация	ОС	Папка расположения <b>agent.ini</b>
Внешняя	Windows	%ALLUSERSPROFILE%\Zodiac\agent
	Linux	/var/zodiac/agent
Внутренняя	Windows	%ProgramFiles%\Zodiac\Agent
	Linux	/opt/zodiac/agent/

### Предупреждение

Динамически можно изменить **не все** параметры, которые могут быть заданы в файлах конфигурации или через параметры установки.

Неосмотрительное переопределение некоторых параметров на большом количестве управляемых компьютеров может вызывать неконсистентное поведение системы. Поэтому элемент директивы «Конфигурация» предоставляет изменение только тех параметров, которые могут быть **безопасно применены**, в том числе **без перезагрузки агентов**.

При крайней необходимости вы всегда можете изменить **внешнюю** конфигурацию соответствующим скриптом. Заданные во внешней конфигурации параметры имеют **приоритет** над заданными во внутренней.

Перечень параметров можно посмотреть в файле **внутренней** конфигурации, которая изначально заполнена разработчиком параметрами по умолчанию. Внутренняя конфигурация не предназначена для ручного изменения, т.к. **будет перезаписана при обновлении агента**.

Пример настройки динамической конфигурации показан на рисунке 3.32.

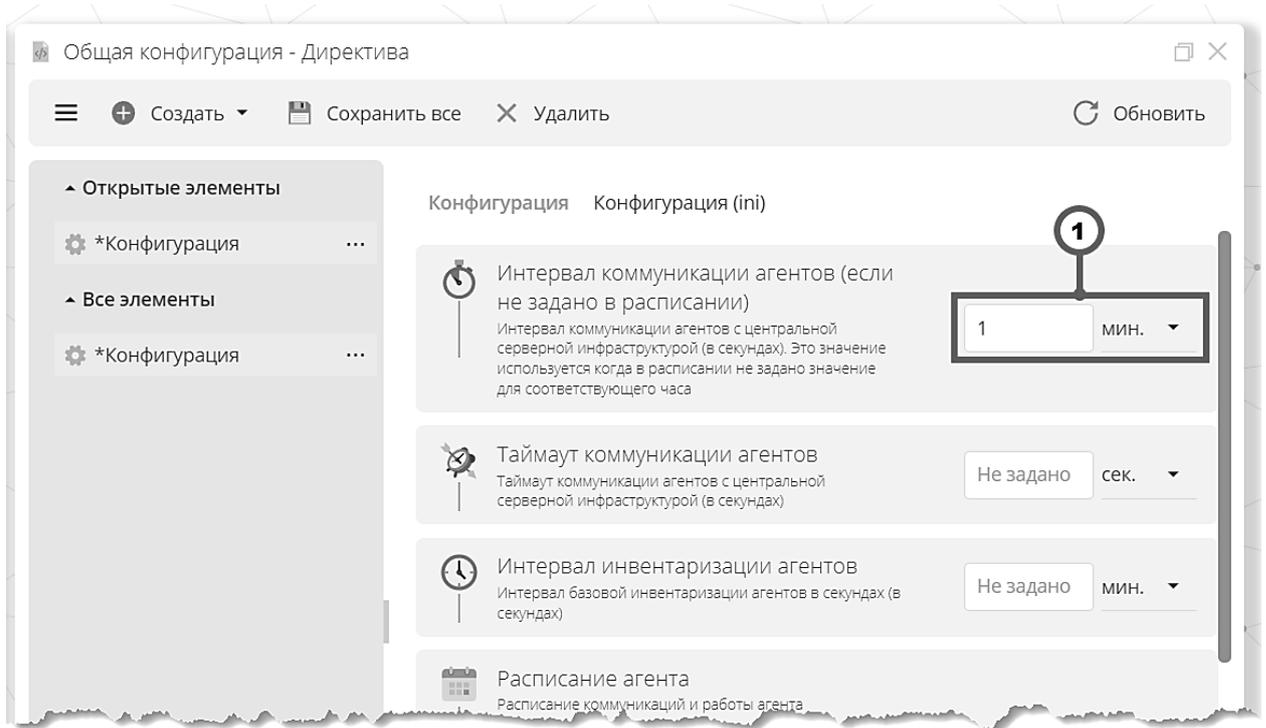


Рисунок 3.32 Настройка параметров динамической конфигурации

1. Пример задания значения **1 мин** для **интервала коммуникации агентов** с центральной серверной инфраструктурой.

#### Совет

Добавление директивы с элементом «Конфигурация» в **глобальную связку** (раздел 3.2.4) позволяет применить изменение параметров ко всем имеющимся и вновь регистрируемым УК.

#### 3.4.6 Элемент директивы «Развертывание»

Для создания элемента директивы откройте окно директивы и на панели инструментов вызовите окно элемента директивы развертывания как показано на рисунке 3.33.



Рисунок 3.33 Создание элемента директивы «Развёртывание»

1. Кнопка **«Создать»** – предназначена для вызова выпадающего списка выбора типа элемента директивы.
2. Кнопка **«Развёртывание»** – предназначена для вызова окна элемента директивы типа «Развёртывание».

На вкладке **«Состав»** выберите **пакеты развёртывания**, которые будут устанавливаться в рамках данного элемента директивы как показано на рисунке 3.34.

Создание и настройка **пакетов развёртывания** описано в разделе 3.16.

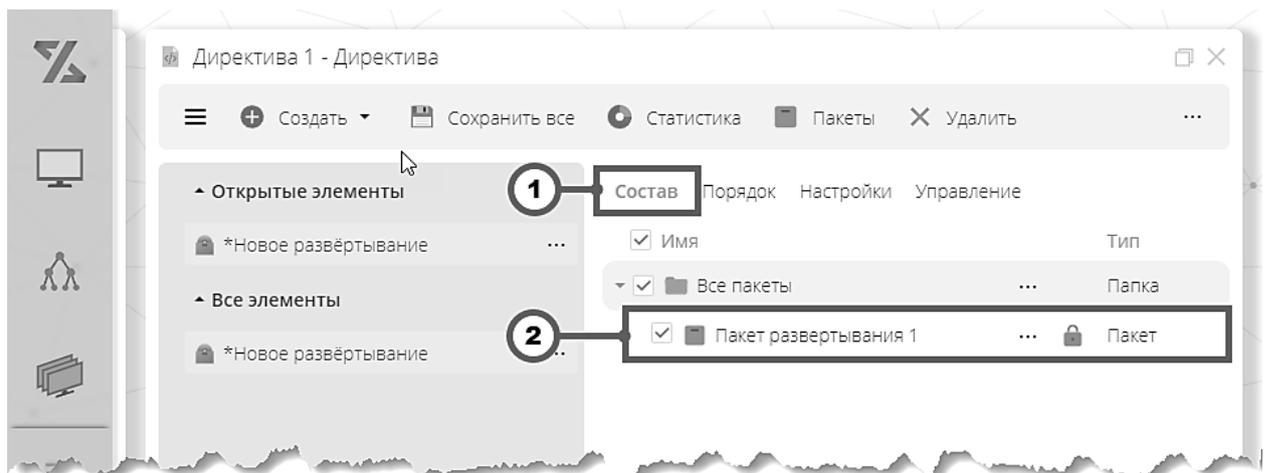


Рисунок 3.34 Настройка состава директивы развёртывания

1. Вкладка **«Состав»** – вызывает панель со списком всех пакетов развёртывания.
2. **Выбранный** пакет развёртывания.

Если элемент директивы предполагает больше одного пакета развертывания и, при этом, важен порядок, в котором они будут устанавливаться, на вкладке **«Порядок»** настройте очередность пакетов (рисунок 3.35).

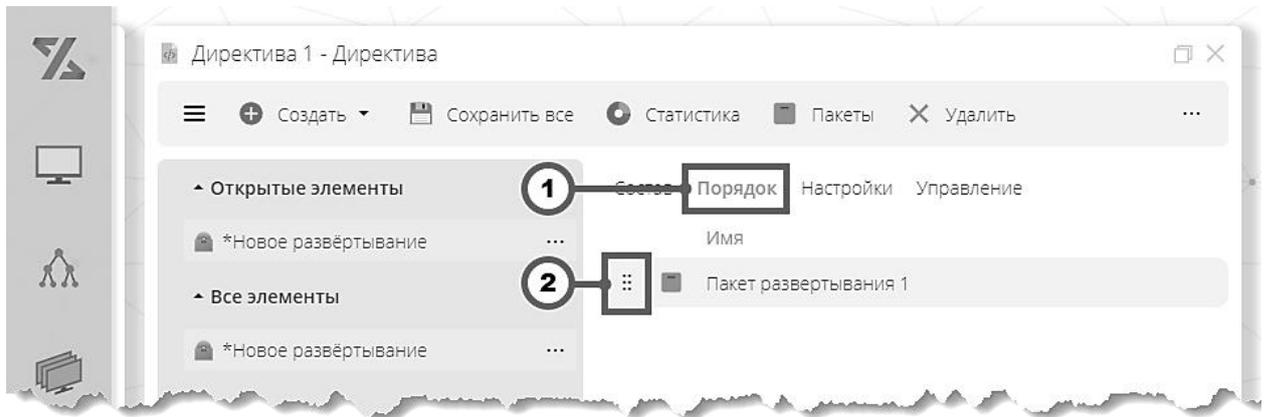


Рисунок 3.35 Настройка порядка развертывания пакетов

1. Вкладка **«Порядок»** – вызывает панель с элементами управления очередностью пакетов развертывания.
  2. «Рукоятка» для перетаскивания элементов списка для изменения их **очередности**.
- Задайте настройки директивы развертывания (рис. 3.36, 3.37).

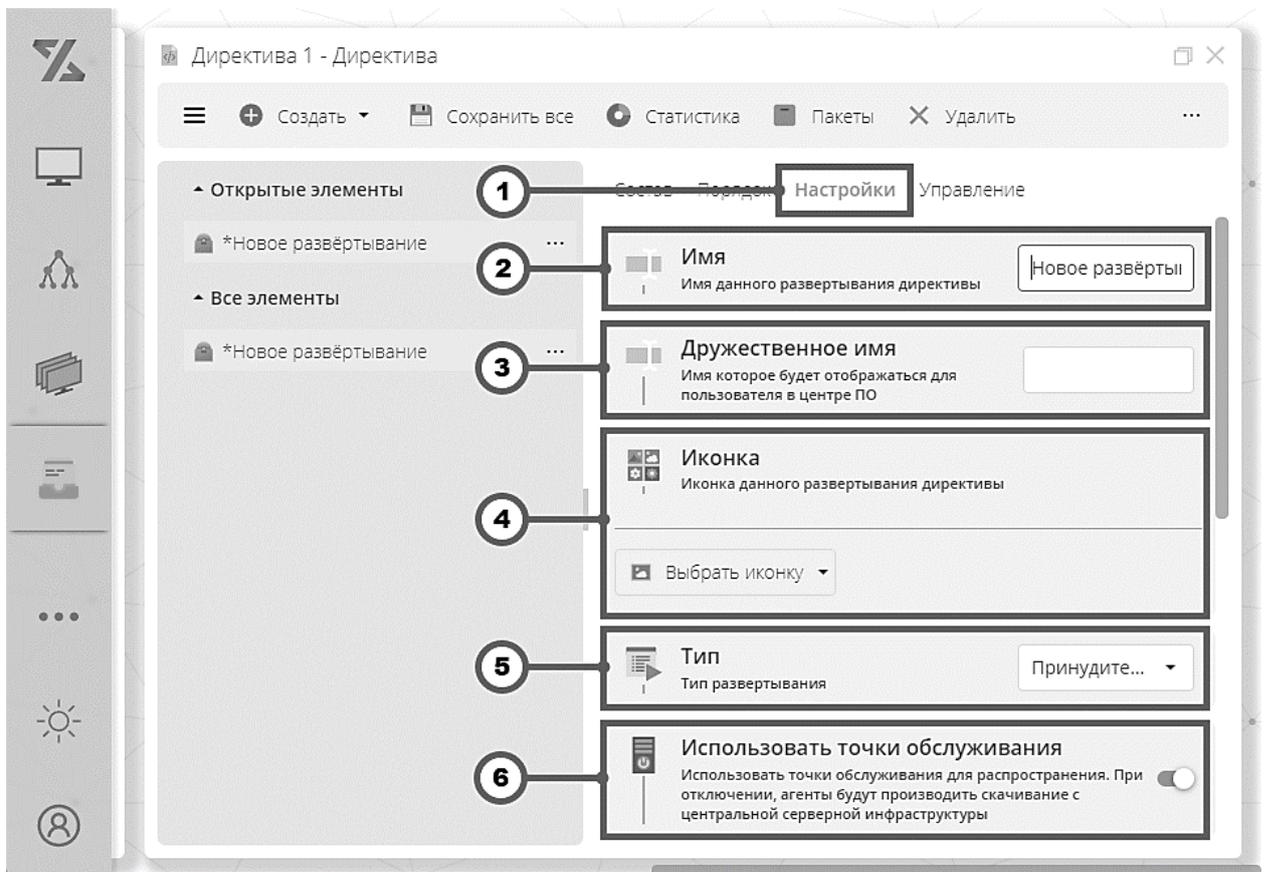


Рисунок 3.36 Настройки директивы развертывания

1. Вкладка **«Настройки»** – вызывает панель с формой для заполнения настроек элемента директивы.
2. Поле ввода **«Имя»** – предназначено для ввода имени элемента директивы.
3. Поле ввода **«Дружественное имя»** – предназначено для ввода имени, которое будет отображаться для пользователя целевого компьютера в **магазине приложений** (раздел 5.1.2).
4. Кнопка **«Выбрать иконку»** – предназначена для выбора файла изображения, или файла, содержащего ресурсы с иконками. В случае выбора файла с ресурсами, содержащими множество иконок, предоставляется возможность выбора подходящей иконки. Выбранная иконка будет отображаться для пользователя целевого компьютера в **магазине приложений** (раздел 5.1.2).
5. Список выбора **«Тип развертывания»** – предназначен для выбора типа развертывания. Для выбора доступно три типа: **«Принудительно (скрыто)»**, **«Принудительно (открыто)»**, **«По требованию»**.
6. Переключатель **«Использовать точки обслуживания»** – предназначен для включения/выключения режима, при котором агенты могут скачивать пакеты развертывания с **точек обслуживания**. При выключении режима агенты будут скачивать пакеты с **центральной серверной инфраструктуры**.

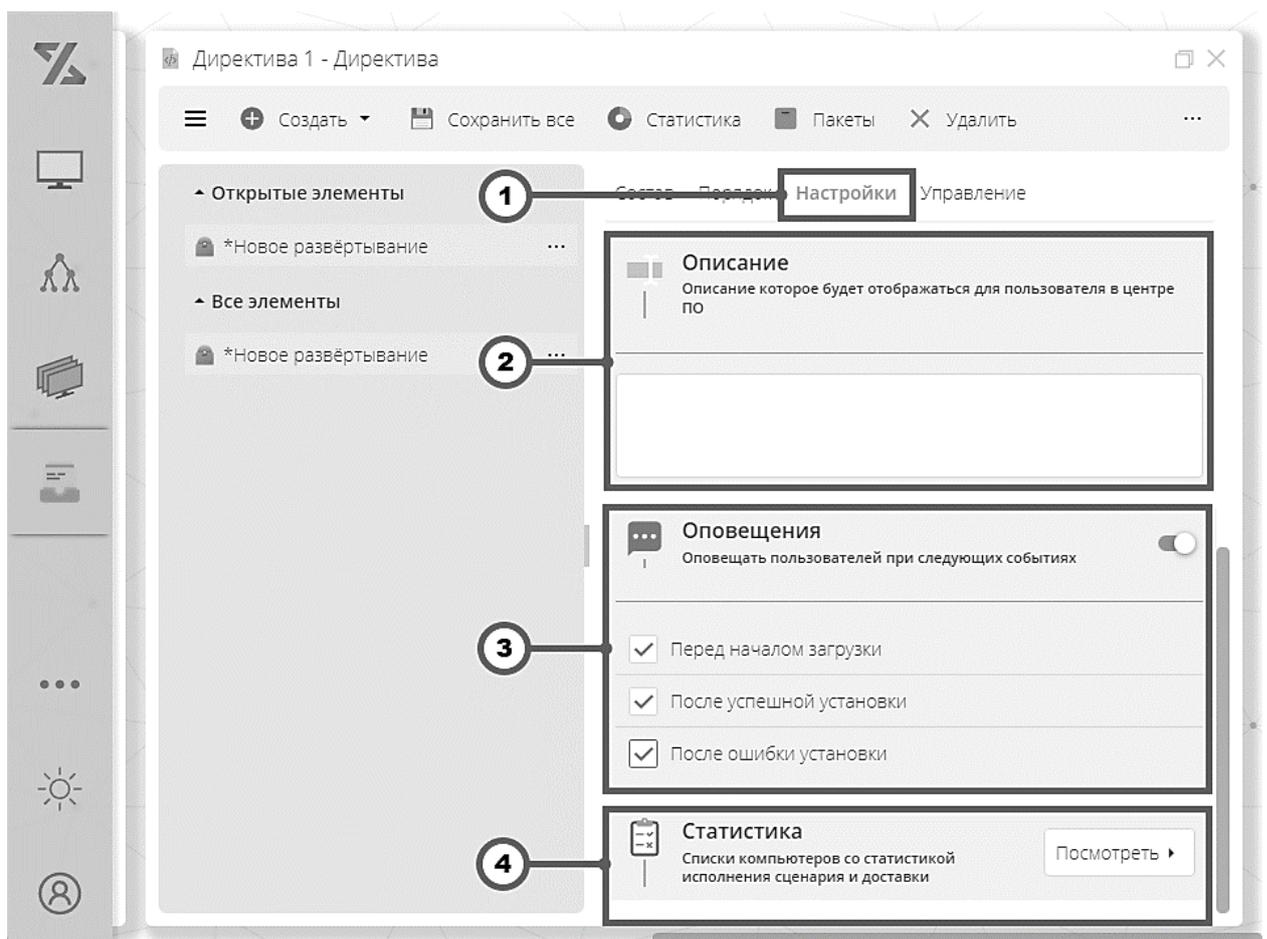


Рисунок 3.37 Настройки директивы развертывания (продолжение)

1. Вкладка **«Настройки»** – вызывает панель с формой для заполнения настроек элемента директивы.
2. Поле ввода **«Описание»** – предназначено для ввода описания программного обеспечения, которое будет отображаться пользователю целевого компьютера в магазине приложений.
3. Группа переключателей **«Оповещения»** – предназначена для включения или выключения оповещения пользователя целевого компьютера при следующих событиях: **«Перед началом загрузки»**, **«После успешной установки»**, **«После ошибки установки»**.
4. Кнопка **«Статистика»** – предназначена для перехода на логическое окно просмотра статистики исполнения сценария и доставки на целевых компьютерах.

**Включите** доставку и установку пакетов развертывания на целевые компьютеры на вкладке **«Управление»**, используя **режим ручного управления** или **управления по расписанию**. При использовании точек обслуживания настройте также доставку на точки обслуживания.

В примере на рисунке 3.38 доставка от центральных серверов до точек обслуживания включена, а от точек обслуживания до целевых компьютеров – выключена.

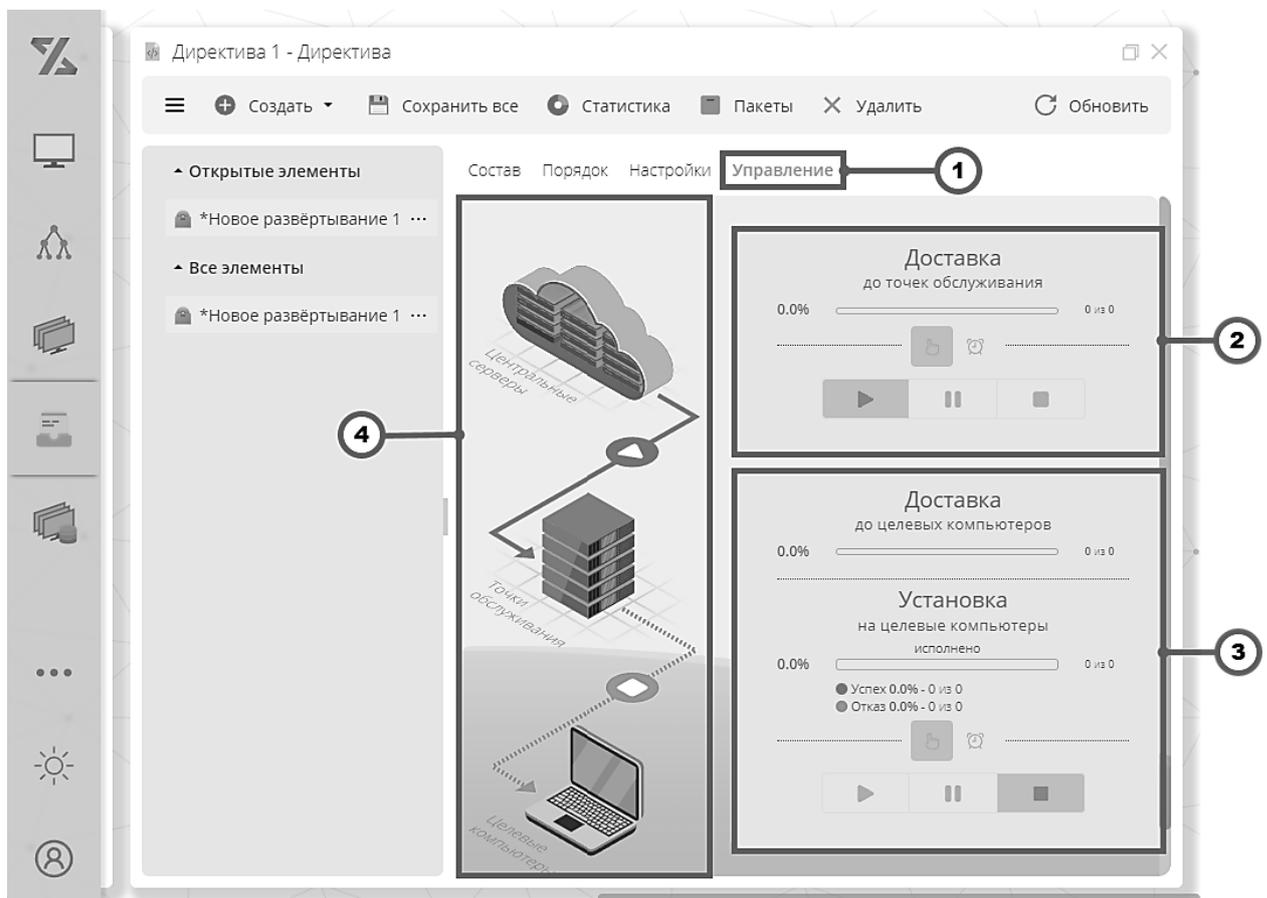


Рисунок 3.38 Управление доставкой пакетов развертывания

1. Вкладка **«Управление»** – вызывает панель с элементами управления доставкой и установкой пакетов развертывания.

2. Панель управления режима работы с **точками обслуживания** – предназначена для задания режима управления доставкой до точек обслуживания. В режиме **«Ручное управление»** доставка управляется кнопками **«Старт»**, **«Пауза»**, **«Стоп»**. В режиме **«Управление по расписанию»** указываются моменты времени начала, приостановки и остановки доставки. Панель содержит также суммарную статистику по доставке.
3. Панель управления режима работы с **целевыми компьютерами** – предназначена для задания режима управления доставкой до целевых компьютеров и установкой на целевые компьютеры. В режиме **«Ручное управление»** доставка и установка управляется кнопками **«Старт»**, **«Пауза»**, **«Стоп»**. В режиме **«Управление по расписанию»** указываются моменты времени начала, приостановки и остановки доставки. Панель содержит также суммарную статистику по доставке и установке.
4. Мнемоническая схема, отображающая **состояние процесса** доставки и установки пакетов развертывания, заданное элементом директивы. В нашем примере доставка от центральных серверов до точек обслуживания **включена**, а от точек обслуживания до целевых компьютеров – **выключена**.

---

 **Примечание**

Панель управления режима работы с точками обслуживания и точки обслуживания на мнемонической схеме отображаются только если в настройках директивы включена настройка **«Использовать точки обслуживания»**.

---

### 3.4.7 Элемент директивы «Сканирование сети»

Результаты работы элемента директивы **«Сканирование сети»** автоматически формируют список **соседних устройств**, описанный в разделе 3.15.

Для создания элемента директивы «Сканирование сети» откройте окно директивы и на панели инструментов вызовите окно элемента директивы **«Сканирование сети»**, как показано на рисунке 3.39.

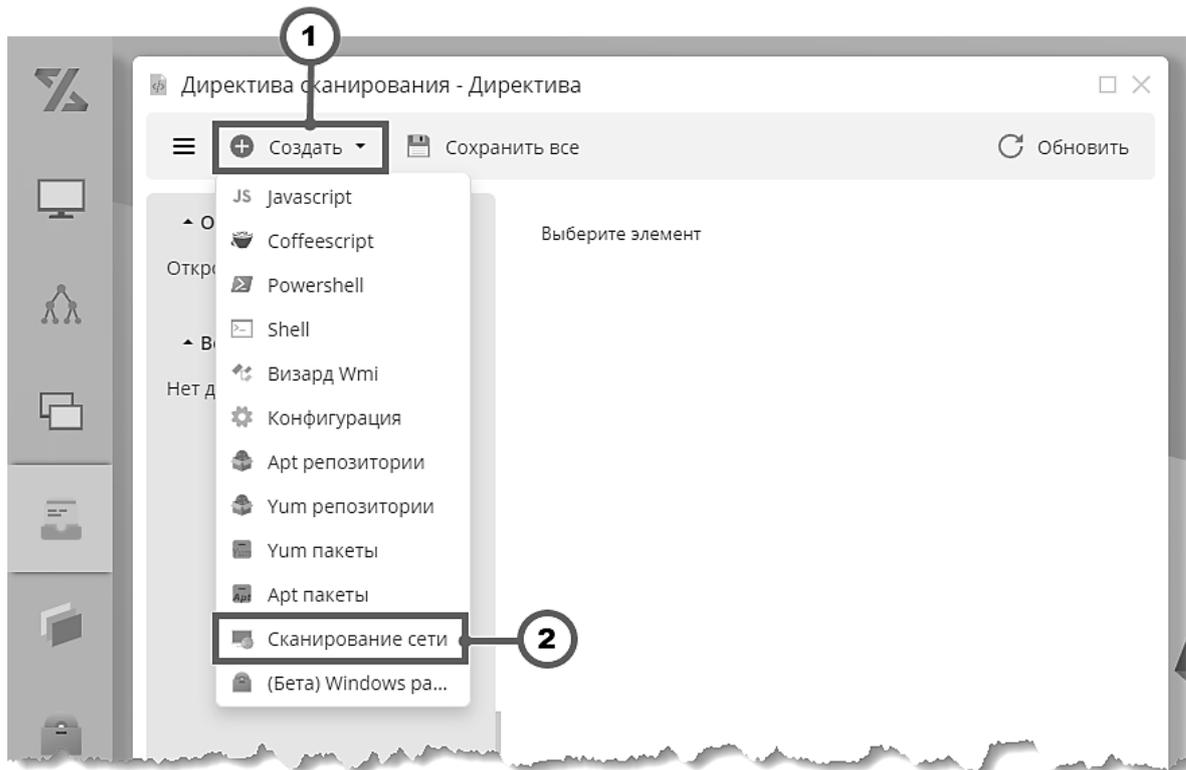


Рисунок 3.39 Создание элемента директивы «Сканирование сети»

1. Кнопка **«Создать»** – предназначена для вызова выпадающего списка выбора типа элемента директивы.
2. Кнопка **«Сканирование сети»** – предназначена для вызова окна элемента директивы типа «Сканирование сети».

В окне элемента директивы введите имя элемента, настройте расписание, как показано на рисунке 3.40, и нажмите **«Сохранить все»**.

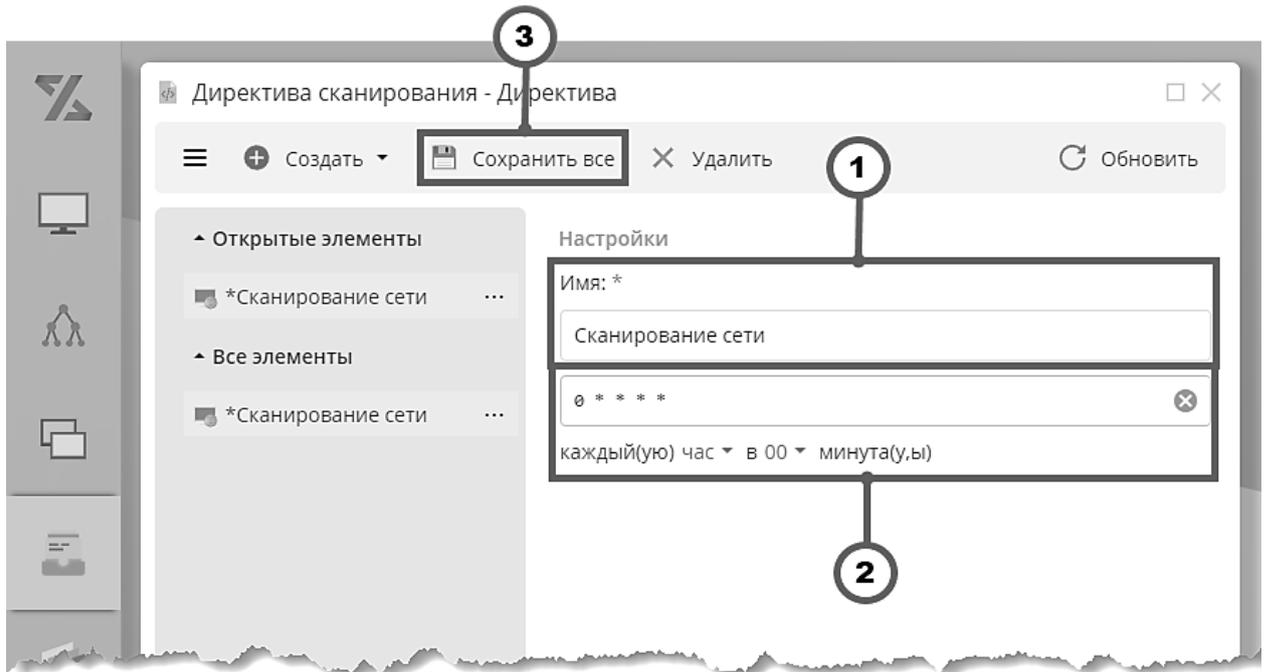


Рисунок 3.40 Настройка элемента директивы «Сканирование сети»

1. Поле ввода **«Имя»** – предназначено для ввода наименования элемента директивы. В нашем примере введено имя «Сканирование сети».
2. Панель **настройки расписания** – предназначена для настройки моментов запуска сканирования. Подробно о работе с панелью в разделе 2.2.5.
3. Кнопка **«Сохранить все»** – сохраняет все внесенные в окне директивы изменения.

### 3.4.8 Элемент директивы «Инвентаризация WMI»

Для создания элемента директивы «Инвентаризация WMI» откройте окно директивы и на панели инструментов вызовите окно элемента директивы «Инвентаризация WMI», как показано на рисунке 3.41

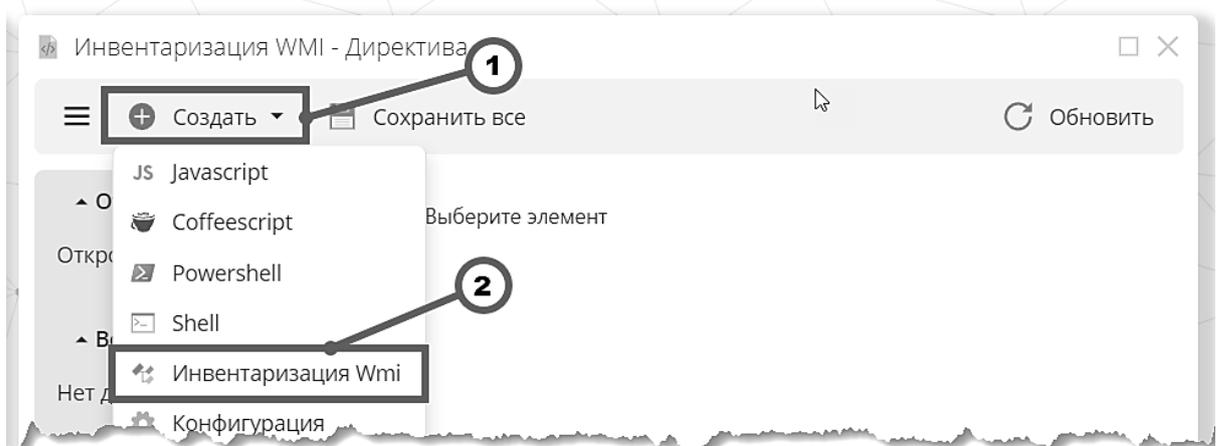


Рисунок 3.41 Создание элемента директивы «Инвентаризация WMI»

1. Кнопка **«Создать»** – предназначена для вызова выпадающего списка выбора типа элемента директивы.
2. Кнопка **«Инвентаризация WMI»** – предназначена для вызова окна элемента директивы типа **«Инвентаризация WMI»**.

На панели **«Класс»** выберите класс WMI как показано на рисунке 3.42.

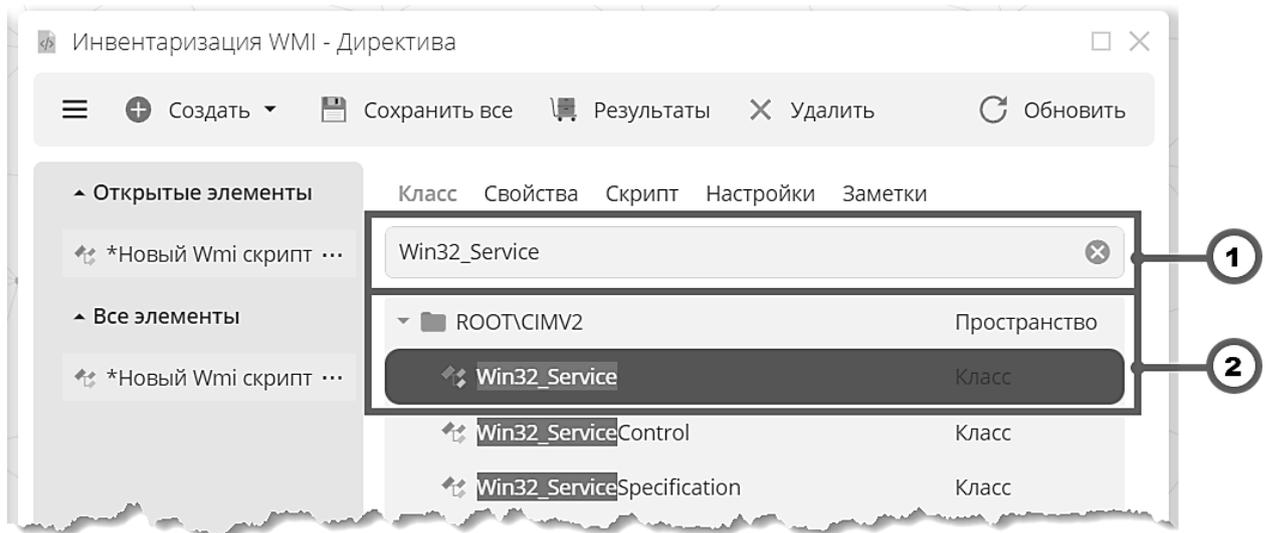


Рисунок 3.42 Выбор класса WMI

1. **Строка поиска** – предназначена для поиска в дереве классов WMI. В нашем примере осуществлялся поиск класса **Win32\_Service**.
2. **Дерево классов WMI** – предназначено для выбора класса WMI, свойства которого могут быть включены в инвентаризацию

Далее, на панели **«Свойства»** выберите свойства класса WMI, выбранного на предыдущем шаге, как показано на рисунке 3.43.

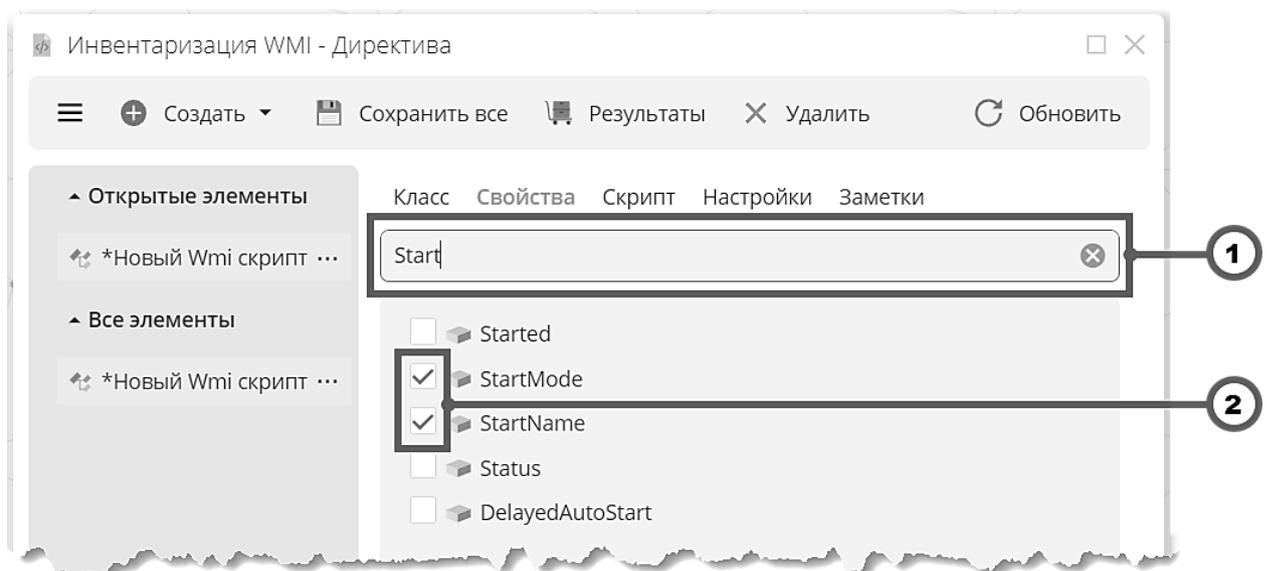


Рисунок 3.43 Выбор свойств класса WMI

1. **Строка поиска** – предназначена для поиска в списке свойств класса WMI
2. **Список свойств** класса WMI – предназначен для выбора свойств класса WMI, включаемых в инвентаризацию. В нашем примере выбраны свойства **StartMode** и **StartName**.

На панели **«Скрипт»** (рисунок 3.44) можно просмотреть скрипт, построенный на основании выбора на предыдущих шагах. Скрипт строится автоматически и **недоступен для редактирования**.

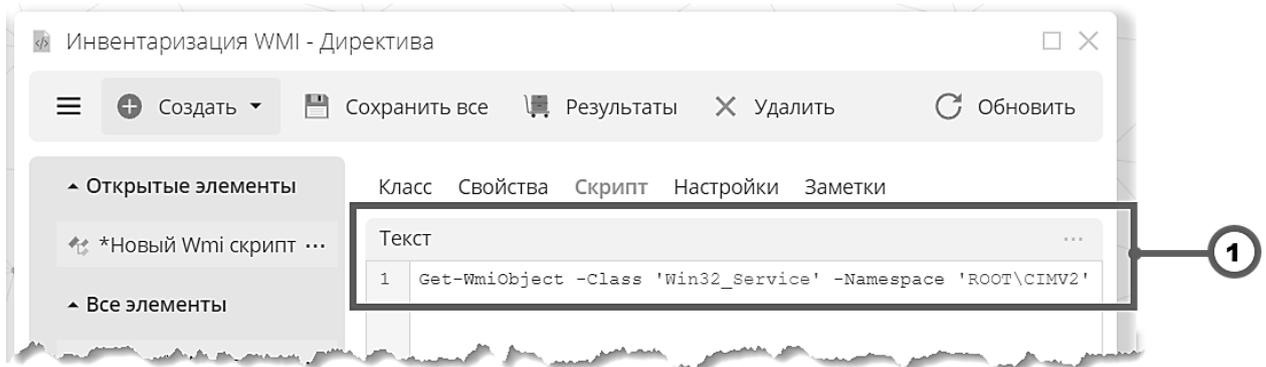


Рисунок 3.44 Текст скрипта инвентаризации WMI

1. Панель просмотра **текста скрипта** инвентаризации WMI, построенного на результате выбора свойств класса WMI.

На панели **«Настройки»** задайте настройки аналогично элементу директивы типа скрипт – за исключением секции **«Результат»**, который **недоступен для редактирования**, поскольку строится автоматически на основе выбранных свойств класса WMI.

Для сохранения действий по созданию элемента директивы нажмите **«Сохранить все»**.

### 3.4.9 Элемент директивы «Инвентаризация пользователей»

Для создания элемента директивы откройте окно директивы и на панели инструментов нажатием кнопки **«Создать»** откройте окно элемента директивы инвентаризации пользователей. Заполните форму настройки инвентаризации, как показано на рисунке 3.45, и нажмите кнопку **«Сохранить все»** на панели инструментов окна директивы.

По результатам работы элемента директивы автоматически формируется каталог пользователей, описанный в разделе 3.7.1.

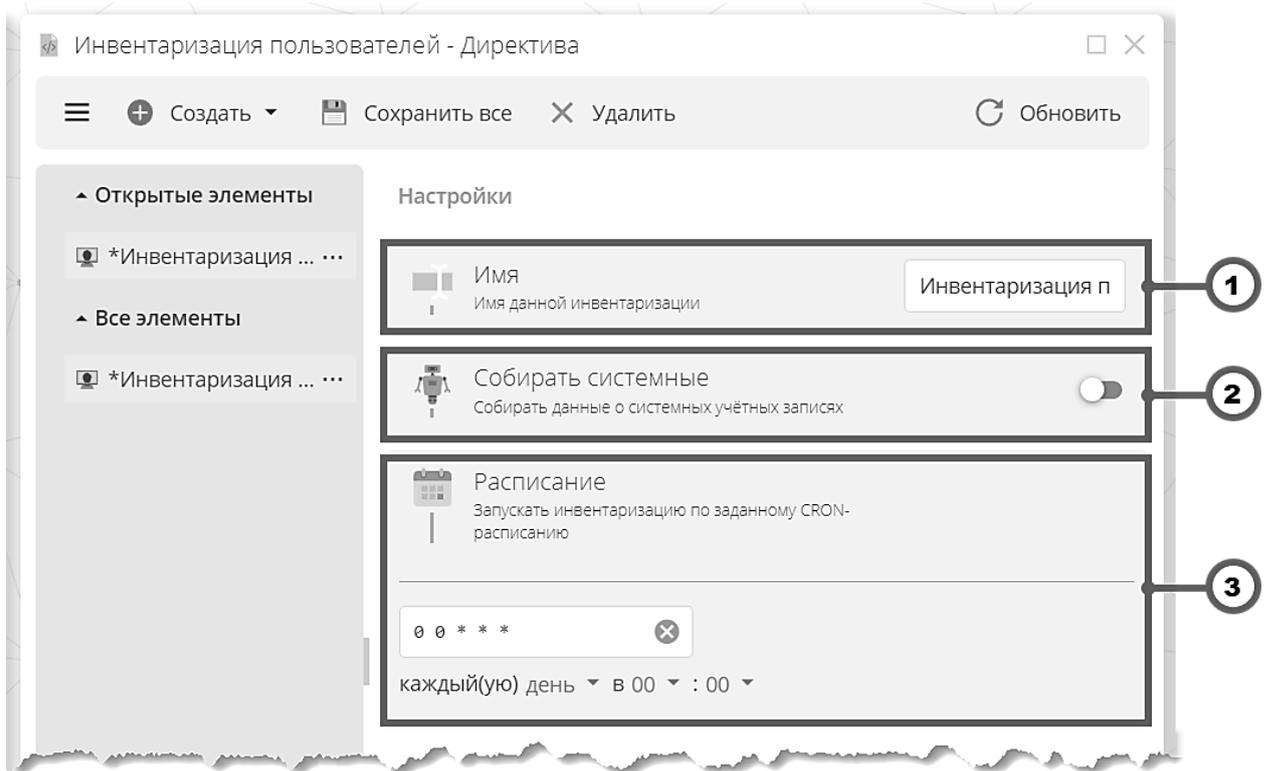


Рисунок 3.45 Создание элемента директивы «Инвентаризация пользователей»

1. Поле ввода **«Имя»** – предназначено для ввода наименования элемента директивы. В нашем примере введено имя «Инвентаризация пользователей».
2. Признак **«Собирать системные»** – предназначено для расширения категорий учетных записей собираемых задач по инвентаризации.
3. Панель **настройки расписания** – предназначена для настройки моментов запуска инвентаризации.

#### 3.4.10 Элемент директивы «Инвентаризация ПО»

Для создания элемента директивы откройте окно директивы и на панели инструментов нажатием кнопки **«Создать»** откройте окно элемента директивы инвентаризации ПО. Заполните форму настройки инвентаризации, как показано на рисунке 3.47, и нажмите кнопку **«Сохранить все»** на панели инструментов окна директивы.

По результатам работы элемента директивы автоматически формируется каталог ПО, описанный в разделе 3.17, а также собранные данные инвентаризации становятся доступны для использования в режиме «Отчеты ПО», описанном в разделе 3.18.

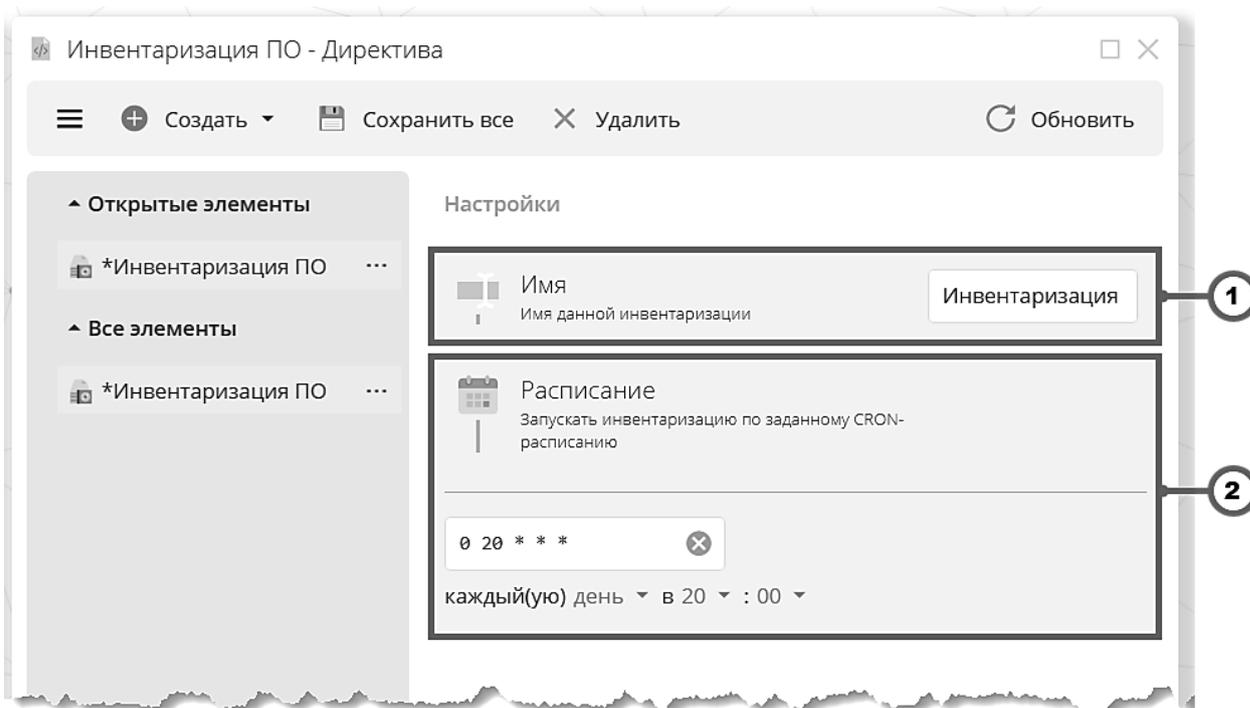


Рисунок 3.46 Создание элемента директивы «Инвентаризация ПО»

1. Поле ввода **«Имя»** – предназначено для ввода наименования элемента директивы. В нашем примере введено имя «Инвентаризация ПО».
2. Панель **настройки расписания** – предназначена для настройки моментов запуска инвентаризации.

#### 3.4.11 Просмотр статистики работы скрипта

Откройте окно директивы, содержащей скрипт, статистику работы которого нужно просмотреть, например, с помощью контекстного меню, как показано на рисунке 3.47.

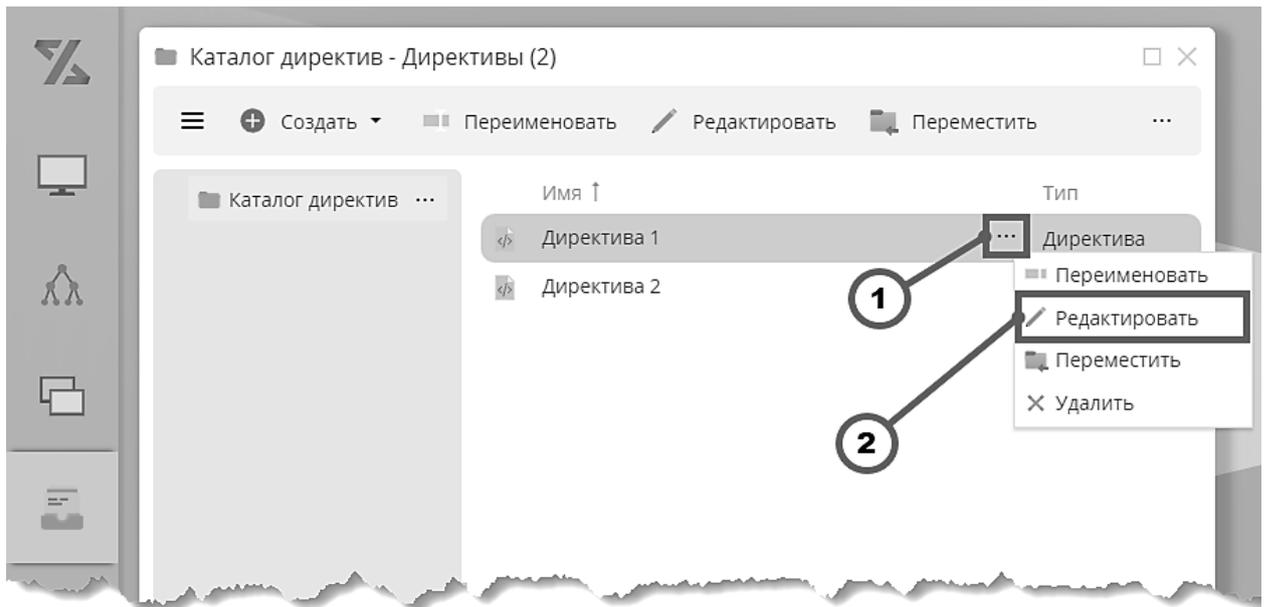


Рисунок 3.47 Вызов окна директивы

1. Кнопка «...» – предназначена для вызова контекстного меню списка директив.
2. Пункт контекстного меню **«Редактировать»** – предназначен для вызова логического окна существующей директивы.

Выберите скрипт, статистику работы которого нужно просмотреть, в списке элементов директивы, как показано на рисунке 3.48, и нажмите в панели инструментов окна директивы кнопку **«Статистика»**.

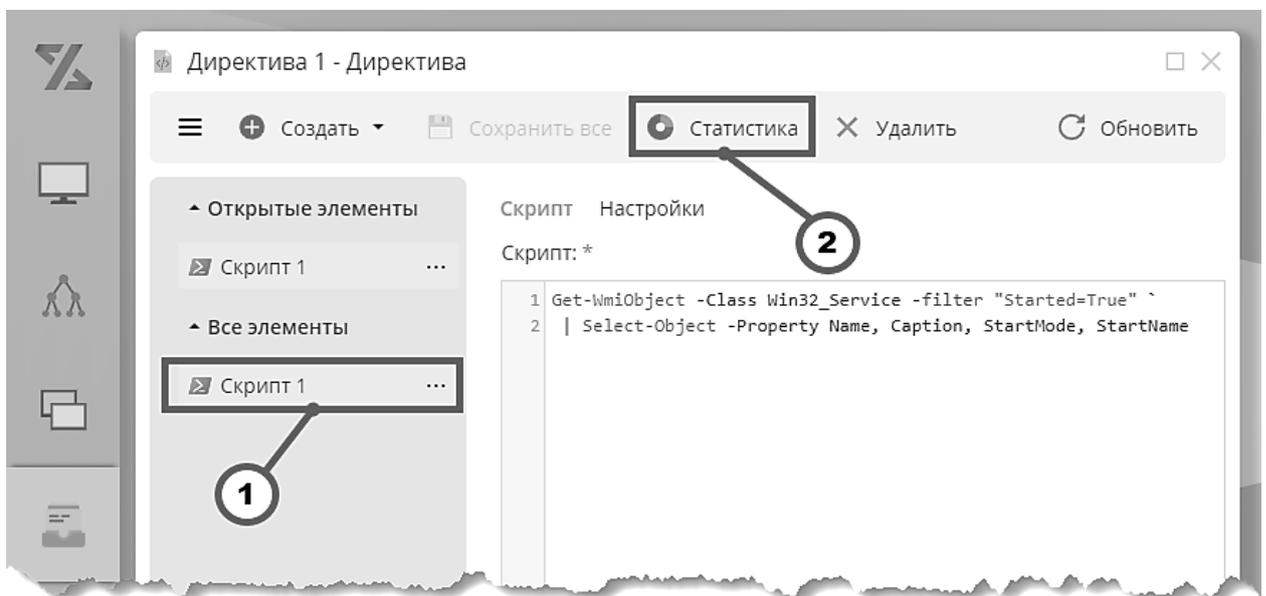


Рисунок 3.48 Вызов окна статистики скрипта

1. **Пункт списка элементов директивы** – предназначен для открытия выбранного элемента директивы (в нашем примере скрипта **«Скрипт 1»**).

2. Кнопка **«Статистика»** – предназначена для вызова логического окна статистики открытого скрипта.

Логическое окно статистики скрипта (рисунок 3.49) содержит таблицу со статистикой работы скрипта на назначенных компьютерах.

Компьютер	Усп. вып.	Неусп. вып.	Посл. усп. вып.	Посл. неусп. вып.
ZDC-TEST-W10	40	0	18.05.2022, 01:52	
ZDC-DEV-W10	47	0	18.05.2022, 01:59	
ZDC-SRV-W12	40	0	18.05.2022, 01:54	

Рисунок 3.49 Окно статистики работы скрипта

1. Столбец **«Компьютер»** – отображает имя компьютера, на который был назначен скрипт.
2. Столбец **«Успешных выполнений»** – отображает количество успешных выполнений скрипта с момента назначения скрипта на соответствующий компьютер.
3. Столбец **«Неуспешных выполнений»** – отображает количество неуспешных выполнений скрипта с момента назначения скрипта на соответствующий компьютер.
4. Столбец **«Последнее успешное выполнение»** – отображает дату и время последнего успешного выполнения скрипта.
5. Столбец **«Последнее неуспешное выполнение»** – отображает дату и время последнего неуспешного выполнения скрипта.

### 3.4.12 Просмотр результатов работы скрипта

Откройте окно директивы, содержащей скрипт, результаты работы которого нужно просмотреть.

Далее, выберите скрипт, результаты работы которого нужно просмотреть, в списке элементов директивы, и нажмите в панели инструментов окна директивы кнопку **«Результаты»** (рисунок 3.50).

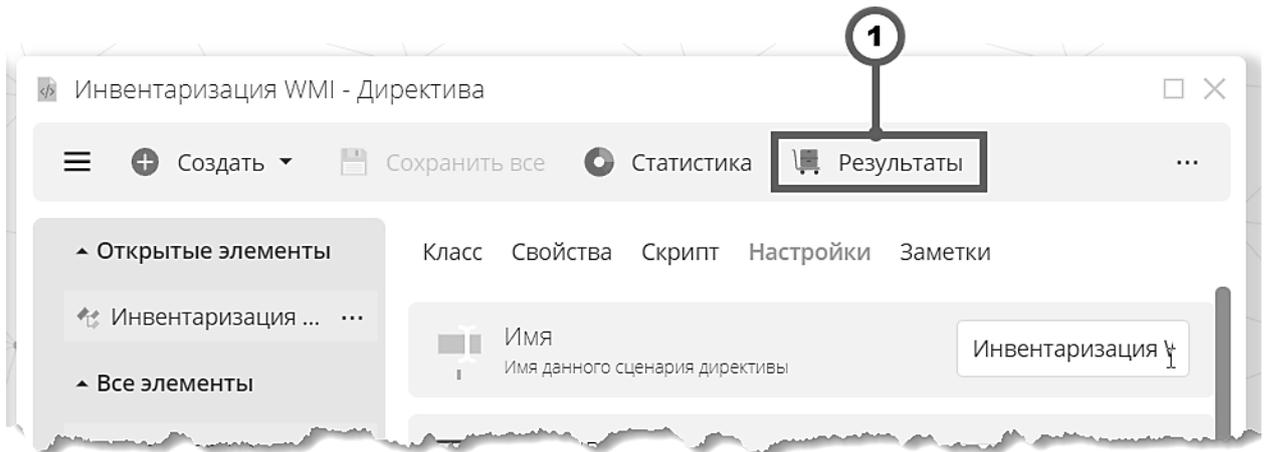


Рисунок 3.50 Вызов окна просмотра результатов работы скрипта

1. Кнопка **«Результаты»** – предназначена для вызова логического окна результатов работы выбранного скрипта.

Логическое окно результатов работы скрипта (рисунок 3.51) содержит таблицу с результатами работы скрипта на назначенных компьютерах.

hostname	le_time	Caption	Name	StartMode	StartName
ZDC-AG-WIN11	13.12.2023, 11:25	Служба шлюза уровня приложения	ALG	Manual	NT AUTHORITY\LocalService
ZDC-AG-WIN11	13.12.2023, 11:25	Удостоверение приложения	AppIDSvc	Manual	NT Authority\LocalService
ZDC-AG-WIN11	13.12.2023, 11:25	Сведения о приложении	Appinfo	Manual	LocalSystem
ZDC-AG-WIN11	13.12.2023, 11:25	Управление приложениями	AppMgmt	Manual	LocalSystem
ZDC-AG-WIN11	13.12.2023, 11:25	Готовность приложений	AppReadiness	Manual	LocalSystem

Рисунок 3.51 Окно просмотра результатов работы скрипта

1. Столбец **«hostname»** – отображает имя компьютера, на который был назначен скрипт.
2. Столбец **«le\_time»** – отображает время последнего исполнения скрипта.
3. Столбцы, соответствующие свойствам, указанным в секции **«Результат»** на панели настроек скрипта.

### 3.4.13 Директивы точки обслуживания

ТО представляет собой агент системы «Зодиак.АйТиЭм», работающий в режиме «Точка обслуживания», и применяется для оптимизации передачи контента большого объема, когда сопоставленные с определенной ТО агенты скачивают контент не с центрального сервера, а с этой ТО. Механизм сопоставления ТО и агентов описан в разделе 3.12.

Для установки ТО используется тот же инсталлятор, что и для агента. Режим точки обслуживания задается параметром **ServicePoint** в файле внешней конфигурации агента (раздел 3.4.5) или при установке:

```
[ServicePoint]
enabled=true
```

УК, на которых агенты работают в режиме ТО, можно увидеть в режиме «Компьютеры» в колонке «ПТО» – «Признак Точки Обслуживания» (рисунок 3.52).

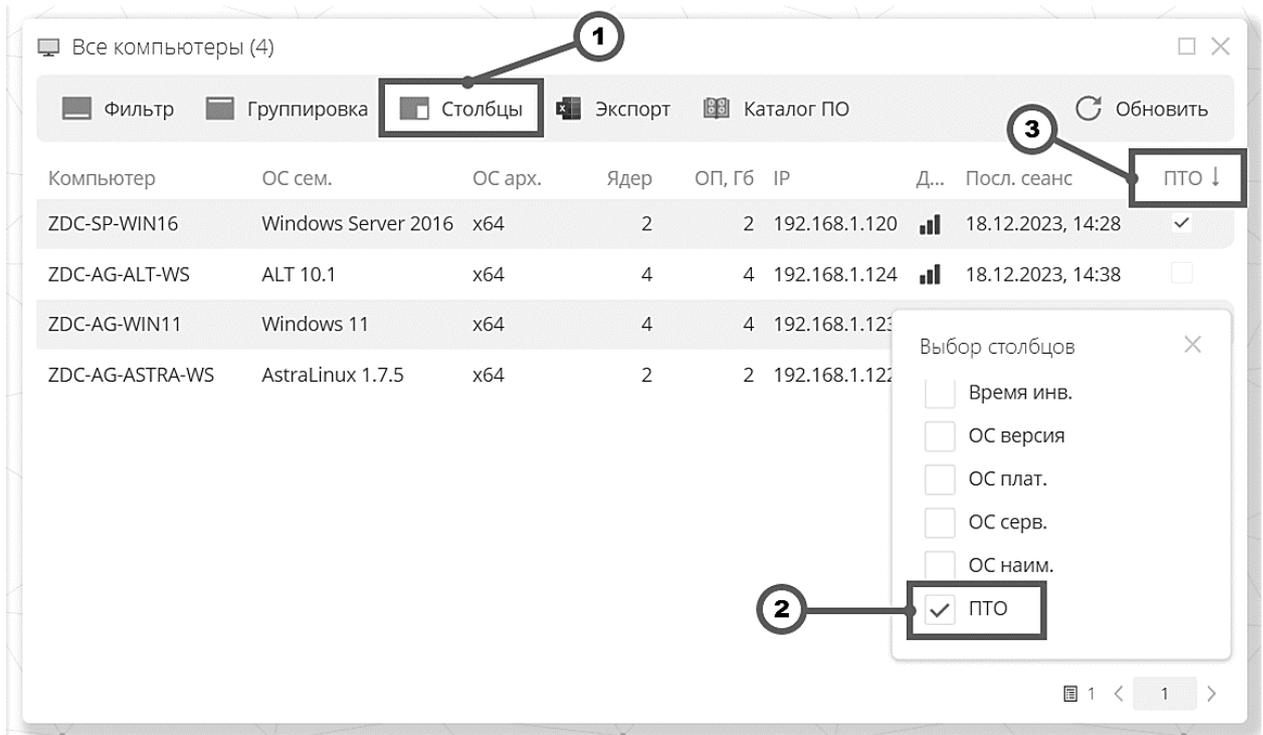


Рисунок 3.52 Признак ТО в списке компьютеров

1. Кнопка «**Столбцы**» – вызывает окно выбора столбцов.
2. Признак «**ПТО**» – включение отображения столбца «ПТО» в списке компьютеров.
3. Столбец «**ПТО**» – отображает признак работы агента в режиме ТО.

### ! Предупреждение

С одной стороны, используя параметр **ServicePoint** можно обеспечить миграцию между ролями агента, т.е. превращать агентов в ТО и обратно. Но более надежный и правильный подход – **запланировать** расстановку ТО в подсетях заранее, и задать режим ТО **при начальной установке**.

Это связано с тем, что **переключение режима** работы агента, а также **актуализация данных** для сопоставления агентов и ТО на центральном сервере может занять заметное время, в течение которого может наблюдаться неконсистентное поведение системы.

Для создания директивы ТО перейдите в каталог директив и на панели инструментов вызовите окно создания директивы ТО как показано на рисунке 3.53.



Рисунок 3.53 Вызов окна создания новой директивы ТО

1. Кнопка «**Создать**» – предназначена для вызова выпадающего списка выбора команды создания элемента каталога директив.
2. Кнопка «**Новая директива ТО**» – предназначена для вызова окна создания директивы ТО.

В окне создания директивы ТО введите имя директивы ТО и нажмите «ОК» (3.54).



Рисунок 3.54 Окно создания директивы ТО

1. Форма – предназначена для ввода полей новой директивы ТО.
2. Поле ввода – предназначено для ввода имени новой директивы ТО.

Дальнейшие действия аналогичны действиям по созданию элементов обычных директив агента.

### ! Предупреждение

Директивы ТО, ассоциированные с некоторой группой **не будут доставлены** на обычные УК в этой группе.

И обратно, если группа, в которую включена ТО, ассоциирована с обычными директивами, они **не будут доставлены** на данную ТО

## Примечание

Различение директивы ТО и обычной директивы агента обусловлено намерением избежать ошибок в распределении задач между ТО и обычными УК.

Случайное назначение задач, характерных обычно для ТО, и носящих массово-инфраструктурный характер (например, сканирование сети, открытие портов, создание общих папок, установка серверного ПО) на обычные рабочие станции может иметь последствия, требующие значительных усилий для их устранения.

И наоборот, случайное назначение задач конфигурирования пользовательских УК (например, установка АРМ-ов, закрытие портов, запрещение общих папок и т.п.) на ТО может привести к потере производительности или неработоспособности ТО.

Вследствие данного подхода к разделению директив, если вам необходимо назначить одинаковый сценарий на УК и ТО, нужно создать две директивы с одним и тем же сценарием – директиву агента и директиву ТО.

## 3.5 Выборки

### 3.5.1 Каталог выборок

Общий вид каталога выборок представлен ниже (рисунок 3.55).

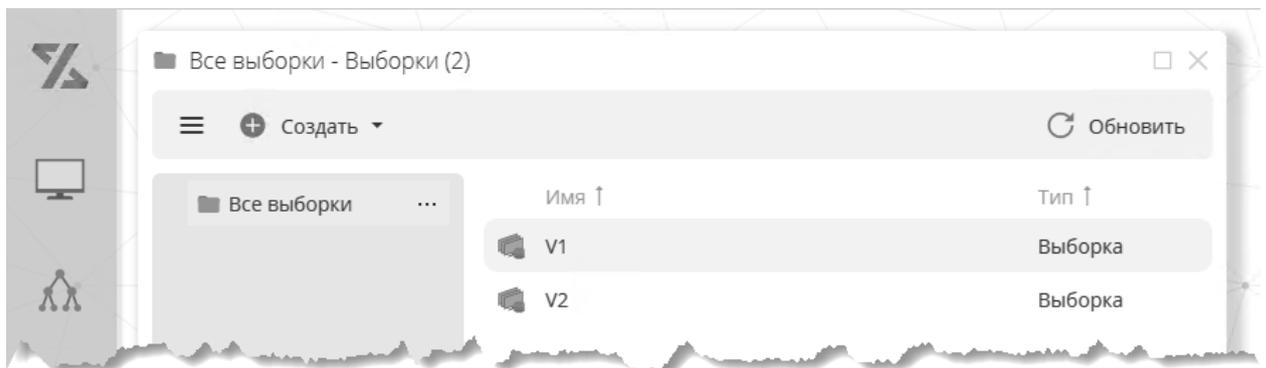


Рисунок 3.55 Каталог выборок

Для создания выборки перейдите в каталог выборок и на панели инструментов вызовите окно создания выборки (рисунок 3.56).

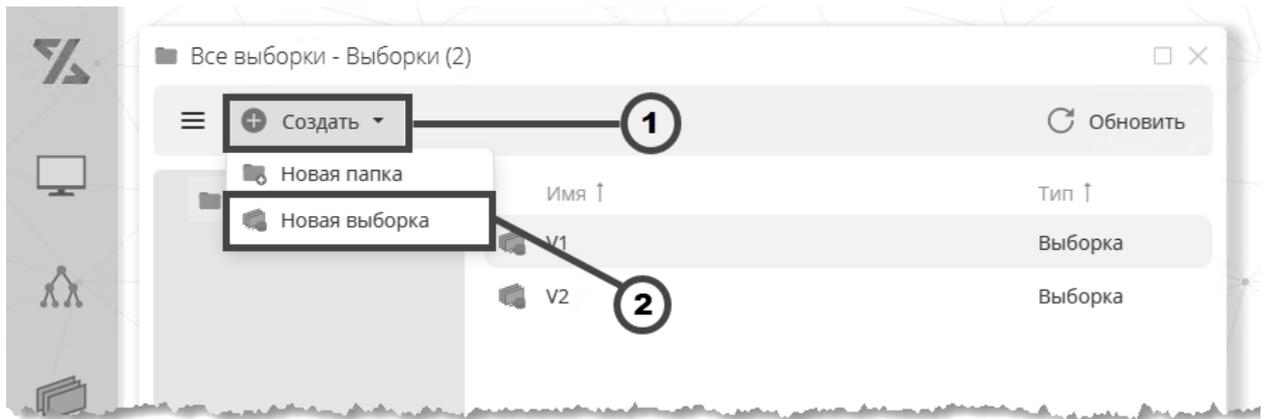


Рисунок 3.56 Вызов окна создания новой выборки

1. Кнопка **«Создать»** – предназначена для вызова выпадающего списка выбора команды создания элемента каталога выборок.
2. Кнопка **«Новая выборка»** – предназначена для вызова окна создания выборки.

В окне создания выборки введите имя новой выборки (в нашем примере – «Выборка 3») и нажмите «ОК» (рисунок 3.57).

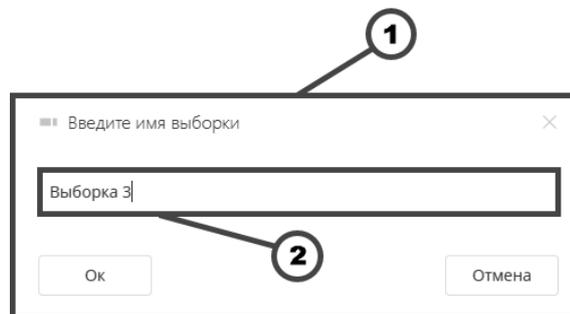


Рисунок 3.57 Окно создания выборки

1. Форма – предназначена для ввода полей новой выборки.
2. Поле ввода – предназначено для ввода имени новой выборки.

В результате предыдущего действия появится логическое окно новой выборки с деревом выбора свойств. Далее, следует произвести **настройку свойств** выборки, как описано в разделе 3.5.2.

### 3.5.2 Настройка свойств выборки

Окно настройки свойств выборки открывается сразу после действия по созданию выборки или после открытия существующей выборки на редактирование одним из способов (рисунок 3.58).

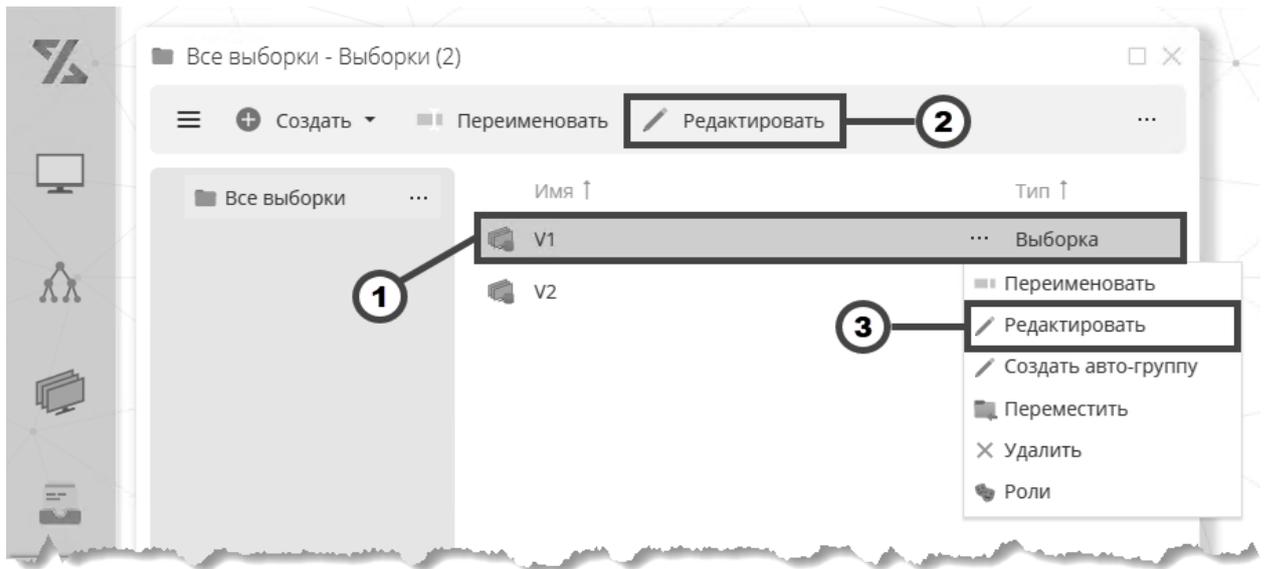


Рисунок 3.58 Вызов окна редактирования выборки

1. Выделение выборки, которую необходимо редактировать.
2. Пункт контекстного меню **«Редактировать»** - предназначен для вызова логического окна существующей выборки.
3. Пункт выпадающего списка **«Редактировать»**.
4. Кнопка **«Роли»** - предназначена для просмотра ролей, которым доступны операции с данной выборкой.

По умолчанию, окно настройки свойств выборки открывается в режиме **«Выбор свойств»**. В списке свойств (рисунок 3.59) выберите **свойства**, которые нужно включить в выборку. Далее, при необходимости отредактируйте отображаемые **имена и типы свойств** (рисунок 3.60), настройте **фильтрацию** (рисунок 3.61) и нажмите **«Сохранить»**.

### Совет

Функционал настройки выборки позволяет **объединять данные** базовой и настраиваемой инвентаризации. Для этого достаточно одновременно выбрать свойства из списков **«Таблицы компьютера»** и **«Каталог директив»** (рисунок 3.59).

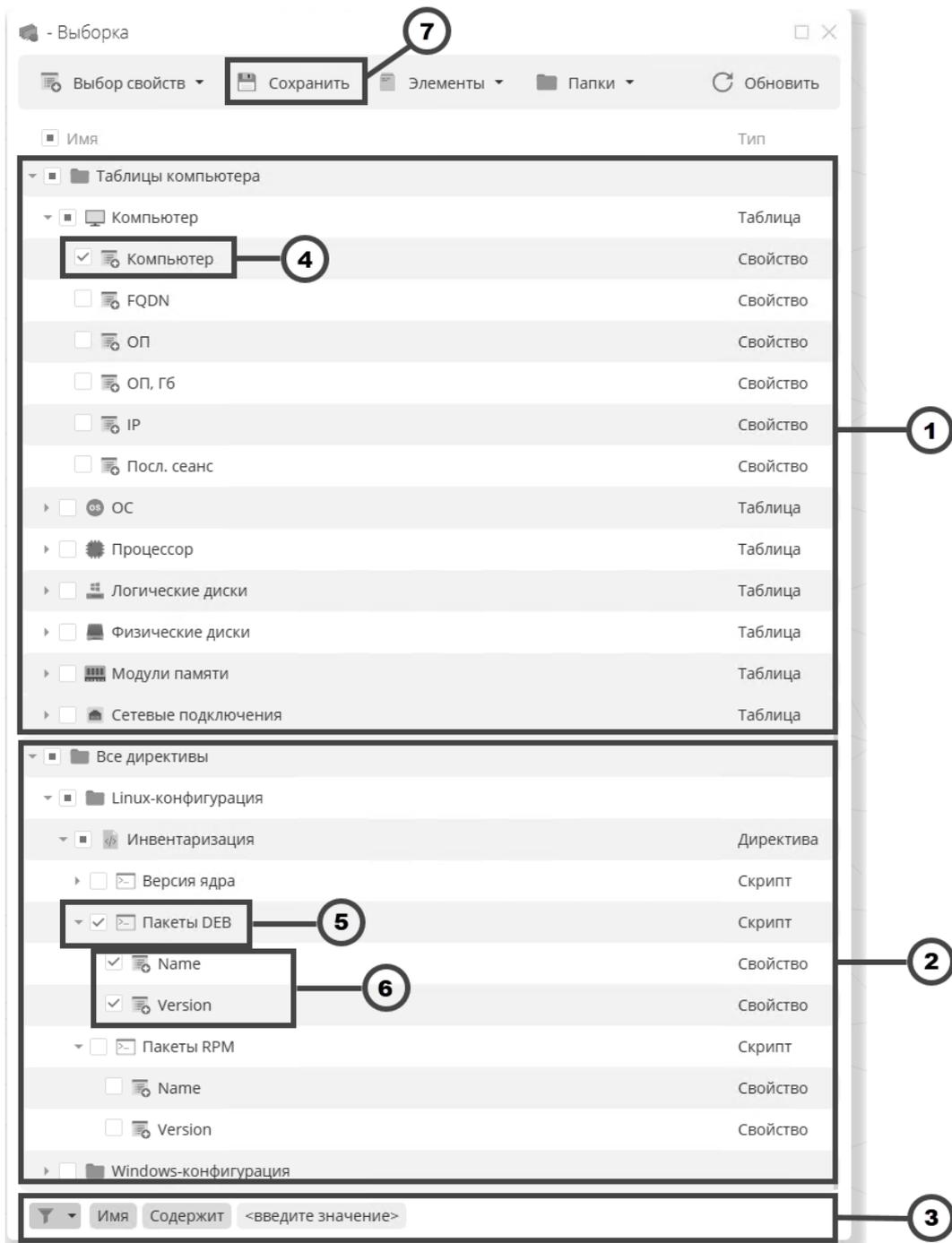


Рисунок 3.59 Выбор свойств в окне выборки

1. Древоподобный список выбора свойств «**Таблицы компьютера**» – предназначен для включения в выборку данных, собранных агентами в процессе базовой инвентаризации и объединения с данными настраиваемой инвентаризации.
2. Древоподобный список выбора свойств «**Все директивы**» – предназначен для включения в выборку данных, собранных агентами в результате настроенной с помощью директив инвентаризации и объединения с данными базовой инвентаризации.

3. **Фильтр** – предназначен для фильтрации списка свойств по произвольным критериям. Использование фильтра описано в разделе 2.2.4.
4. Свойство **«Компьютер»** (имя компьютера) – пример выбранного свойства из **таблицы «Компьютер»**, сформированной из данных базовой инвентаризации.
5. **Скрипт «Пакеты DEB»**, включенный в **директиву «Инвентаризация»** – пример выбранного скрипта, по результатам работы которого формируются данные настраиваемой инвентаризации для включения в выборку. Настройка скриптов, возвращающих результат, описана в разделе 3.4.2.
6. Свойства **«Name»** и **«Version»** – пример выбранных для включения в выборку свойств настраиваемой инвентаризации, данные которой формируются по результатам работы скрипта.
7. Кнопка **«Сохранить»** – предназначена для сохранения всех изменений в окне настройки выборки.

Общий вид окна редактирования выборки в режиме **«Настройки»** приведен на рисунке 3.60.

### **Примечание**

Выбранный тип свойства влияет на результаты сортировки в таблице данных.

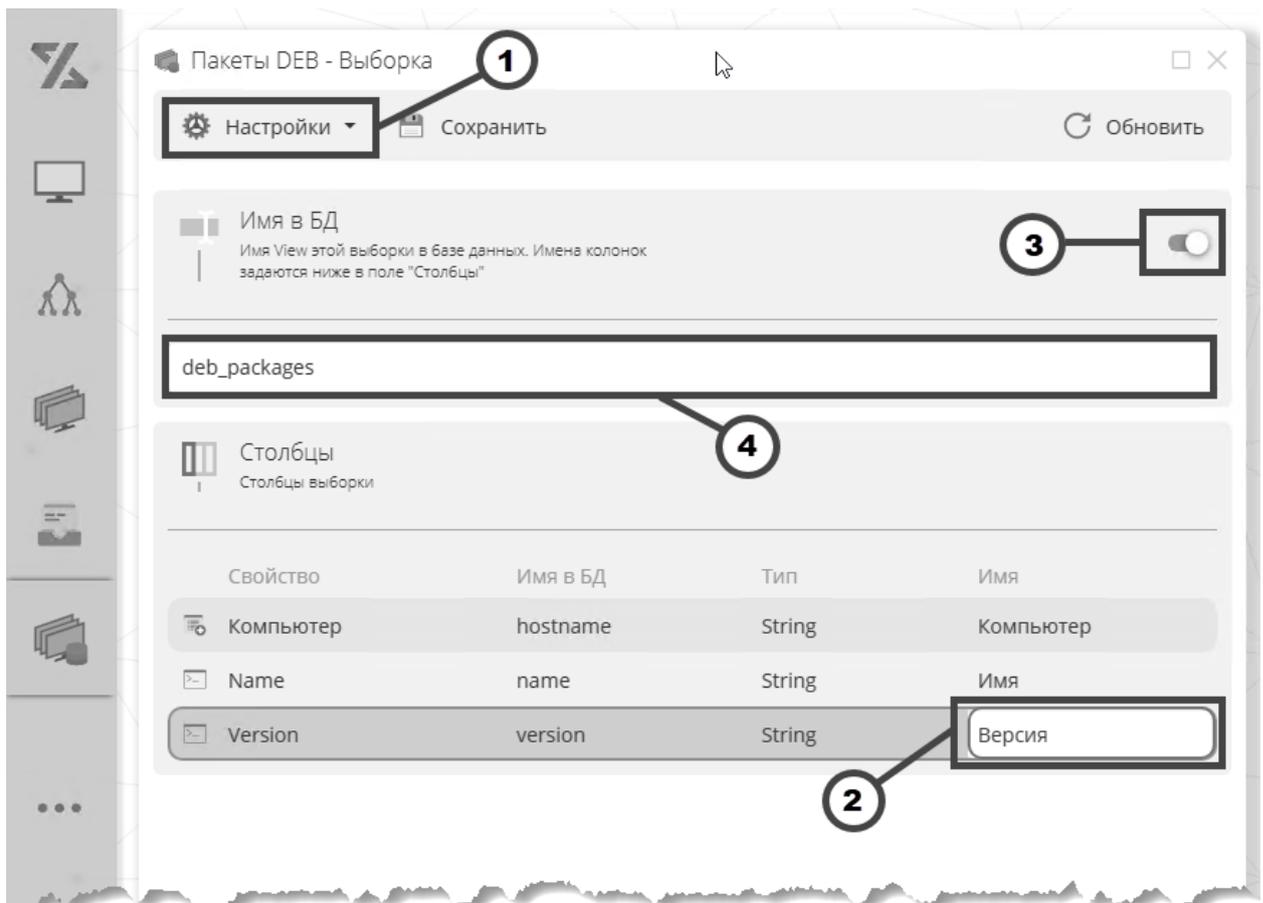


Рисунок 3.60 Редактирование отображаемых имен и типов свойств

1. Режим **«Настройки»** – предоставляет функционал редактирования отображаемых имен и типов свойств.
2. Нажатие на строку в колонке **«Имя»** или **«Тип»** переводит строку в режим редактирования (в нашем примере – редактирования отображаемого имени свойства «Version»).
3. Нажатие на переключатель в строке **«Имя в БД»** сделает доступной возможность редактирования имени выборки в базе данных.
4. Строка в режиме редактирования имени выборки в базе данных.

Общий вид окна настройки выборки в режиме **«Фильтры и предпросмотр»** приведен на рисунке 3.61.

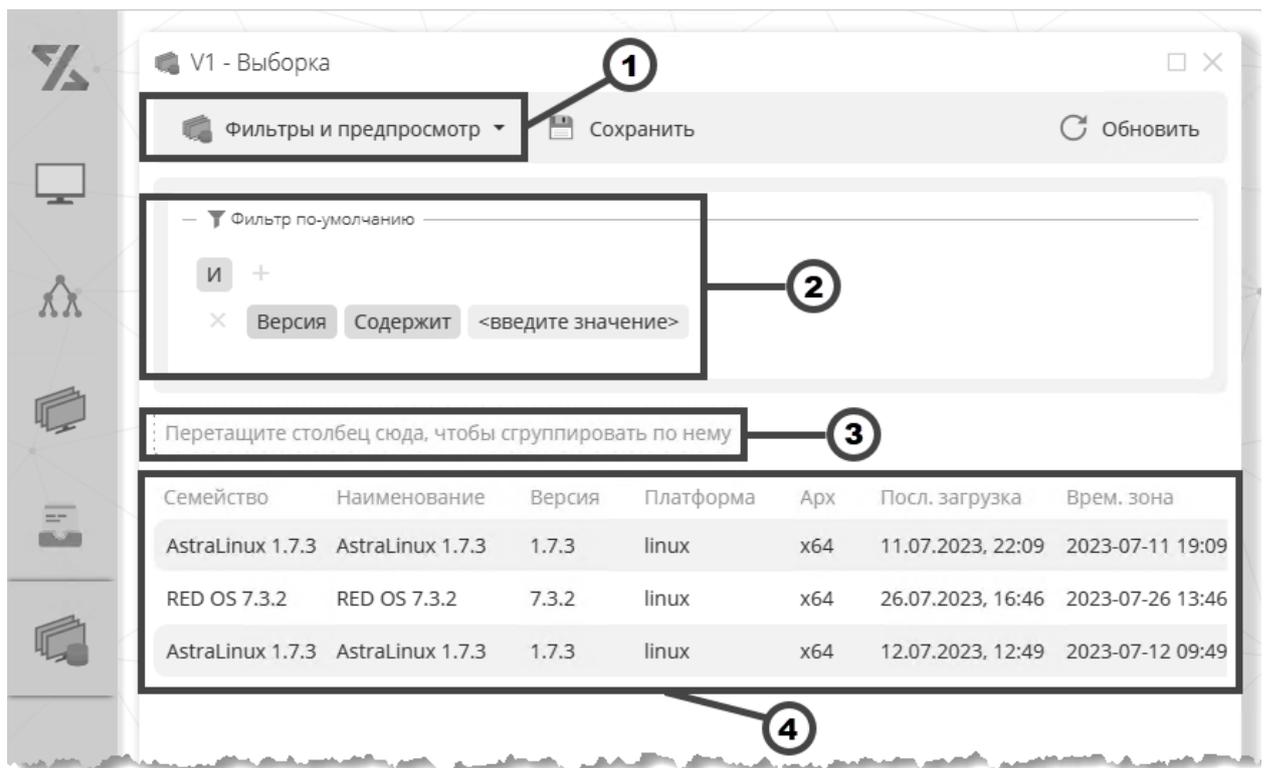


Рисунок 3.61 Настройка фильтрации по умолчанию

1. Режим **«Фильтры и предпросмотр»** – предоставляет функционал настройки фильтрации для выборки по умолчанию.
2. **Конструктор фильтра** – предназначен для редактирования критериев фильтрации. Использование конструктора фильтра описано в разделе 2.2.4.
3. **Область группировки** – перетаскивание заголовка столбца из таблицы на эту область приводит к группировке таблицы по этому столбцу.
4. **Предварительный просмотр** – таблица, отображающая данные в выборке с примененным фильтром.

 **Совет**

Перетаскиванием заголовка столбца между заголовками других столбцов можно менять порядок расположения столбцов в таблице.

### 3.5.3 Просмотр выборки

Логическое окно просмотра выборки вызывается двойным кликом по имени выборки в каталоге выборок. Общий вид окна просмотра выборки представлен на рисунке 3.62.

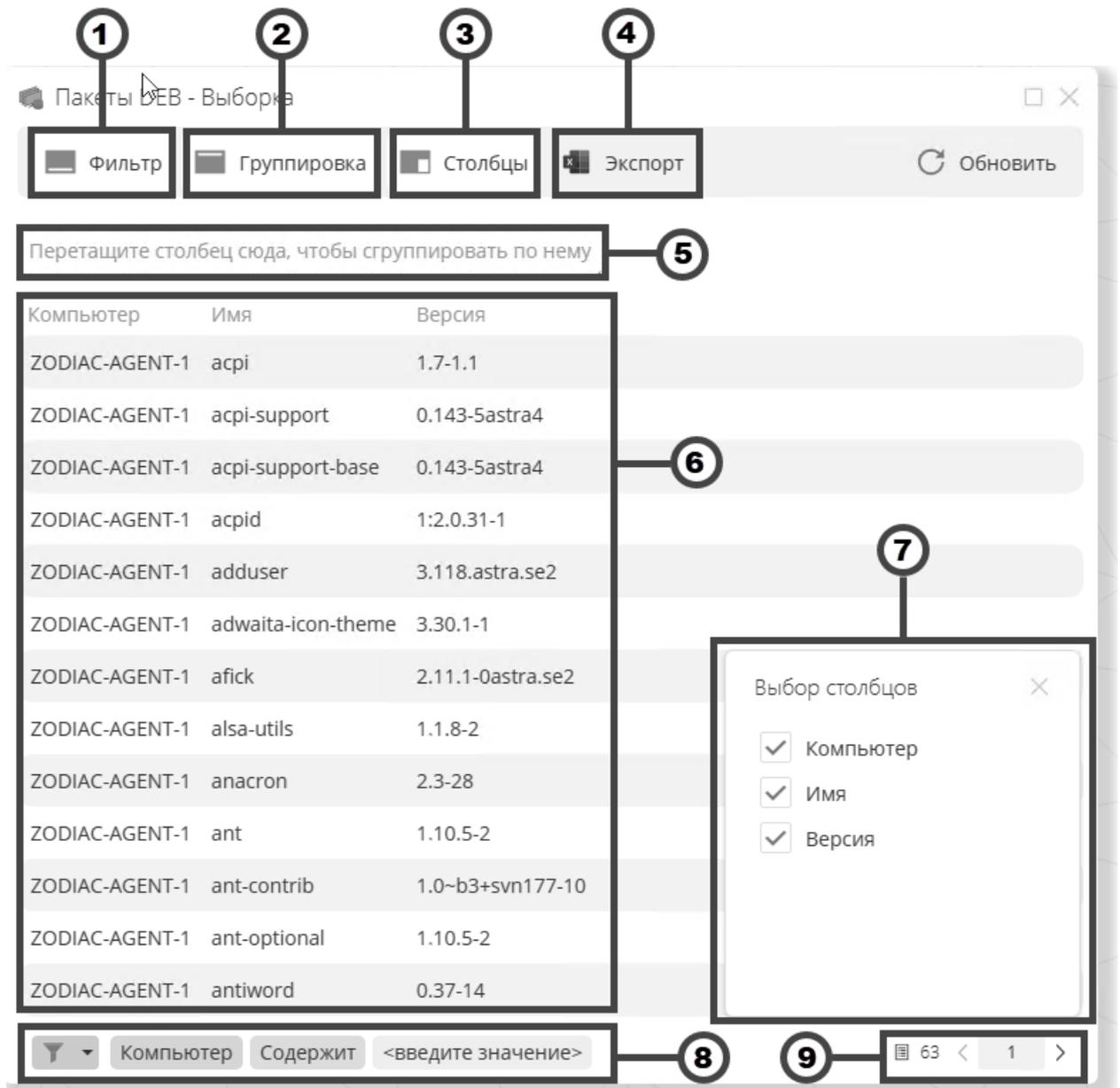


Рисунок 3.62 Общий вид окна просмотра выборки

1. Кнопка «**Фильтр**» – предназначена для включения/выключения видимости фильтра, находящегося в нижней части окна (8).

2. Кнопка **«Группировка»** – предназначена для включения/выключения видимости области группировки, находящейся над таблицей с данными (5).
3. Кнопка **«Столбцы»** – предназначена для включения/выключения видимости окна выбора столбцов (7).
4. Кнопка **«Экспорт»** – предназначена для сохранения данных в выборке в файле формата MS Excel.
5. **Область группировки** – перетаскивание заголовка столбца из таблицы на эту область приводит к группировке таблицы по этому столбцу.
6. Таблица с данными выборки.
7. Окно **«Выбор столбцов»** – предназначено для включения/выключения видимости столбцов в выборке.
8. **Фильтр** – предназначен для фильтрации списка свойств по произвольным критериям. Использование фильтра описано в разделе 2.2.4.
9. **Пейджер** – предназначен для переключения между страницами табличной выборки и отображения номера текущей страницы.

## 3.6 Представления

Режим представляет собой механизм по визуализации данных инвентаризации путем написания SQL-запросов к базе данных.

### 3.6.1 Каталог представлений

Общий вид каталога представлений представлен ниже (рисунок 3.63).

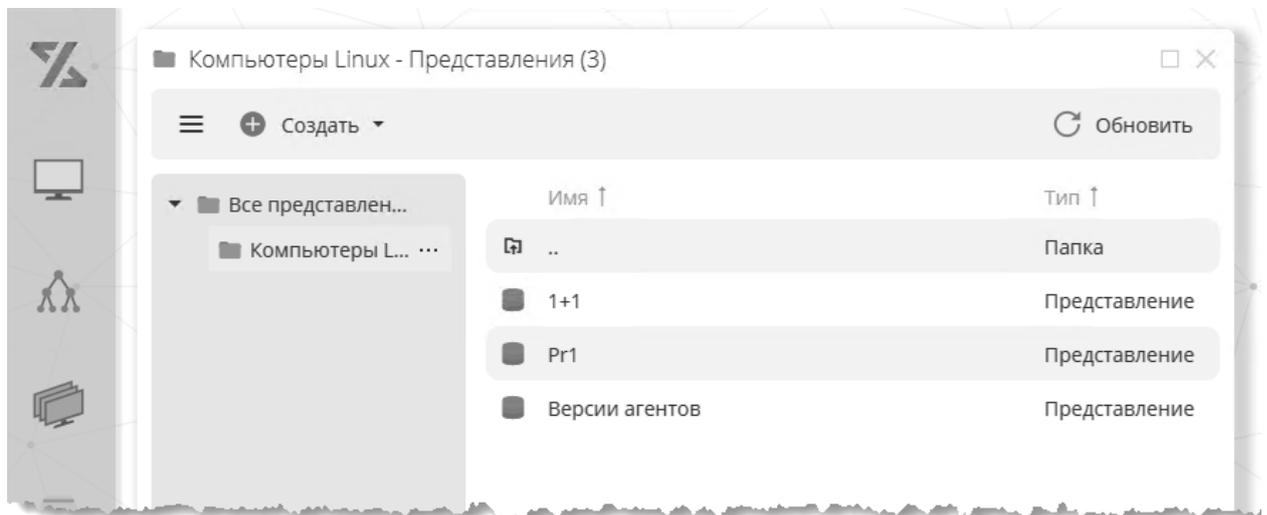


Рисунок 3.63 Общий вид каталога представлений

### 3.6.2 Создание представления

Для создания представления перейдите в каталог представлений и на панели инструментов вызовите окно создания представления (рисунок 3.64).

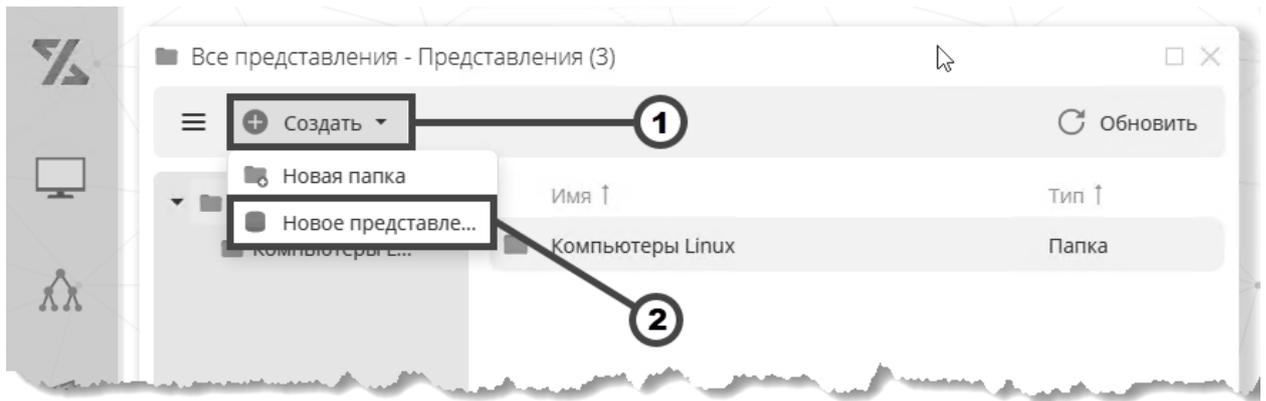


Рисунок 3.64 Вызов окна создания нового представления

1. Кнопка **«Создать»** – предназначена для вызова выпадающего списка выбора команды создания элемента каталога представлений.
2. Кнопка **«Новое представление»** – предназначена для вызова окна создания представления.

В окне создания представления введите имя нового представления (в нашем примере – «Представление 1») и нажмите «ОК» (рисунок 3.65).

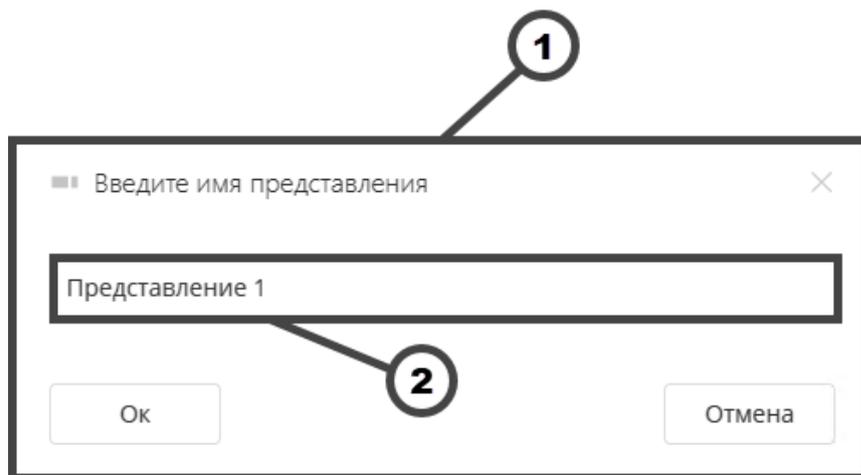


Рисунок 3.65 Окно создания представления

1. Форма – предназначена для ввода полей нового представления.
2. Поле ввода – предназначено для ввода имени нового представления.

В результате предыдущего действия откроется логическое окно нового представления с полем для ввода SQL-запроса. Далее следует произвести настройку представления в соответствии с пунктом 3.6.3.

### 3.6.3 Настройка представления

Окно настройки представления открывается сразу после действия по созданию представления (раздел 3.6.2) или после открытия существующего представления на редактирование одним из способов (рисунок 3.66).

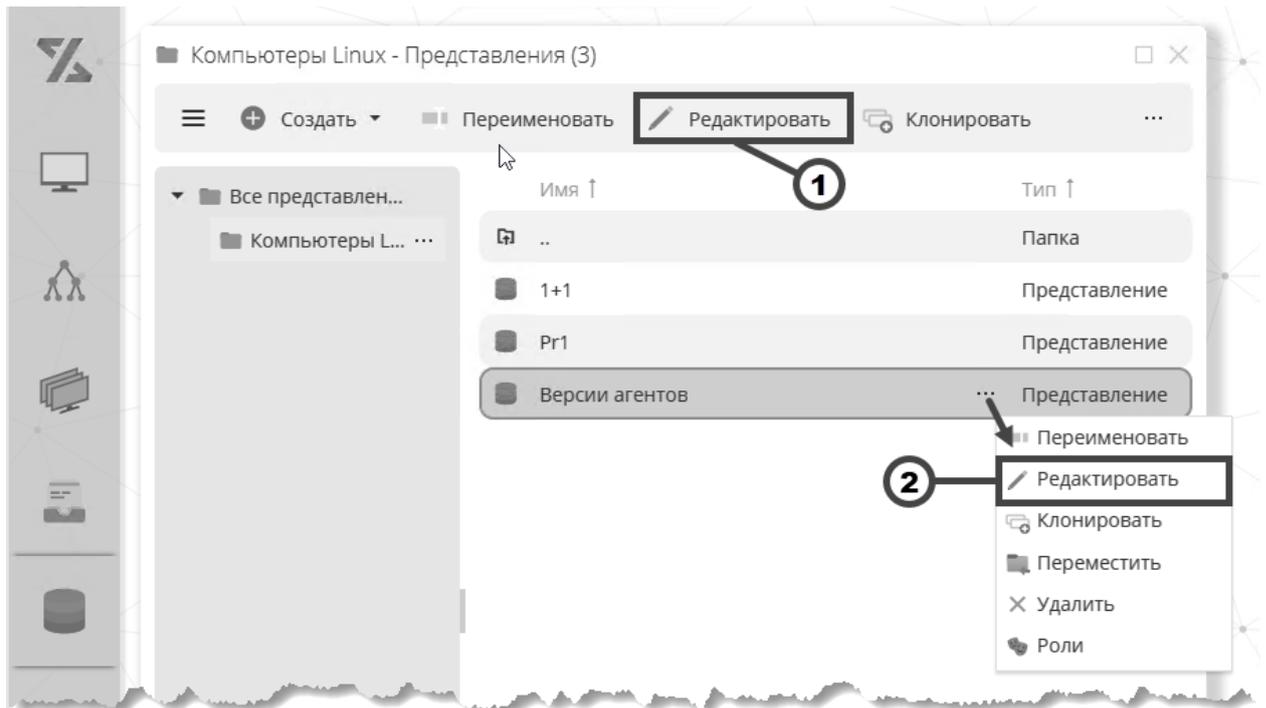


Рисунок 3.66 Вызов окна редактирования представления

1. Вызов окна редактирования представления производится нажатием кнопки в панели инструментов каталога представлений.
2. Вызов окна редактирования представления производится кликом на контекстное меню в имени представления, а затем нажатием кнопки в контекстном меню.

По умолчанию, окно настройки представления открывается в режиме **«Источник данных»** (1) с полем для ввода SQL-запроса (2) (рисунок 3.67). При написании запроса система автоматически подсказывает имена таблиц и их полей.

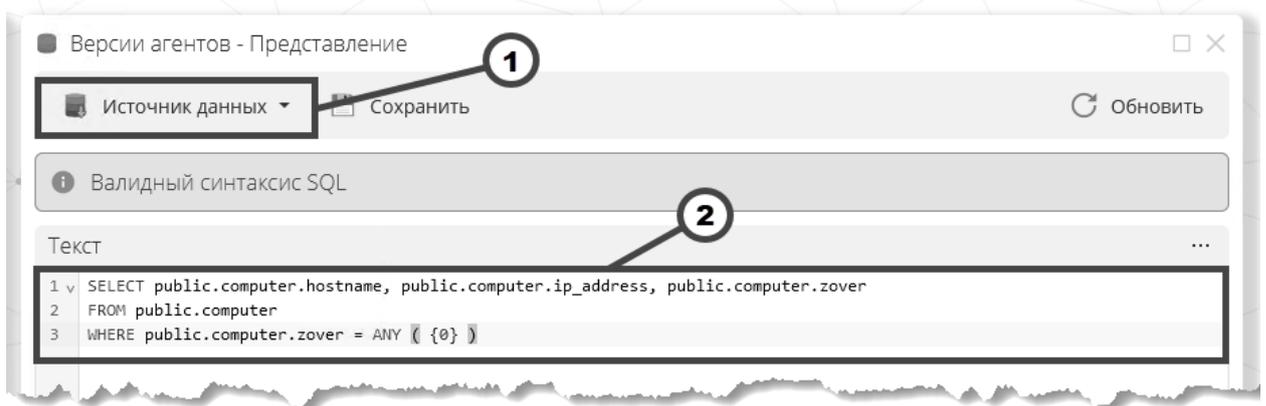


Рисунок 3.67 Окно настройки представления

## ! Осторожно

При написании SQL-запроса не используйте SELECT \*, так как система не поддерживает данный функционал. Указывайте названия столбцов вручную.

Далее можно использовать несколько режимов для настройки представления (рисунок 3.68).

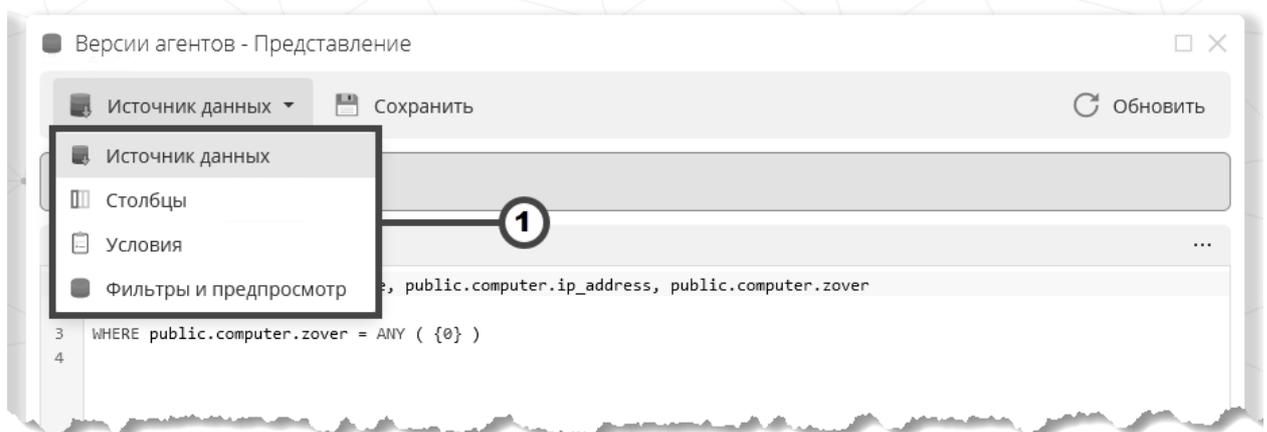


Рисунок 3.68 Список режимов настройки представления

Режим «Столбцы» для настройки отображения таблицы представления (рисунок 3.69).

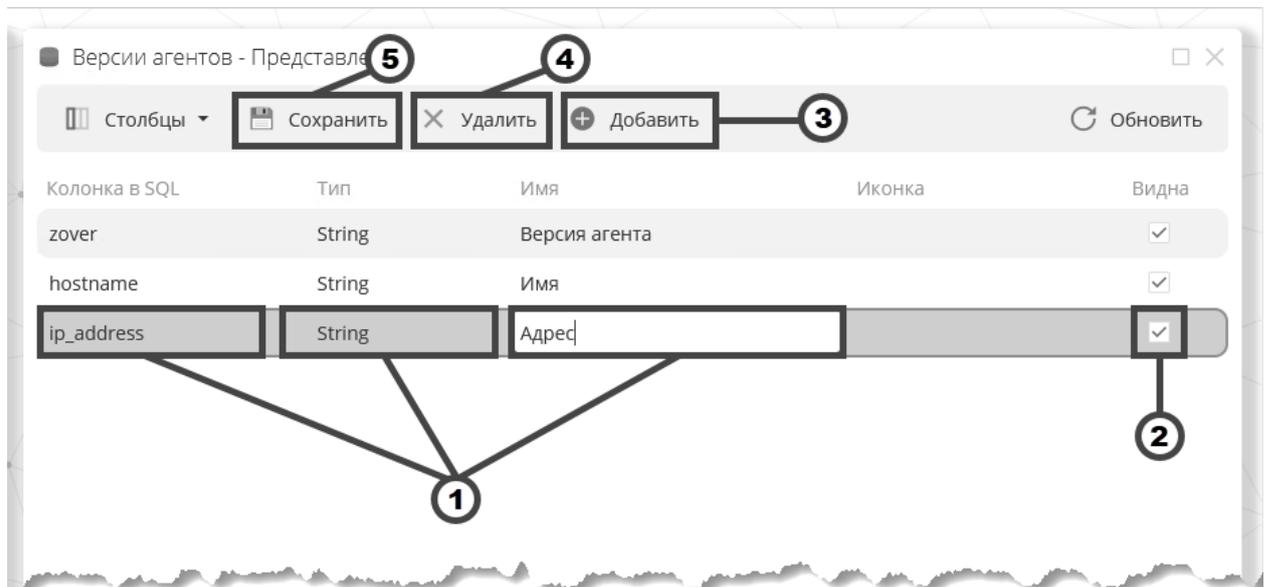


Рисунок 3.69 Режим настройки представления «Столбцы»

1. Функционал для редактирования и настройки данных, которые будут отображаться в представлении.
2. Настройка отображения/скрытия отдельных колонок таблицы.
3. Кнопка «**Добавить**» - предназначена для включения/выключения в представление колонки из БД.

4. Кнопка «Удалить» - предназначена для удаления из представления колонки из БД.
5. Кнопка «Сохранить» - предназначена для сохранения настроек режима.

Режим «Условия» предназначен для редактирования справочника, то есть тех параметров, которые предлагаются на выбор при пользовании представлением (рисунок 3.70).



Рисунок 3.70 Режим настройки представления «Условия»

1. Функционал для редактирования и настройки данных, которые будут отображаться в условиях представления.
2. При нажатии на данную пиктограмму откроется окно для редактирования справочника. В окне доступны 3 вкладки: «Источник данных» (рисунок 3.71), «Столбцы» (рисунок 3.72), «Конфигурирование и предпросмотр» (рисунок 3.73).
3. При нажатии на данную пиктограмму окно с SQL-запросом очистится, запрос удалится.
4. Сохранение настроек режима.

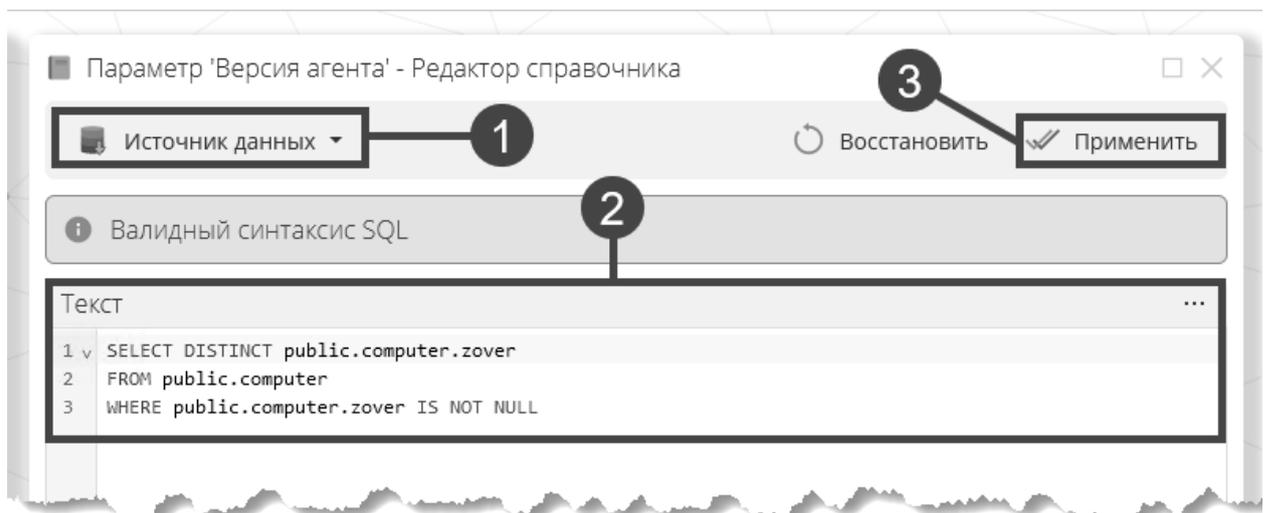


Рисунок 3.71 Окно для редактирования справочника представления – вкладка «Источник данных»

1. Вкладка «Источник данных».
2. Текстовое поле - предназначено для ввода SQL-запроса.

3. Кнопка **«Применить»** - предназначена для сохранения изменений.

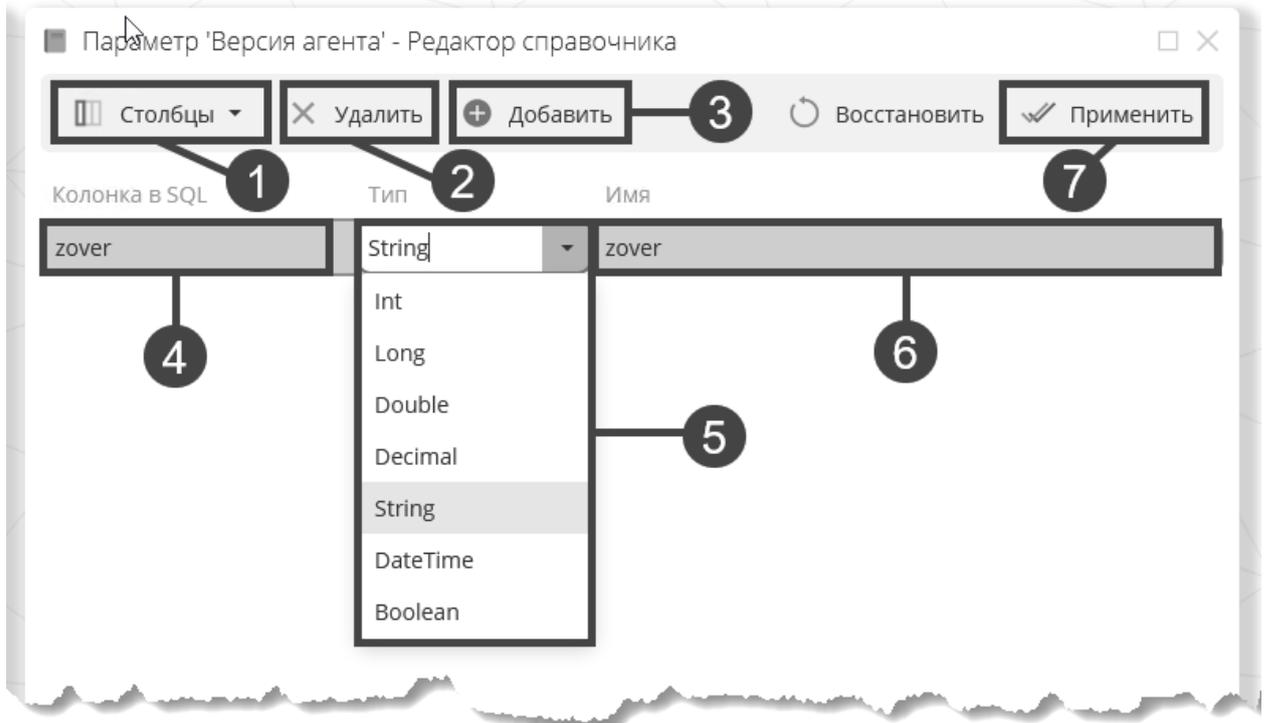


Рисунок 3.72 Окно для редактирования справочника представления – вкладка **«Столбцы»**

1. Вкладка **«Столбцы»**.
2. Кнопка **«Удалить»** - предназначена для удаления записи в таблице.
3. Кнопка **«Добавить»** - предназначена для добавления записи в таблице.
4. Текстовое поле **«Колонка в SQL»** - предназначено для ввода названия колонки из БД.
5. Выпадающий список **«Тип»** - предназначен для указания типа данных.
6. Текстовое поле **«Имя»** - предназначено для ввода отображаемого имени.
7. Кнопка **«Применить»** - предназначена для применения изменений.

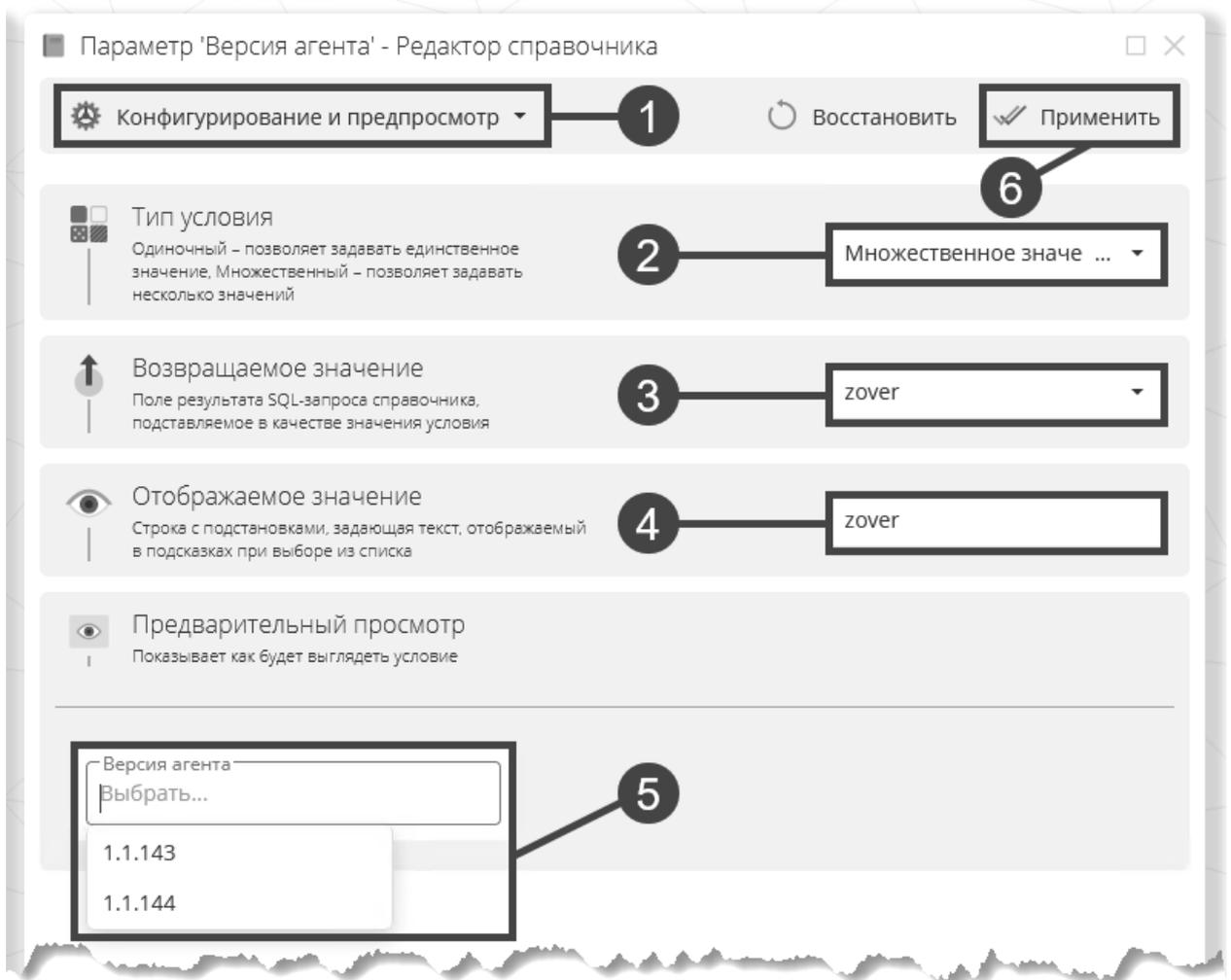


Рисунок 3.73 Окно для редактирования справочника представления – вкладка «Конфигурирование и предпросмотр»

1. Вкладка «**Конфигурирование и предпросмотр**»
2. Выпадающий список «**Тип условия**» – предназначен для выбора одиночного или множественного значения отображаемых условий справочника.
3. Выпадающий список «**Возвращаемое значение**» – предназначен для выбора значения, подставляемого в качестве условия в исходный SQL-запрос представления.
4. Текстовое поле «**Отображаемое значение**» - предназначено для указания имени колонки из запроса, по которой осуществляется выбор из списка.
5. Поле «**Предварительный просмотр**» - предназначено для предварительного просмотра справочника условий.
6. Кнопка «**Применить**» - предназначена для применения изменений.

Следующий режим «**Фильтры и предпросмотр**» предоставляет функционал настройки фильтрации для представления по умолчанию (рисунок 3.74).

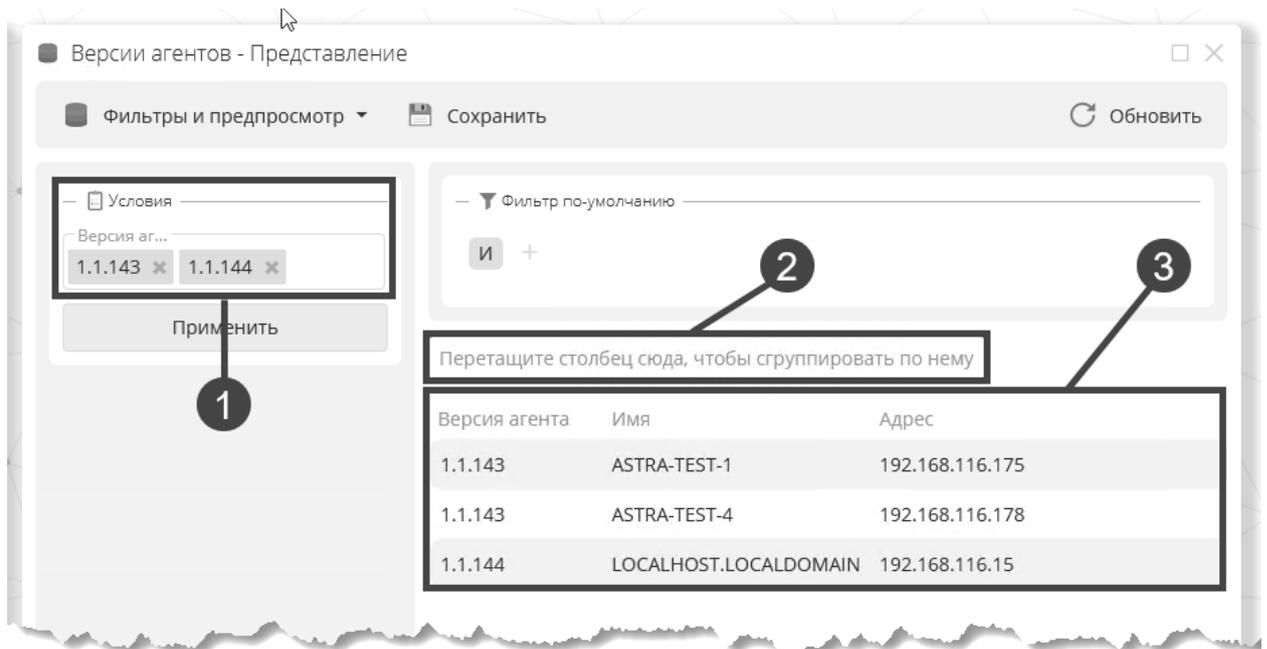


Рисунок 3.74 Режим "Фильтры и предпросмотр"

1. Режим **«Условия»** предназначен для редактирования критериев фильтрации.
2. **Область группировки** – перетаскивание заголовка столбца из таблицы на эту область приводит к группировке таблицы по этому столбцу.
3. **Предварительный просмотр** – таблица, отображающая данные в представлении с примененным фильтром.

## 3.7 Пользователи

### 3.7.1 Каталог пользователей

Каталог пользователей формируется автоматически по результатам работы элемента директивы **«Инвентаризация пользователей»**, описанного в разделе 3.4.9.

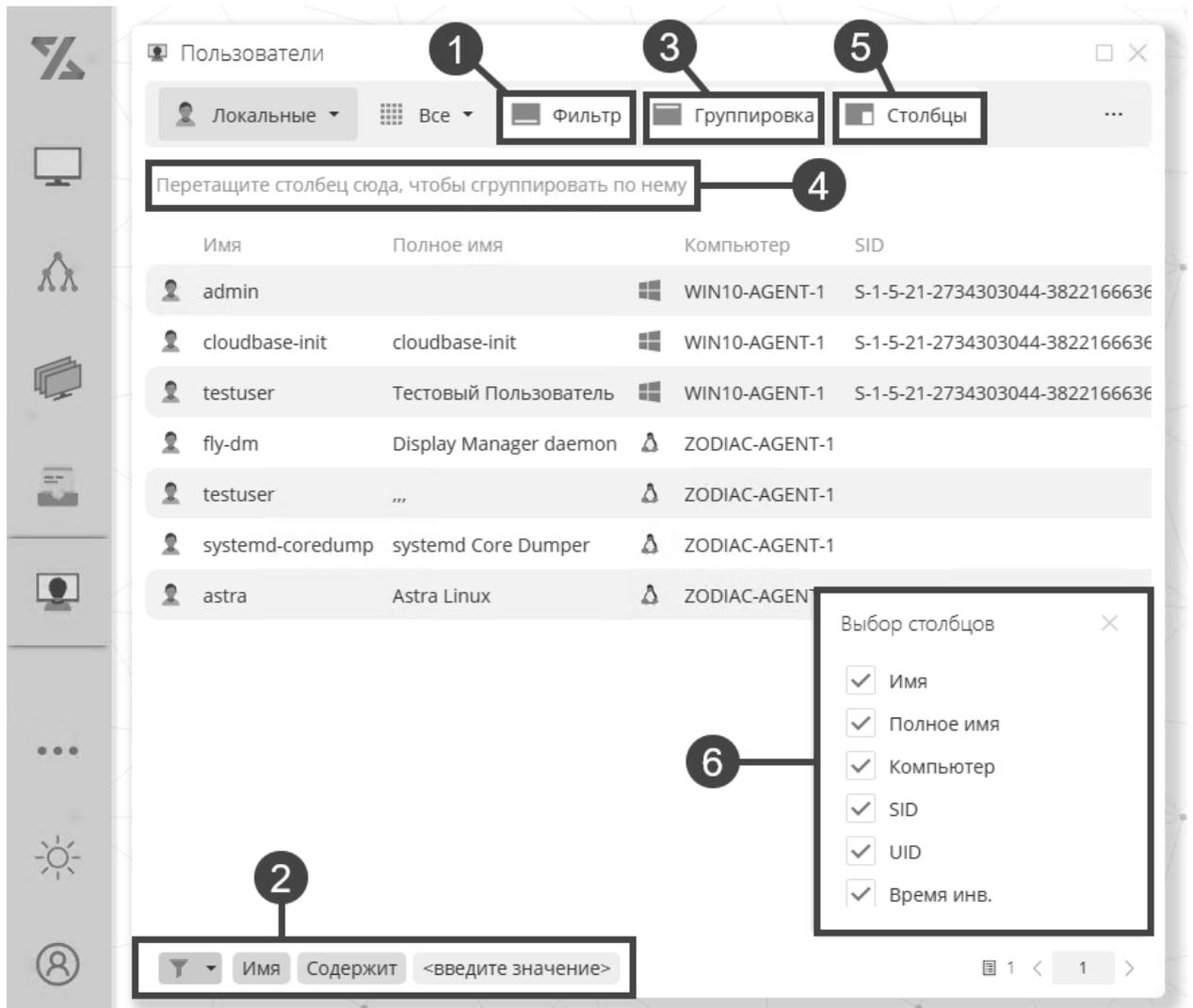


Рисунок 3.75 Окно "Пользователи"

1. Кнопка **«Фильтр»** – предназначена для включения/выключения видимости фильтра, находящегося в нижней части окна (2).
2. **Фильтр** – предназначен для фильтрации списка свойств по произвольным критериям. Использование фильтра описано в разделе 2.2.4.
3. Кнопка **«Группировка»** – предназначена для включения/выключения видимости области группировки, находящегося над таблицей с данными (4).
4. Область **группировки** – перетаскивание заголовка столбца из таблицы на эту область приводит к группировке таблицы по этому столбцу.
5. Кнопка **«Столбцы»** – предназначена для включения/выключения видимости окна выбора столбцов (6).
6. **Окно «Выбор столбцов»** – предназначено для включения/выключения видимости столбцов в каталоге пользователей.

Система автоматически собирает данные о пользователях, которые хотя бы 1 раз заходили на тот или иной сервер или рабочую станцию, и выполняет их автоматическую

категоризацию по трем категориям: «Доменные» учетные записи, «Локальные», «Системные» (рисунок 3.76).

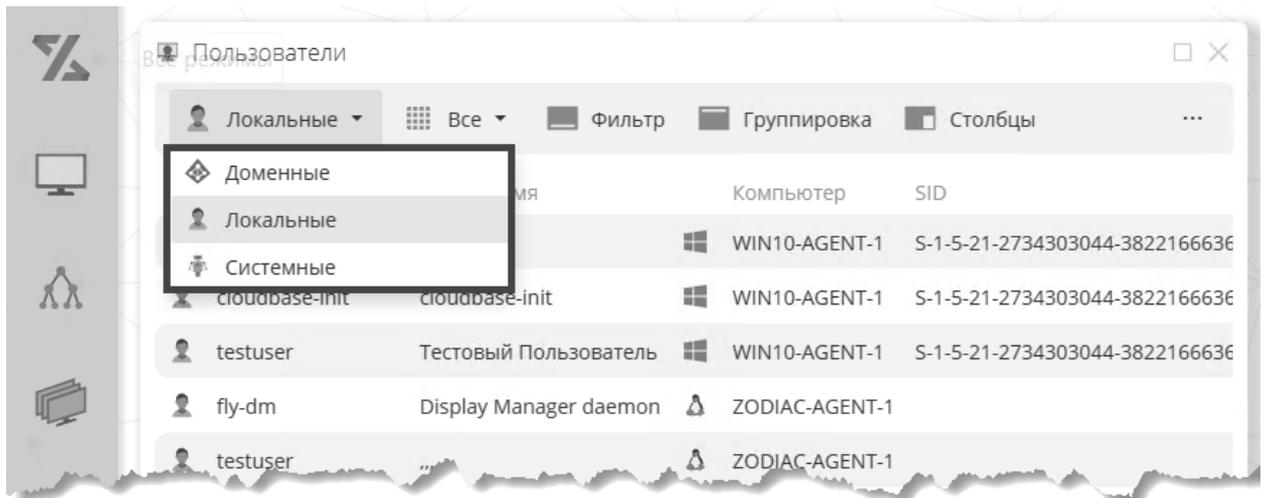


Рисунок 3.76 Выбор категории пользователей

Также система позволяет настроить фильтрацию отображения пользователей по используемой ОС: «Все», «Windows», «Linux» (рисунок 3.77).

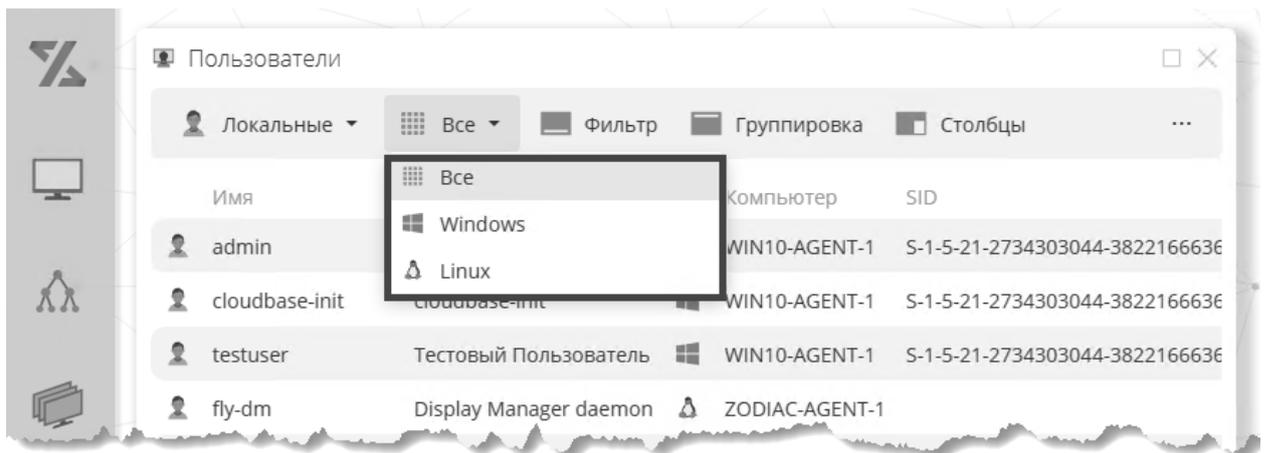


Рисунок 3.77 Отображение пользователей по ОС

## 3.8 Подключения

### 3.8.1 Каталог подключений

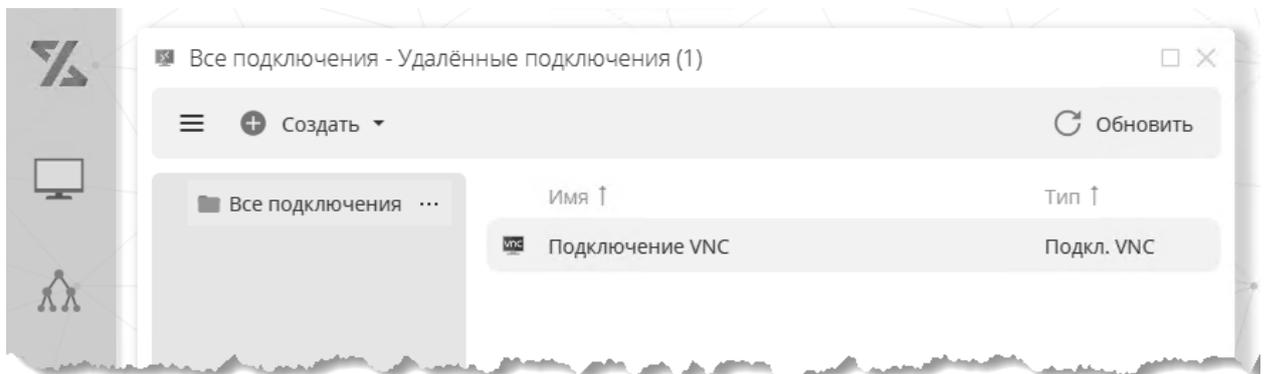


Рисунок 3.78 Общий вид окна "Подключения"

Система поддерживает удаленные подключения как по визуальным, так и по текстовым протоколам. Подключиться можно к любой машине, которая входит в список «Все компьютеры» или в список конкретной группы. Подключиться можно как к машинам, которые покрыты агентами, так и к машинам, которые не покрыты, - при помощи механизма именованных подключений.

Для подключений поддерживается режим полного экрана, визуализация скорости подключения и различные варианты использования клавиатуры.

### 3.8.2 Создание нового подключения

Для создания подключения перейдите в «Подключения» на панели инструментов и вызовите окно создания подключения (рисунок 3.79).

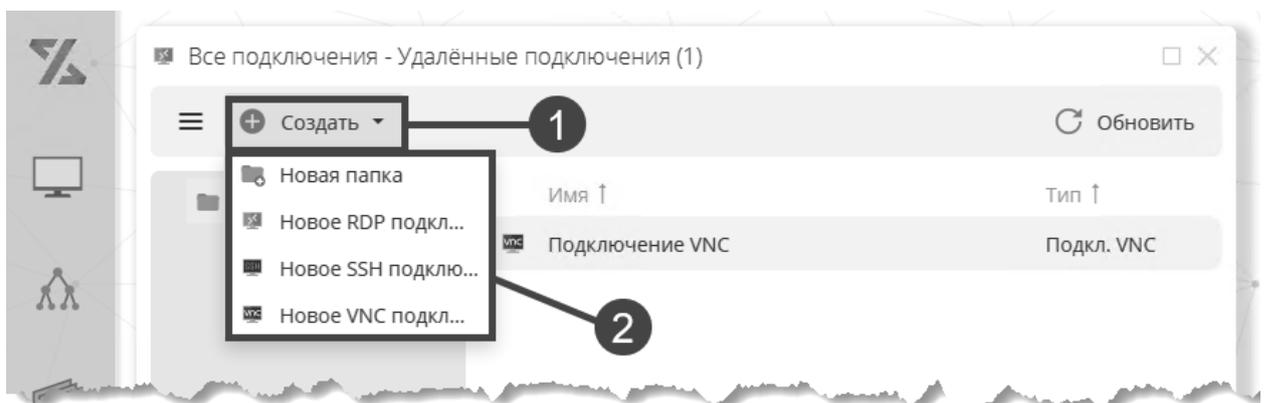


Рисунок 3.79 Вызов окна создания нового подключения

1. Команда **«Создать»** предназначена для вызова выпадающего списка выбора команды создания элемента каталога подключений.
2. Команды выпадающего списка предназначены для создания нового типа **RDP**, **SSH** или **VNC** подключения.

В окне создания подключения введите имя подключения и нажмите «ОК» (рисунок 3.80).



Рисунок 3.80 Окно создания подключения

1. Форма – предназначена для ввода полей нового подключения.
2. Поле ввода – предназначено для ввода имени нового подключения.

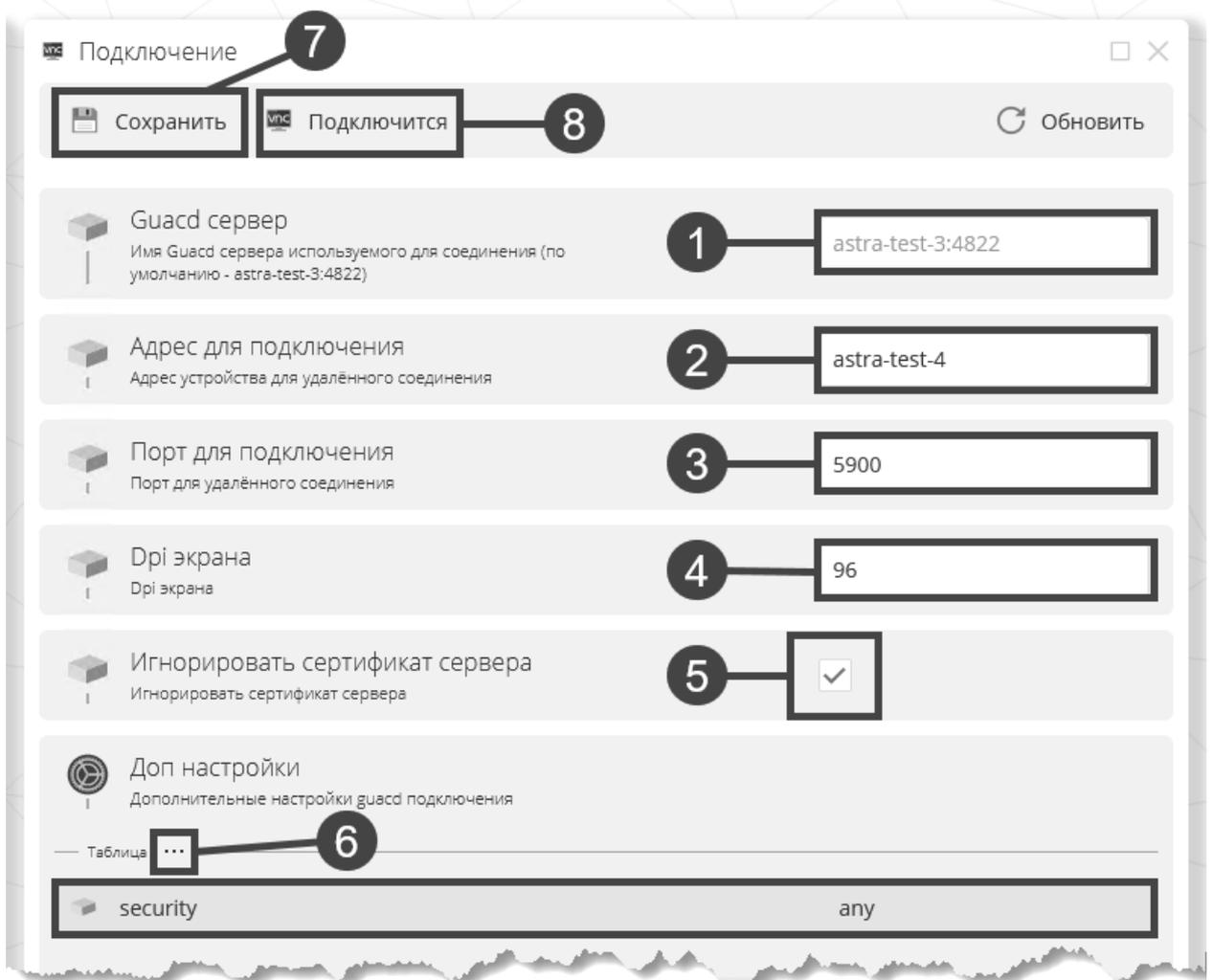


Рисунок 3.81 Окно настройки подключения

1. Текстовое поле «**Guacd сервер**» заполнено по умолчанию, но имеет возможность редактирования. Определяет сервер, через который осуществляется подключение.

2. Текстовое поле «**Адрес для подключения**» предназначено для ввода адреса для подключения.
3. Текстовое поле «**Порт для подключения**» предназначено для ввода порта для подключения.
4. Текстовое поле «**Дрi экрана**» заполнено по умолчанию, но имеет возможность редактирования.
5. Поле «**Игнорировать сертификат сервера**» имеет область для установки/снятия отметки.
6. Кнопка «...» - предназначена для вызова контекстного меню для добавления новой записи.
7. Кнопка «**Сохранить**» предназначена для сохранения изменений настроек подключения.
8. Кнопка «**Подключиться**» предназначена для того, чтобы произвести подключение.



Рисунок 3.82 Окно подключения

1. Форма – предназначена для авторизации системы «Зодиак.АйТиЭм» в Guacd сервере.
2. Поле ввода – предназначено для ввода логина и пароля подключения к **Guacd серверу**, через который производится подключение к удаленному компьютеру.
3. Кнопка «**Продолжить без авторизации**» предназначена для подключения к Guacd серверу, которое не требует ввода логина и пароля.

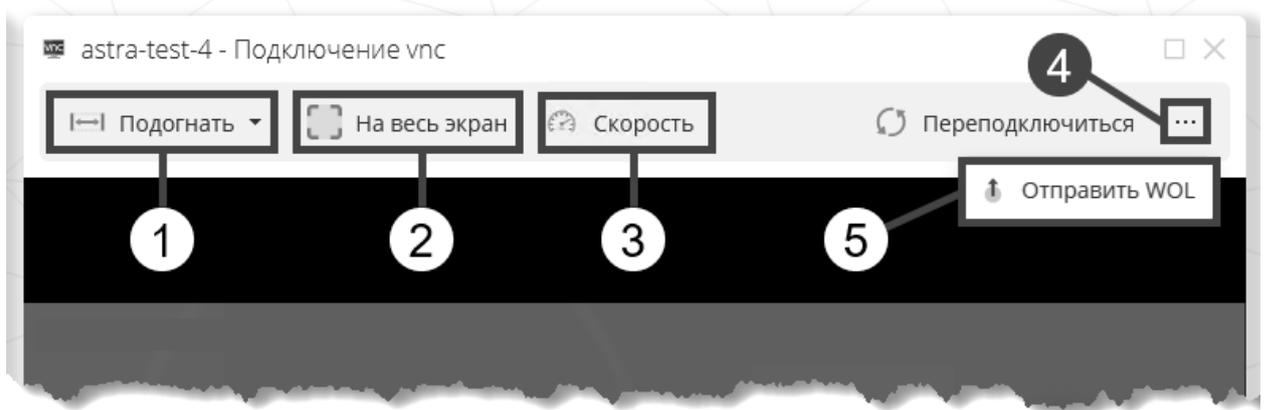


Рисунок 3.83 Окно активного подключения

1. Кнопка «**Подогнать**» - предназначена для изменения отображения экрана в подключении.
2. Кнопка «**На весь экран**» - предназначена для развертывания отображения подключения на весь экран.
3. Кнопка «**Скорость**» - предназначена для визуализации скорости подключения.
4. Кнопка «...» - предназначена для вызова выпадающего списка с кнопкой «**Отправить WOL**» (5) для отправки Wake-on-Lan команды на удаленный ПК.

## 3.9 Роли

### 3.9.1 Каталог ролей

В рамках ролевой модели системы можно поддерживать различные варианты баланса между ведением ролей в рамках внешнего SSO-провайдера, так и ведением роли внутри «Зодиак.АйТиЭм».

Ролевая модель должна быть включена в конфигурации сервера администрирования (см. руководство по установке системы «Зодиак.АйТиЭм»).

Общий вид каталога ролей представлен на рисунке (рисунок 3.84).

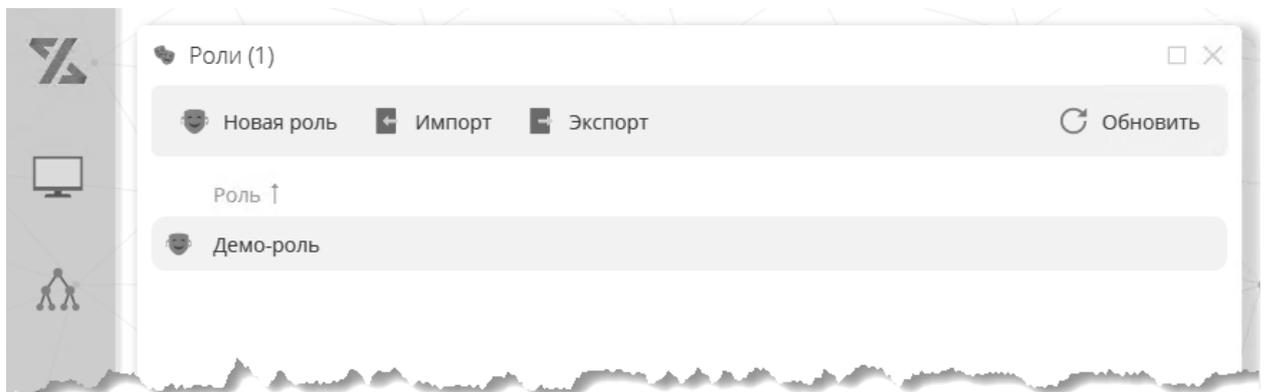


Рисунок 3.84 Каталог ролей

### 3.9.2 Создание новой роли

Для создания новой роли необходимо нажать на кнопку «Новая роль» в контекстном меню.

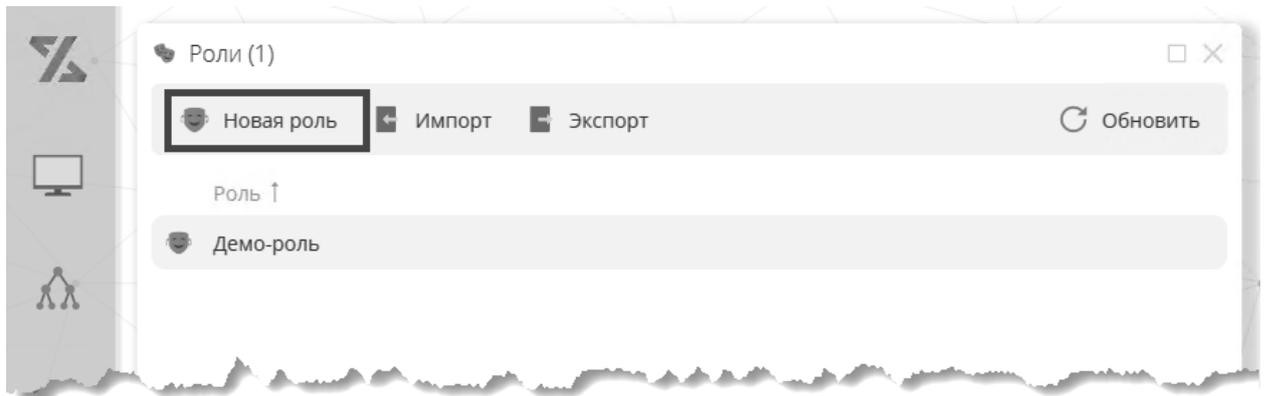


Рисунок 3.85 Создание новой роли

В окне создания роли введите имя роли и нажмите «ОК» (рисунок 3.86).

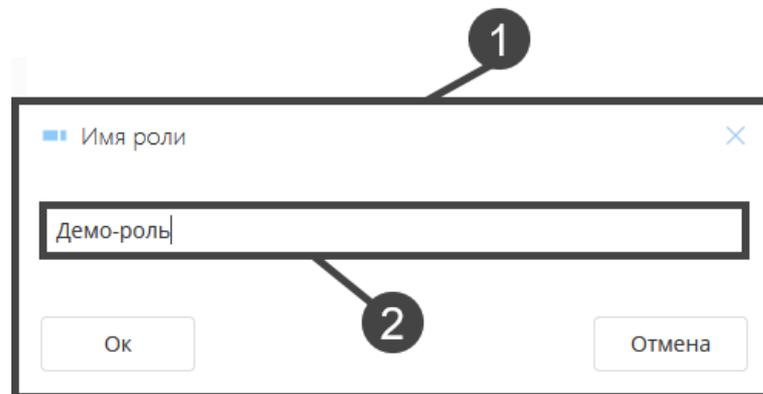


Рисунок 3.86 Окно создания роли

1. Форма – предназначена для ввода полей новой роли.
2. Поле ввода – предназначено для ввода имени новой роли.

Далее откроется логическое окно настройки роли (рисунок 3.87).



Рисунок 3.87 Настройка роли (1)

1. Поле **«Условия»** - предназначено для ввода JSON пути SSO токенов. При использовании KeyCloak, в путях, предлагаемых по умолчанию, можно просто заменить название роли на то, которое задано в KeyCloak.
2. **Переключатели** – предназначены для активации путей. По умолчанию установлены в положение **«активны»**.
3. Кнопка **«...»** - предназначена для добавления нового JSON пути.
4. Кнопка **«...»** - предназначена для добавления нового разрешения.
5. Выпадающий список **«Тип элемента»** - предназначен для выбора раздела системы «Зодиак.АйТиЭм», к которому необходимо предоставить доступ данной роли. **Если раздел не добавлен в поле «Разрешения», то данной роли доступ к нему не предоставляется.**



Рисунок 3.88 Создание роли (2)

1. Выпадающий список **«Тип доступа»** - предназначен для настройки параметров доступа для данной роли.
2. Текстовое поле **«Glob-шаблон путей»** – предназначено для установки элементов внутри раздела, к которым настраивается доступ.
3. **Переключатели** – предназначены для активации разрешений. По умолчанию установлены в положение «активны».
4. Кнопка **«Сохранить»** - предназначена для сохранения изменений в настройке роли.

## 3.10 Планировщик задач

### 3.10.1 Каталог задач

В системе представлен планировщик серверных задач, с помощью которого можно осуществлять автоматическое исполнение задач не только на стороне агентов, но и на стороне сервера.

В текущей версии представлен один вид задач – автоматический пересчет групп. Можно задать расписание и выбрать обновление всех автоматических групп.

При активации режима **«Планировщик задач»** открывается логическое окно **«Все задачи»** (рисунок 3.89).

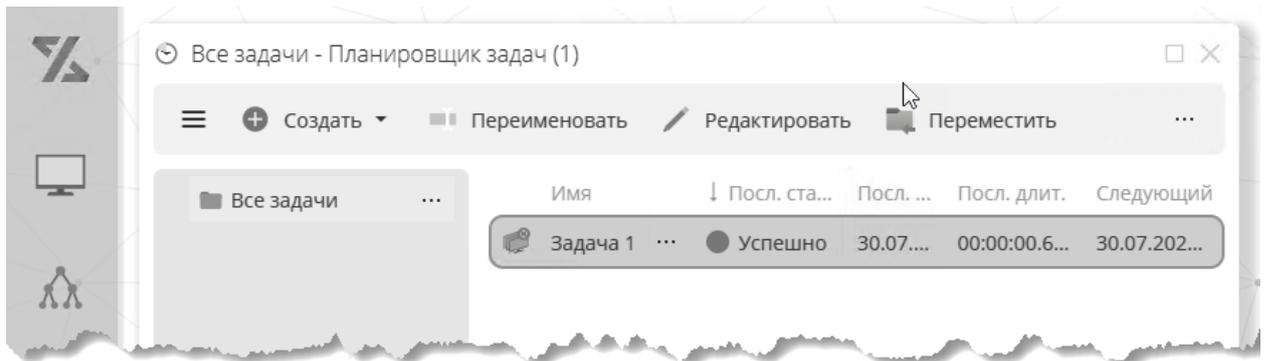


Рисунок 3.89 Вид общего списка задач планировщика

### 3.10.2 Создание новой задачи

Для создания новой задачи в окне «Все задачи» вызовите окно создания задачи (рисунок 3.90).

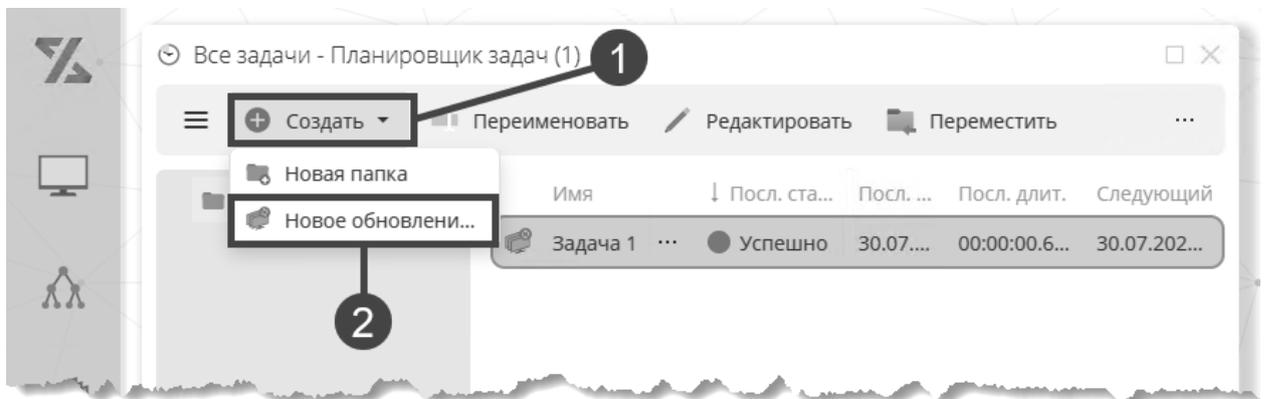


Рисунок 3.90 Создание задачи

1. Кнопка **«Создать»** - предназначена для вызова выпадающего списка выбора команды создания элемента планировщика задач.
2. Кнопка **«Новое обновление авто-групп»** - предназначена для вызова окна создания задачи.

В окне создания задачи введите имя задачи и нажмите «ОК» (рисунок 3.91).

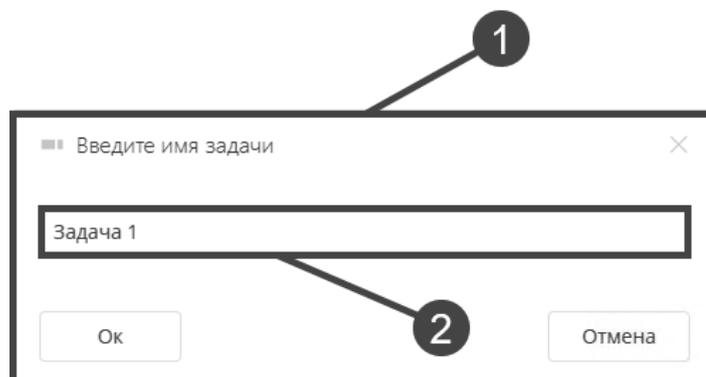


Рисунок 3.91 Окно создания задачи

1. Форма - предназначена для ввода полей новой задачи.
2. Поле ввода - предназначено для ввода имени новой задачи.

После нажатия кнопки «ОК» откроется логическое окно создания задачи планировщика (рисунок 3.92).



Рисунок 3.92 Окно настройки задачи обновления авто-групп

1. В поле «**Расписание**» необходимо установить периодичность выполнения задачи. Подробнее о работе с панелью в разделе 2.2.5.
2. В поле «**Обновлять все авто-группы**» есть возможность активировать переключатель. Тогда вкладка для выбора авто-групп из списка будет недоступна.
3. Кнопка «**Сохранить**» - предназначена для сохранения настроек задачи.
4. При нажатии на поле «**Авто-группы**» можно выбрать авто-группы из списка.

## 3.11 Версии агентов

### 3.11.1 Общий вид окна «Версии агентов»

В системе представлен функционал, дающий информацию о версиях агентов (рисунок 3.93).

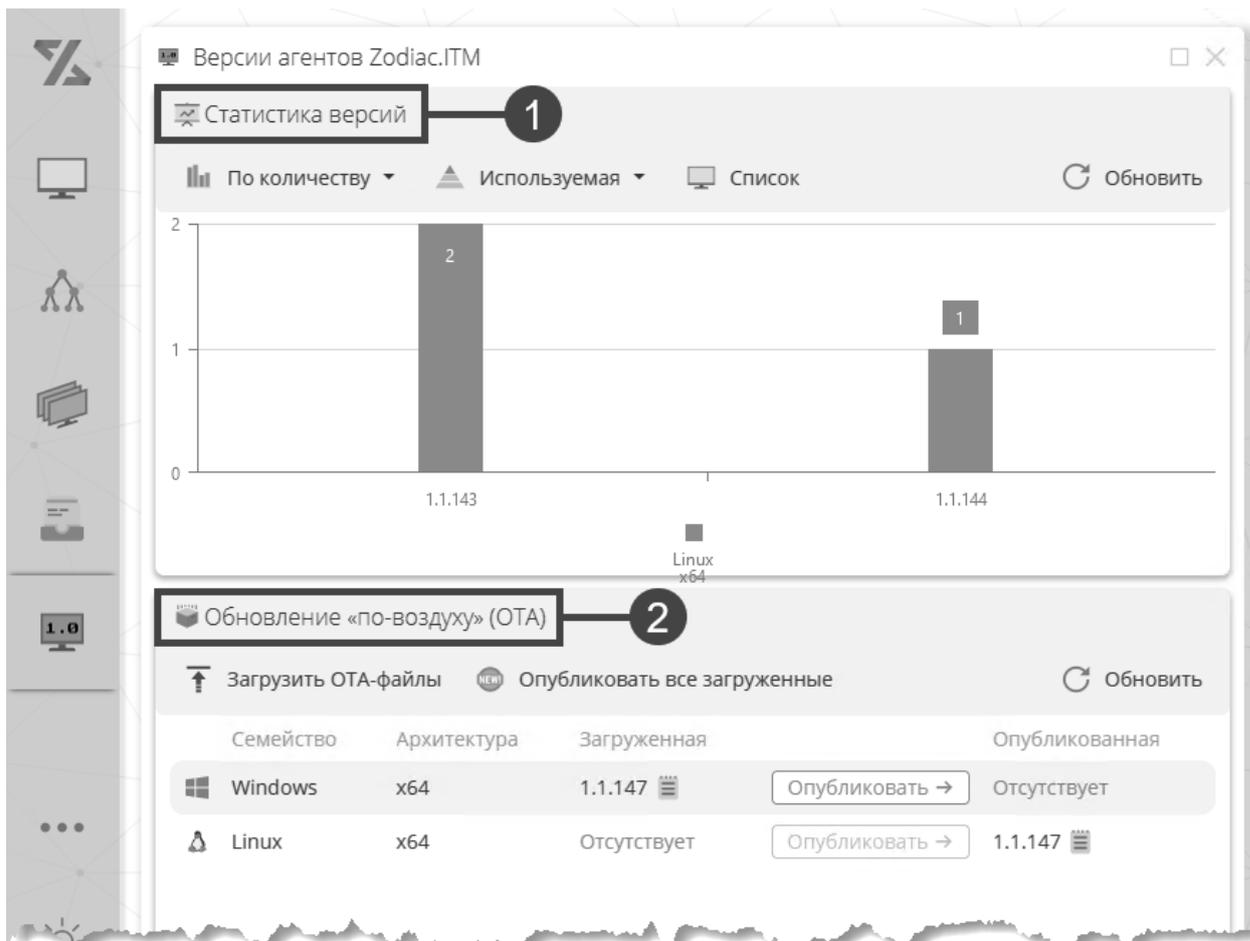


Рисунок 3.93 Окно "Версии агентов"

1. Верхняя область окна отображает **статистику версий агентов**. При нажатии на столбец диаграммы отображается список компьютеров с выбранным агентом.
2. Нижняя область окна дает доступ к функционалу **обновления «по воздуху» (OTA)**.

### 3.11.2 Статистика версий

При вызове выпадающего списка выбора варианта просмотра статистики доступны 2 режима (рисунок 3.94).

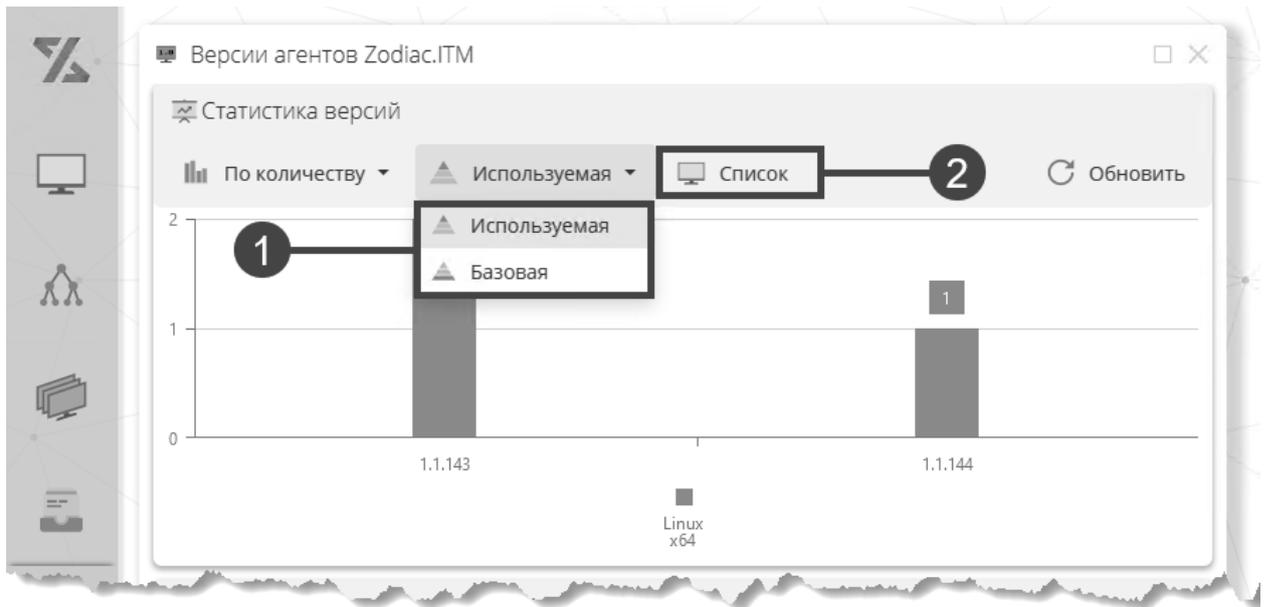


Рисунок 3.94 Выбор отображения версий агентов

1. Агент обновляется без само-переустановки, т.е. путем дополнения папки 'ver'. Файлы, установленные изначально через ms/rpm/deb остаются нетронутыми. Версия, которая была установлена через инсталлятор, а не при помощи автообновления, называется **базовой**. Информация по базовой версии нужна для диагностики.
2. При нажатии на **«Список»** откроется логическое окно «Все компьютеры» (пункт 3.1.1).
3. При нажатии на часть столбца в диаграмме откроется логическое окно «Компьютеры», в котором будут указаны компьютеры с соответствующей версией.

### 3.11.3 Обновление «по воздуху» (OTA)

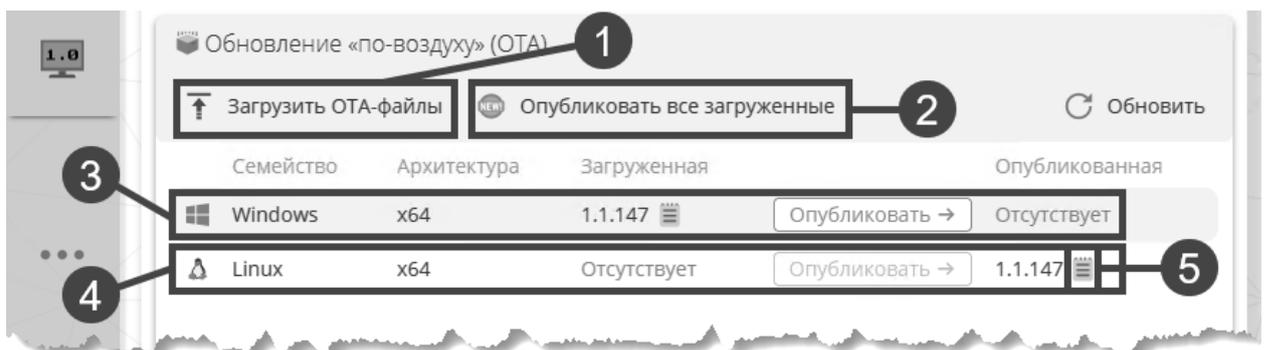


Рисунок 3.95 Обновление «по воздуху» (OTA)

1. Кнопка **«Загрузить ОТА-файлы»** - предназначена для вызова логического окна для загрузки файлов (рисунок 3.96).
2. Кнопка **«Опубликовать все загруженные»** - предназначена для того, чтобы опубликовать сразу все загруженные версии.
3. Данная строка таблицы демонстрирует пример **неопубликованной версии**.

4. Данная строка таблицы демонстрирует пример **опубликованной версии**.
5. **Пиктограмма** – предназначена для вызова окна информации со списком изменений в данной версии.

Для того, чтобы произвести обновление «по воздуху», нажмите на кнопку **«Загрузить ОТА-файлы» (1)** (рисунок 3.95). Откроется логическое окно **«Загрузка новых версий агентов»** (рисунок 3.96). Далее нажмите на кнопку **«Добавить файлы» (1)**, выберите необходимый файл. Он отобразится в списке **(3)**. Затем нажмите кнопку **«Загрузить все в систему» (5)**. Загруженный файл с новой версией агента отобразится в **списке (рисунок 3.95) (3)**. Для того, чтобы опубликовать версию, нажмите кнопку **«Опубликовать»** в строке нужной версии. Агенты произведут обновление автоматически.

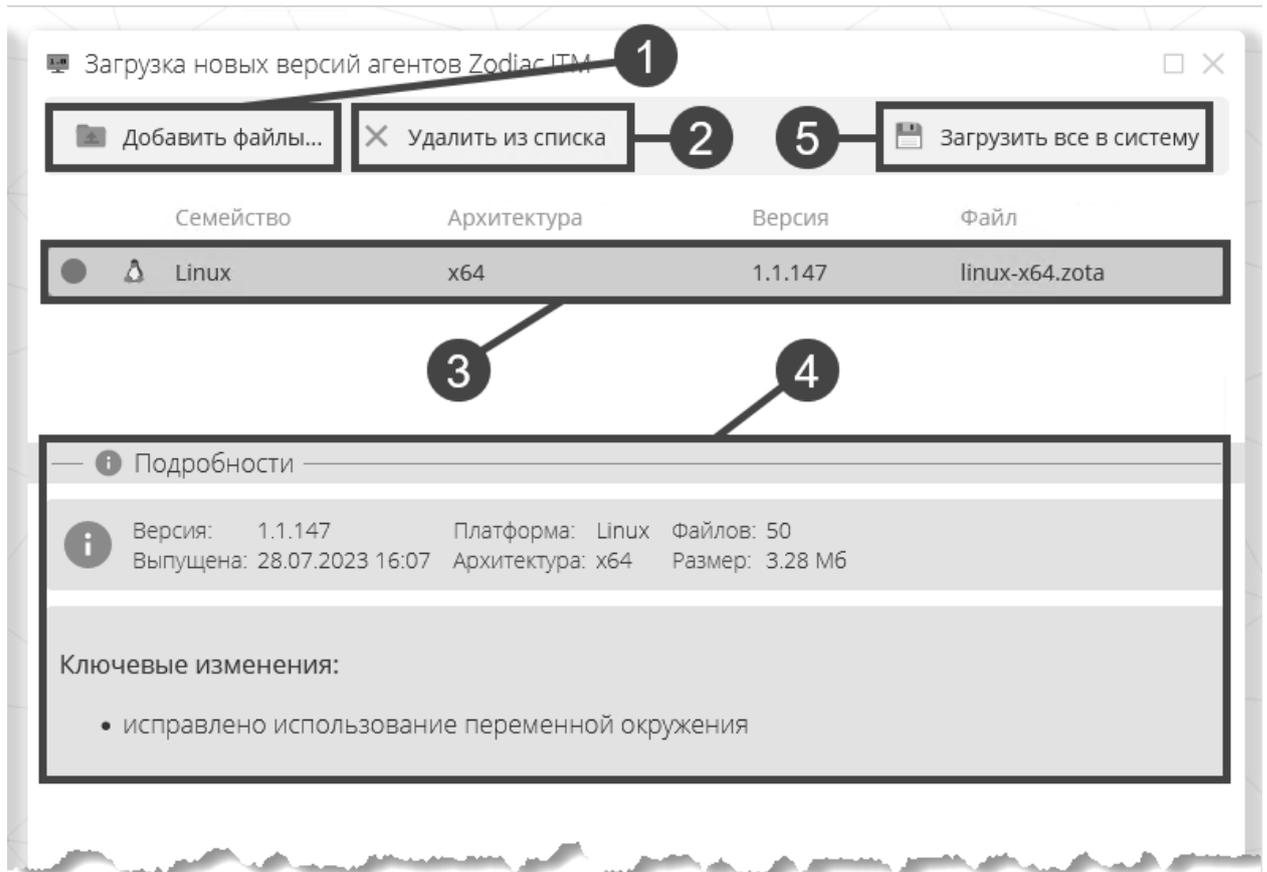


Рисунок 3.96 Загрузка новых версий агентов «Зодиак.АйТиЭм»

1. Кнопка **«Добавить файлы»** - предназначена для вызова логического окна для выбора файлов из каталога.
2. Кнопка **«Удалить из списка»** - предназначена для удаления добавленного файла.
3. Данная строка демонстрирует пример отображения **загруженного файла**.
4. Поле **«Подробности»** - предназначено для отображения информации о загруженном файле и изменениях, которые в нем произведены.
5. Кнопка **«Загрузить все в систему»** - предназначена для загрузки файла в систему.

## 3.12 Границы

### 3.12.1 Каталог границ

Основная задача, которую решает данный режим – сопоставление агентов и точек обслуживания. В качестве алгоритма для определения границ может быть использован произвольный сценарий.

Общий вид каталога границ представлен ниже (рисунок 3.97).

В режиме «Границы» Web-консоли управления «Зодиак.АйТиЭм» предусмотрено переключение между двумя представлениями:

1. **Границы** - на верхнем уровне находятся обнаруженные границы (уникальные строки, которые вернули скрипты), внутри которых указываются скрипты, сформировавшие данную границу.
2. **Сценарии** - на верхнем уровне находятся скрипты, внутри которых указываются обнаруженные границы (уникальные строки, которые вернул скрипт на разных компьютерах).

Двойной щелчок по границе открывает список компьютеров и точек обслуживания, отнесенных к этой границе. Двойной щелчок по скрипту открывает его редактор.

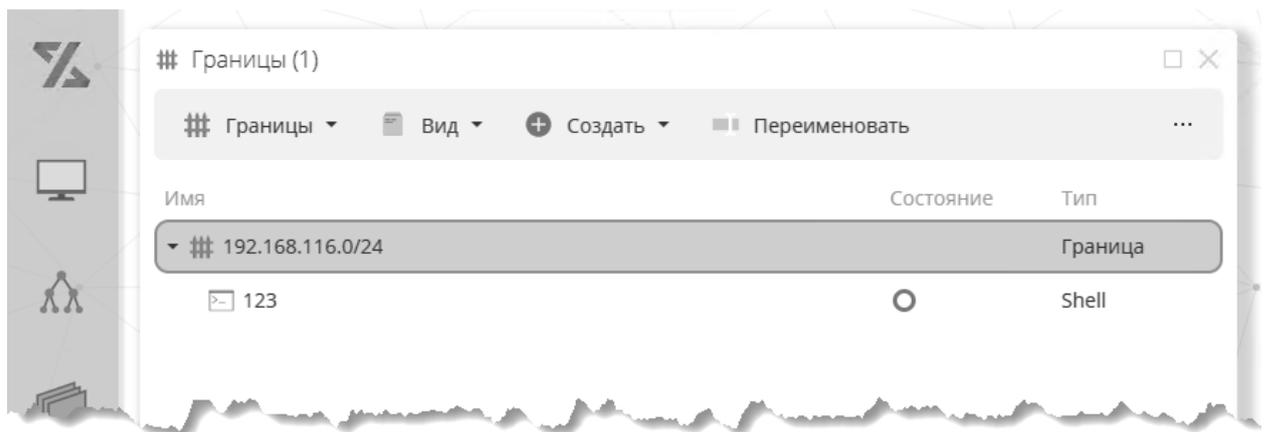


Рисунок 3.97 Общий вид каталога границ

Для настройки любого из перечисленных способов необходимо создать в режиме "Границы" Web-консоли управления «Зодиак.АйТиЭм» новый сценарий (скрипт) определения границ на одном из поддерживаемых языков (JavaScript, CoffeeScript, PowerShell, Shell) (рисунок 3.98).

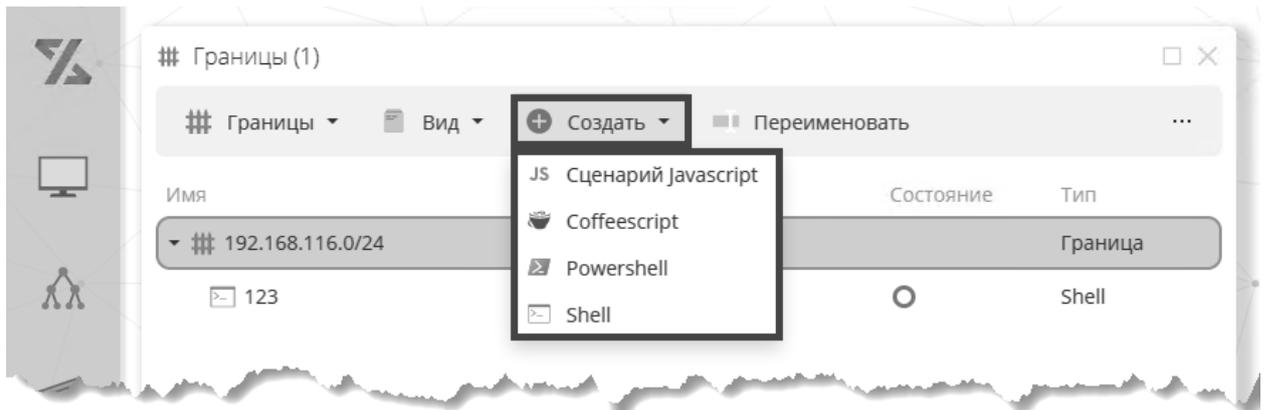


Рисунок 3.98 Создание сценария определения границ



Рисунок 3.99 Окно создания сценария

Далее откроется логическое окно создания нового сценария (рисунок 3.100).



Рисунок 3.100 Создание сценария

1. Окно создания скрипта по умолчанию открывается во вкладке **«Скрипт»**.
2. **Текстовое поле** – предназначено для ввода скрипта. Здесь представлен пример скрипта определения границ на основе маски подсети для Linux.

Пример скрипта определения границ на основе маски подсети для Windows выглядит так:

```
$MainIP = (Get-NetIPAddress -AddressFamily IPv4 | Where-Object
{$_ .AddressState -eq "Preferred" -and $_.ValidLifetime -lt
"24:00:00"});

[IPAddress]$Mask = (([string]'1'* $MainIP.PrefixLength +
[string]'0'*(32- $MainIP.PrefixLength)) -split "(\\d{8})" -match "\\d" |
% {[convert]::ToInt32($_,2)}) -split "\\D" -join "."

$SplitIPAddress = [int[]]@($MainIP.IPAddress -split "\\." -match "\\d")
$SplitMask = $Mask -split "\\." -match "\\d"

$IPBin = ($SplitIPAddress | %
{[convert]::ToString($_,2).PadLeft(8,"0")}) -join "."
$MaskBin = ($SplitMask | % {[convert]::ToString($_,2).PadLeft(8,"0")})
-join "."

$Subnet = ((0..31 | % {@($IPBin -split "" -match "\\d")[$_] -band
@($MaskBin -split "" -match "\\d")[$_]}) -join ' ' -split "(\\d{8})" -
match "\\d" | % {[convert]::ToInt32($_,2)}) -join "."
$CIDR = $Subnet + '/' + $MainIP.PrefixLength

return @{ 'boundaries' = $CIDR }
```

Страница настроек скрипта границ содержит описание для каждого конфигурационного параметра (рисунок 3.101). Каждый скрипт границ может находиться в одном из следующих состояний:

- **отключен** (скрипт не выполняется, его результаты не используются для определения адресов точек обслуживания)
- **включен** (скрипт выполняется, его результаты используются для определения адресов точек обслуживания)
- **тест** (скрипт выполняется, его результаты отображаются в режиме «Границы», но не используются для определения адресов точек обслуживания).

Созданные скрипты границ регулярно исполняются как на всех компьютерах с агентами.

## Предупреждение

Назначение скриптов границ на компьютеры производится автоматически, для этого не требуется использование механизма «Ассоциации».

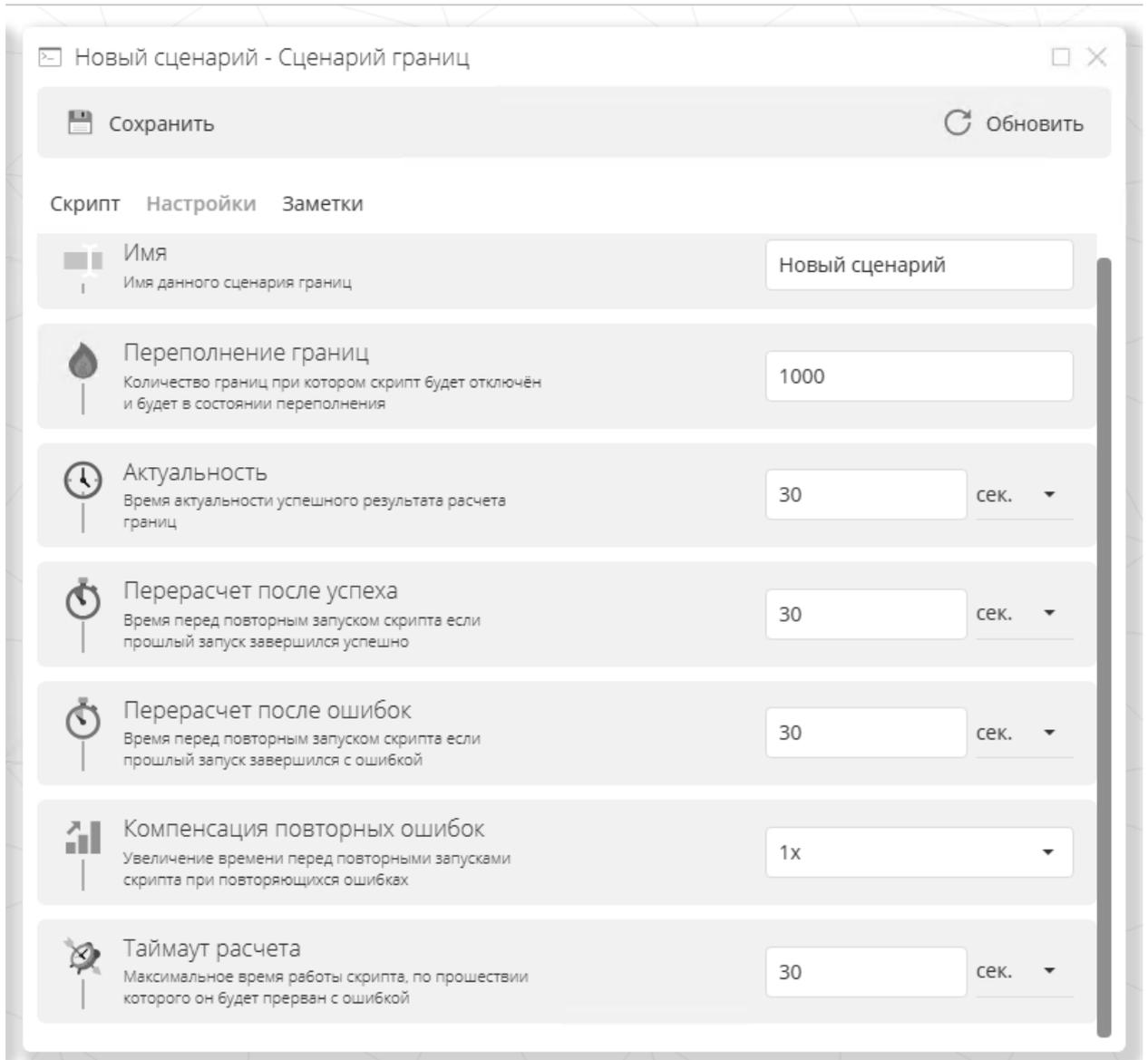


Рисунок 3.101 Страница настроек скрипта границ

В рамках создаваемого скрипта необходимо описать логику определения границ и вернуть результат определения. Форматы возврата результата совпадают со сценариями в директивах:

- для PowerShell - PObject или HashTable,
- для Shell - построчный вывод в env-формате «ключ: значение».

От любого сценария вне зависимости от его языка ожидается присутствие в результатах характеристики **boundaries**. Характеристика «**boundaries**» используется для определения границ с использованием централизованной серверной инфраструктуры. В качестве значения характеристики «boundaries» может выступать любая строка, вычисляемая на управляемом компьютере.

Типичными значениями характеристики «boundaries» являются:

- префикс IPv6;

- маска IPv4;
- сайт AD;
- префикс/суффикс имени компьютера.

Если и компьютер, и точка обслуживания вернули в качестве значения характеристики «boundaries» одну и ту же строку - централизованная серверная инфраструктура Зодиак.АйТиЭм это определяет и сообщает IP-адрес и DNS-имя точки обслуживания компьютеру.

---

### **Примечание**

В характеристике **boundaries** могут быть несколько имен границ.

Максимальное количество границ в результате ограничено числом 10.

Максимальная длина каждой границы – 50 символов.

---

### 3.13 Лог изменений

#### 3.13.1 Каталог лога изменений

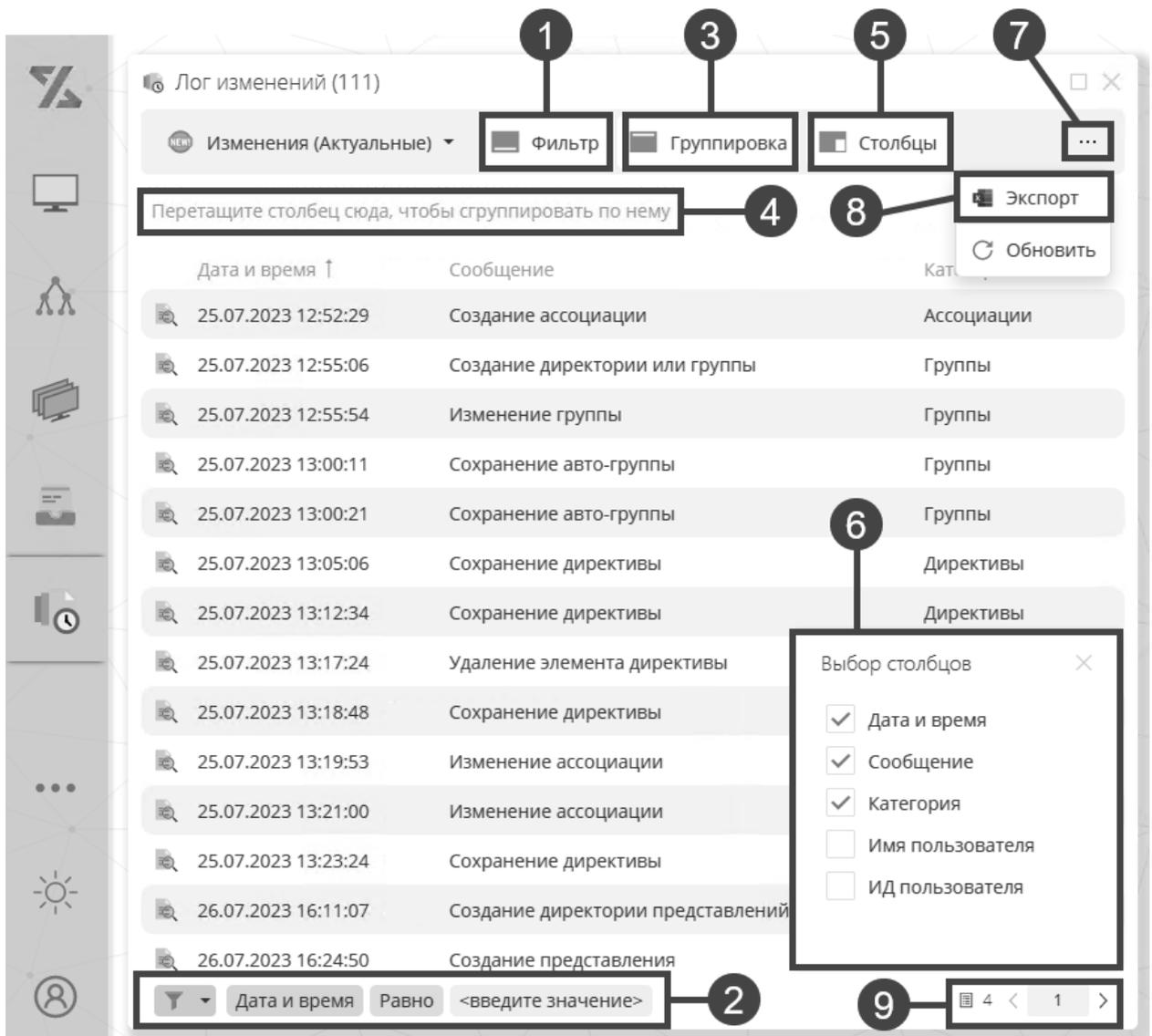


Рисунок 3.102 Окно «Лог изменений»

1. Кнопка «**Фильтр**» - предназначена для включения/выключения видимости фильтра, находящегося в нижней части окна (2).
2. **Конструктор фильтра** - предназначен для настройки визуализации таблицы лога изменений. Использование конструктора фильтра описано в разделе 2.2.4.
3. Кнопка «**Группировка**» – предназначена для включения/выключения видимости области группировки, находящейся над таблицей с данными (4).
4. Область **группировки** – перетаскивание заголовка столбца из таблицы на эту область приводит к группировке таблицы по этому столбцу.
5. Кнопка «**Столбцы**» – предназначена для включения/выключения видимости окна выбора столбцов (6).

6. Окно **«Выбор столбцов»** - предназначено для включения/выключения видимости столбцов в таблице.
7. Кнопка **«...»** - предназначена для вызова контекстного меню таблицы лога изменений.
8. Кнопка **«Экспорт»** - предназначена для сохранения данных в файле формата MS Excel.
9. **Пейджер** – предназначен для переключения между страницами табличного представления и отображения номера текущей страницы.

## 3.14 Серверы

### 3.14.1 Каталог серверов

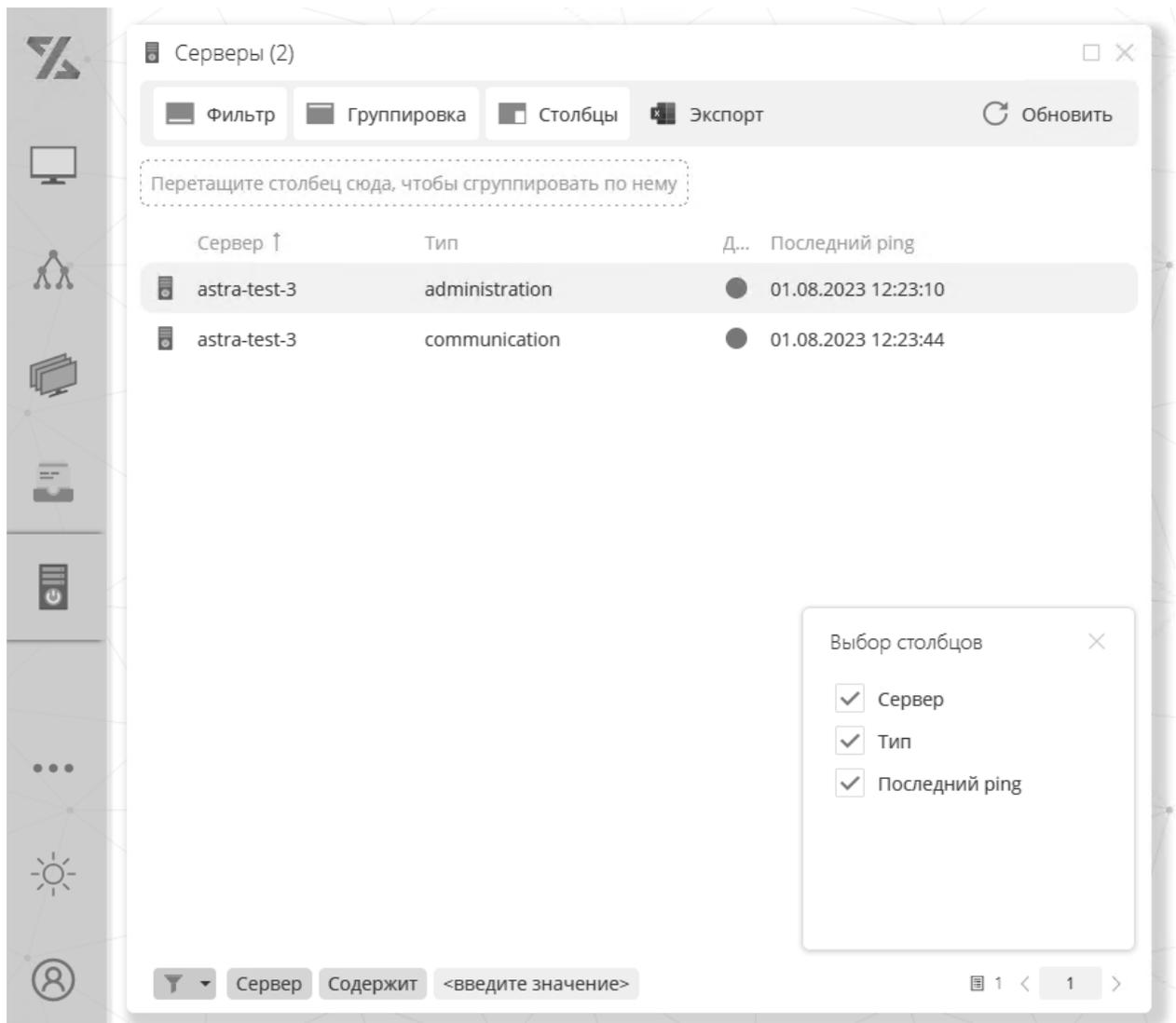


Рисунок 3.103 Общий вид окна "Серверы"

Общий вид окна **«Серверы»** представлен выше (рисунок 3.103) и содержит функционал, аналогичный окну **«Лог изменений»** в пункте 3.14.1.

## 3.15 Соседние устройства

### 3.15.1 Список соседних устройств

Список соседних устройств формируется автоматически по результатам работы элемента директивы «Сканирование сети» описанного в разделе 3.4.7.

Общий вид окна «Соседние устройства» с заполненными данными сканирования сети приведен на рисунке 3.104.



#### Совет

Двойной клик по строке таблицы с данными соседнего устройства приводит к открытию **окна подключения** к этому устройству (разд. 3.15.2).

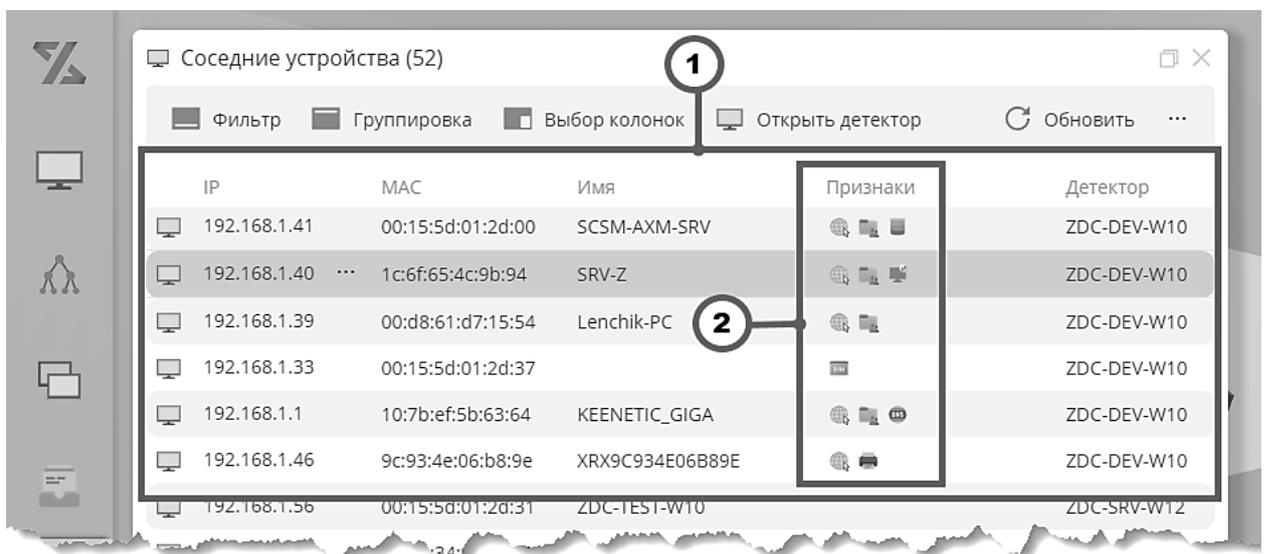


Рисунок 3.104 Общий вид окна «Соседние устройства»

1. Таблица с данными сканирования сети.
2. Колонка «**Признаки**» – набор признаков, обозначающих предположительные роли или функции устройства. Предназначается для быстрого отбора нужных устройств по их функции при использовании в фильтре.

Данные сканирования сети включают в себя следующий фиксированный список параметров, которые можно включать в таблицу с помощью функционала «**Выбор колонок**», а также использовать в конструкторе фильтра:

- **IP** – IP-адрес
- **MAC** – MAC-адрес
- **Имя** – имя устройства
- **Признаки** – теги предположительных ролей/сервисов (Printer, Database Server, DNS Server, File Share), предоставляемых устройством (на основе открытых портов)
- **Детектор** – устройство с агентом «Зодиак.АйТиЭм», проводившим сканирование
- **Производитель MAC**

- Дата регистрации MAC
- Имена – все сетевые имена
- Открытые порты
- Сервисы mdns

### 3.15.2 Подключение к соседним устройствам

Окно подключения к устройству можно вызвать в окне «Соседние устройства» (рис. 3.104) из панели инструментов или контекстного меню.

Общий вид окна подключения представлен на рисунке 3.105.

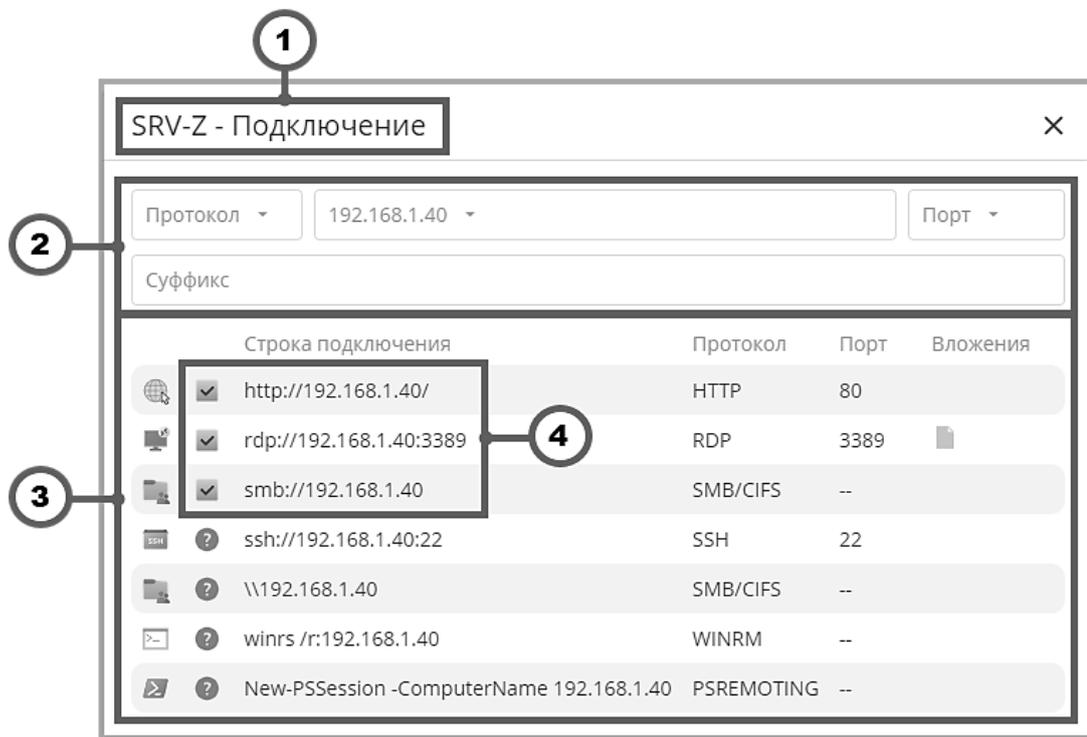


Рисунок 3.105 Окно подключения к устройству

1. **Имя устройства**, к которому осуществляется подключение.
2. **Фильтр** списка подключений.
3. Список **поддерживаемых** видов подключений.
4. **Предположительно доступные** для устройства подключения.

## 3.16 Пакеты развертывания

### 3.16.1 Создание пакета развертывания

Для создания пакета развертывания перейдите в каталог пакетов развертывания и на панели инструментов вызовите окно создания пакета, как показано на рисунке 3.106.



Рисунок 3.106 Вызов окна создания нового пакета развертывания

1. Кнопка **«Создать»** – предназначена для вызова выпадающего списка выбора команды создания элемента каталога пакетов развертывания.
2. Кнопка **«Новый пакет»** – предназначена для вызова окна создания пакета развертывания

В окне создания пакета введите имя пакета и нажмите «ОК» (рис. 3.107).

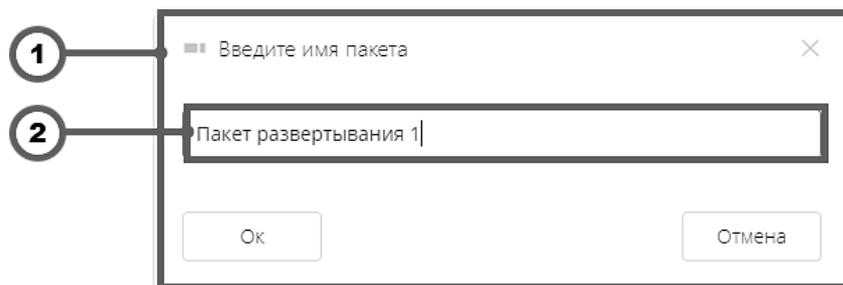


Рисунок 3.107 Окно создания пакета развертывания

1. **Форма** – предназначена для ввода полей, необходимых для создания пакета
2. **Поле ввода** – предназначено для ввода имени нового пакета

В результате появится логическое окно пакета развертывания без структуры. Формирование структуры пакета описано в разделе 3.16.2. Нажатие кнопки «Сохранить» приведет к сохранению пустого пакета, структуру которого можно будет сформировать позднее.

### 3.16.2 Формирование структуры пакета развертывания

Для формирования структуры пакета развертывания откройте логическое окно пакета и перейдите на вкладку **«Структура»** (рис. 3.108). Используйте кнопку **«Загрузить файлы»** для добавления файлов в состав пакета. Файлы в пакете можно организовать в папочную структуру. Для добавления папки используйте кнопку **«Новая папка»**. Имеется также возможность добавить в пакет директорию с файлами с помощью кнопки **«Загрузить директорию»**.

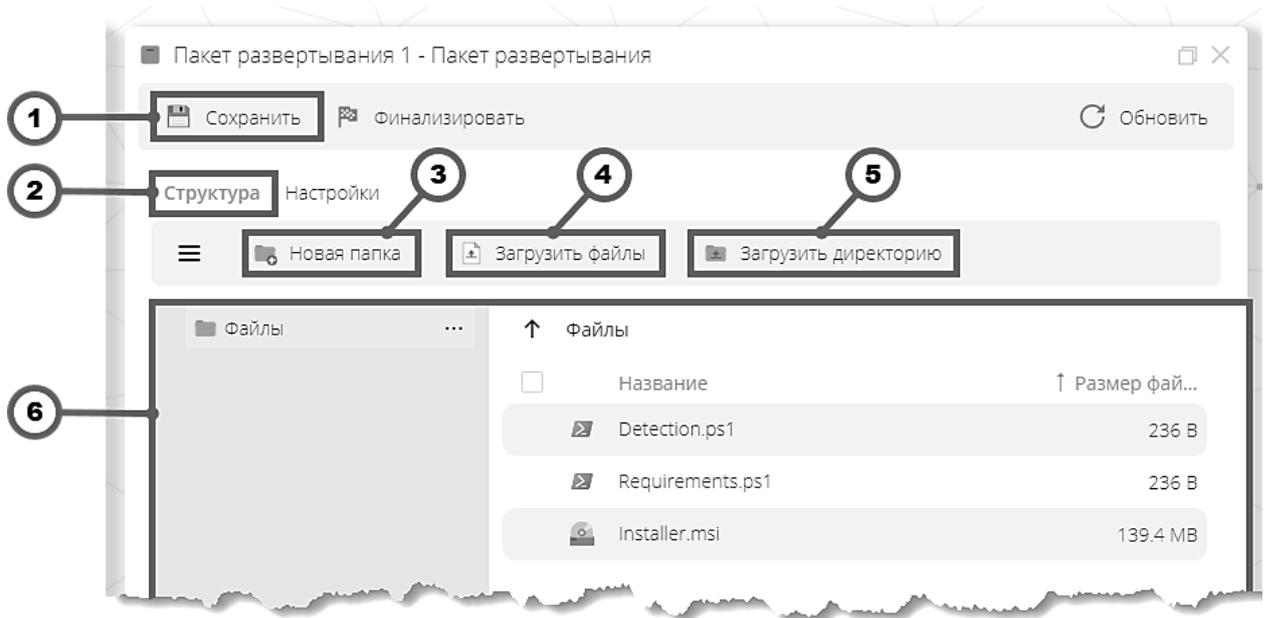


Рисунок 3.108 Формирование структуры пакета развертывания

1. Кнопка **«Сохранить»** – сохраняет все внесенные в окне пакета изменения.
2. Вкладка **«Структура»** – содержит элементы пользовательского интерфейса для управления структурой пакета развертывания.
3. Кнопка **«Новая папка»** – предназначена для создания папок в составе папочной структуры пакета.
4. Кнопка **«Загрузить файлы»** – предназначена для добавления файлов в состав пакета.
5. Кнопка **«Загрузить директорию»** – предназначена для директории с файлами в состав пакета.
6. Файловый менеджер – предназначен для просмотра и выбора элементов **файлово-папочной структуры** пакета. В нашем примере пакет состоит из трех файлов – Installer.msi, Detection.ps1, Requirements.ps1.

### 3.16.3 Настройка инсталляции пакета

Для настройки процесса инсталляции пакета развертывания откройте логическое окно пакета и перейдите на вкладку **«Настройки»** (рис. 3.109). В поле **«Команда инсталляции»** введите команду, которую ОС управляемого компьютера выполнит после доставки пакета на управляемый компьютер.

Управление доставкой пакетов развертывания осуществляется посредством **элемента директивы типа развертывание** (раздел 3.4.3)

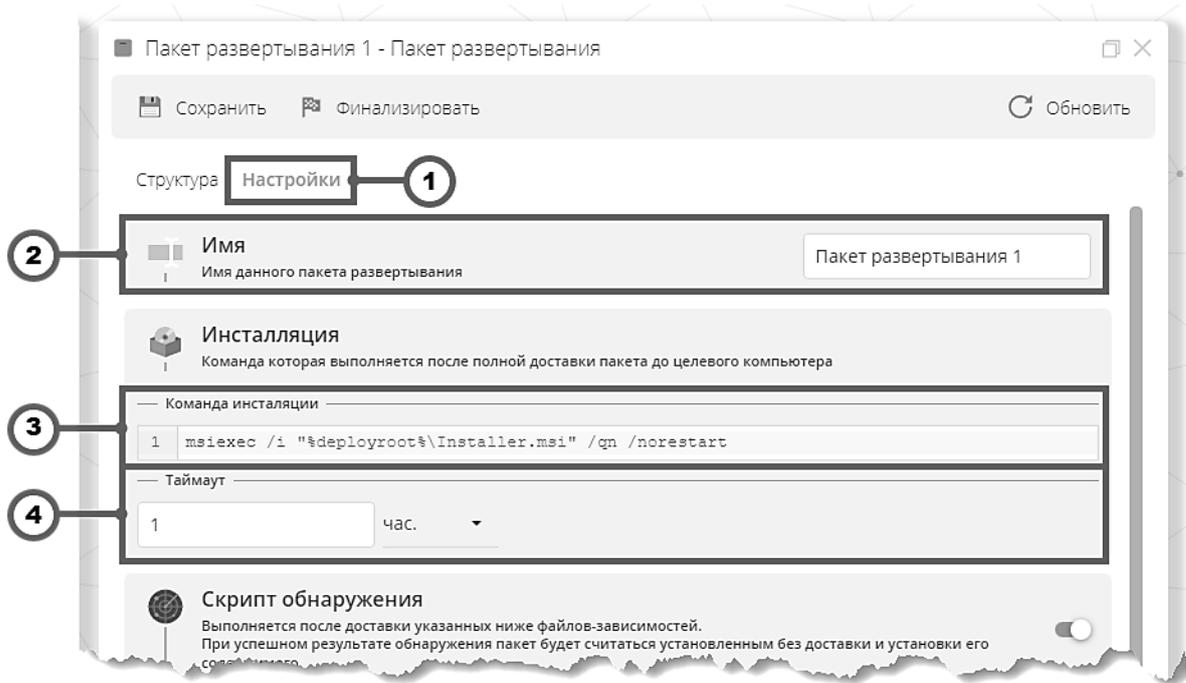


Рисунок 3.109 Настройка процесса инсталляции пакета развертывания

1. Вкладка **«Структура»** – содержит элементы пользовательского интерфейса для управления структурой пакета развертывания.
2. Поле **«Имя»** – предназначено для редактирования имени пакета развертывания.
3. Поле **«Команда инсталляции»** – задает команду ОС, которая будет выполнена для развертывания пакета на управляемом компьютере.
4. Поле **«Таймаут»** – предназначено для ограничения времени работы команды инсталляции.

Если для запуска процесса инсталляции должен использоваться один из **исполняемых файлов**, входящих в пакет развертывания, есть возможность быстро назначить **команду инсталляции** из контекстного меню файлового менеджера пакета на вкладке **«Структура»**, как показано на рисунке 3.110.

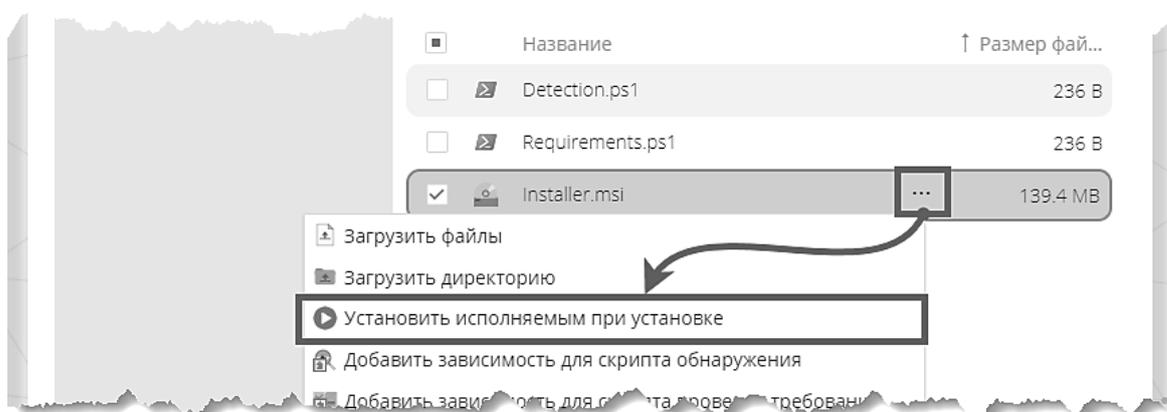


Рисунок 3.110 Установка файла пакета в качестве исполняемого

Для популярных типов исполняемых файлов, таких как .msi, .ps1, .bat, .cmd, .rpm, .deb, .sh, также будет сформирован типовой набор аргументов командной строки.

### 3.16.4 Настройка обнаружения установленного пакета

В целях экономии сетевых и вычислительных ресурсов рекомендуется настроить механизм **обнаружения** пакета развертывания на управляемом компьютере. С этой целью на вкладке «**Настройки**» в секции «**Скрипт обнаружения**» задайте **команду обнаружения** и включите ее выполнение, как показано на рисунке 3.111. Также добавьте необходимые для исполнения команды **файлы-зависимости** из состава пакета.

При успешном выполнении команды обнаружения на управляемом компьютере пакет будет считаться **установленным**, вследствие чего доставка и установка содержимого пакета проводиться не будет.

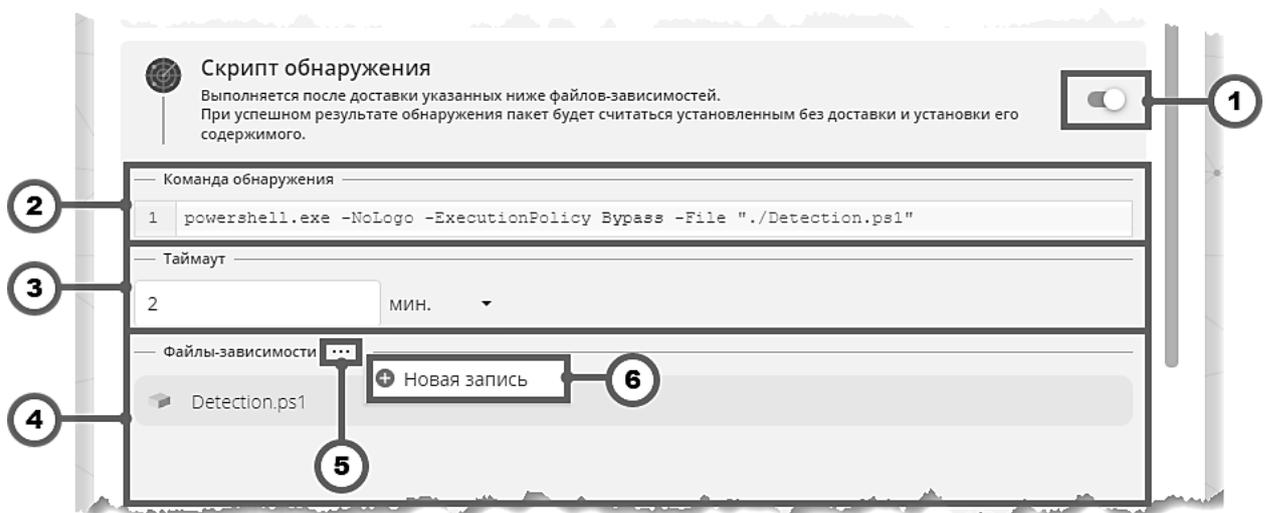


Рисунок 3.111 Настройка обнаружения установленного пакета

1. Переключатель **режима** команды обнаружения – включено / выключено.
2. Поле «**Команда обнаружения**» – задает команду ОС, которая будет выполнена для обнаружения установленного пакета на управляемом компьютере.
3. Поле «**Таймаут**» – предназначено для ограничения времени работы команды обнаружения.
4. Панель **файлов-зависимостей** – предназначена для управления списком файлов из состава пакета, которые должны быть доставлены на управляемый компьютер перед запуском команды обнаружения.
5. Кнопка **вызова контекстного меню** панели файлов-зависимостей.
6. **Контекстное меню** панели файлов-зависимостей.

Если для команды обнаружения должен использоваться один из **исполняемых файлов**, входящих в пакет развертывания, есть возможность быстро назначить **команду обнаружения** из контекстного меню файлового менеджера пакета на вкладке «**Структура**», как показано на рисунке 3.112.

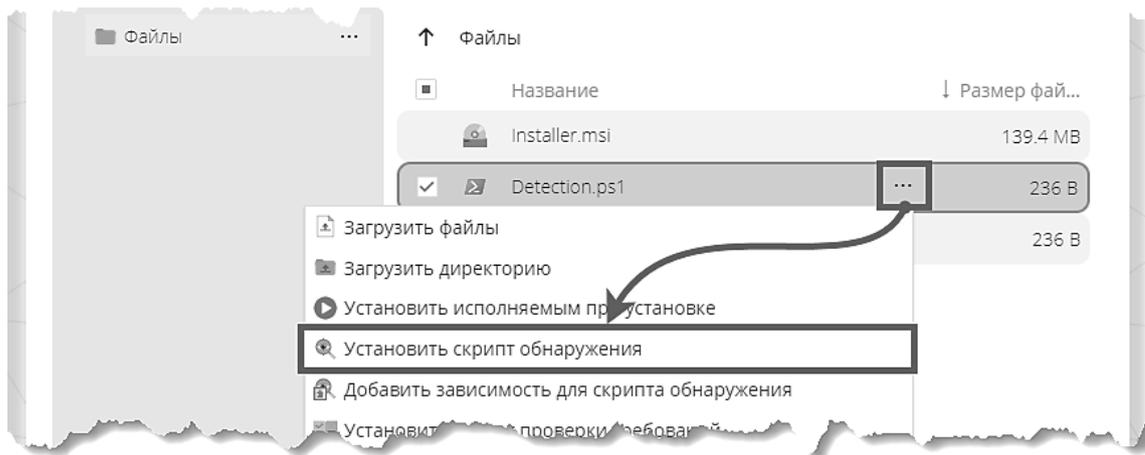


Рисунок 3.112 Установка файла пакета в качестве скрипта обнаружения

При установке **скрипта обнаружения** из контекстного меню для популярных типов исполняемых файлов, таких как .msi, .ps1, .bat, .cmd, .rpm, .deb, .sh, автоматически также будет сформирован типовой набор аргументов командной строки (рис. 3.111).

---

 **Совет**

Добавление файлов-зависимостей скрипта обнаружения также можно провести из контекстного меню файлового менеджера пакета на вкладке **«Структура»** с помощью команды **«Добавить зависимость для скрипта обнаружения»**.

---

### 3.16.5 Настройка проверки требований для пакета

В целях экономии сетевых и вычислительных ресурсов рекомендуется настроить механизм **проверки требований** для пакета развертывания на управляемом компьютере. С этой целью на вкладке **«Настройки»** в секции **«Скрипт проверки требований»** задайте **команду проверки требований** и включите ее выполнение, как показано на рисунке 3.113. Также добавьте необходимые для исполнения команды **файлы-зависимости** из состава пакета.

При успешном выполнении команды проверки требований на управляемом компьютере пакет будет считаться **разрешенным** для доставки и установки его содержимого.

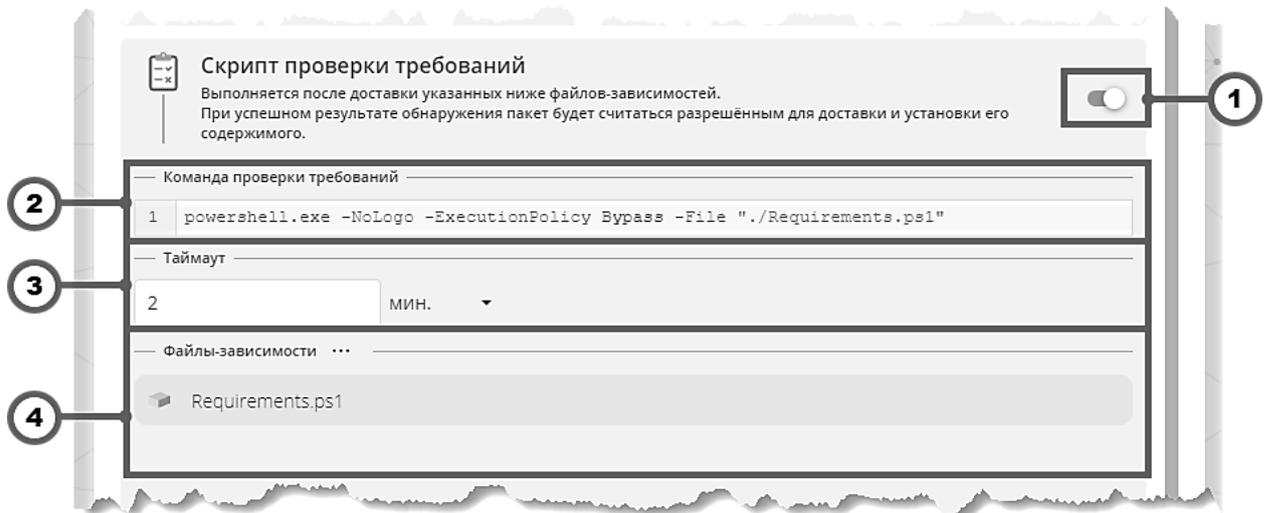


Рисунок 3.113 Настройка проверки требований для пакета

1. Переключатель **режима** команды проверки требований – включено / выключено.
2. Поле **«Команда проверки требований»** – задает команду ОС, которая будет выполнена для определения разрешения для установки пакета на управляемом компьютере.
3. Поле **«Таймаут»** – предназначено для ограничения времени работы команды проверки требований.
4. Панель **файлов-зависимостей** – предназначена для управления списком файлов из состава пакета, которые должны быть доставлены на управляемый компьютер перед запуском команды проверки требований.

Если для команды проверки требований должен использоваться один из **исполняемых файлов**, входящих в пакет развертывания, есть возможность быстро назначить **команду проверки требований** из контекстного меню файлового менеджера пакета на вкладке **«Структура»**, как показано на рисунке 3.114.

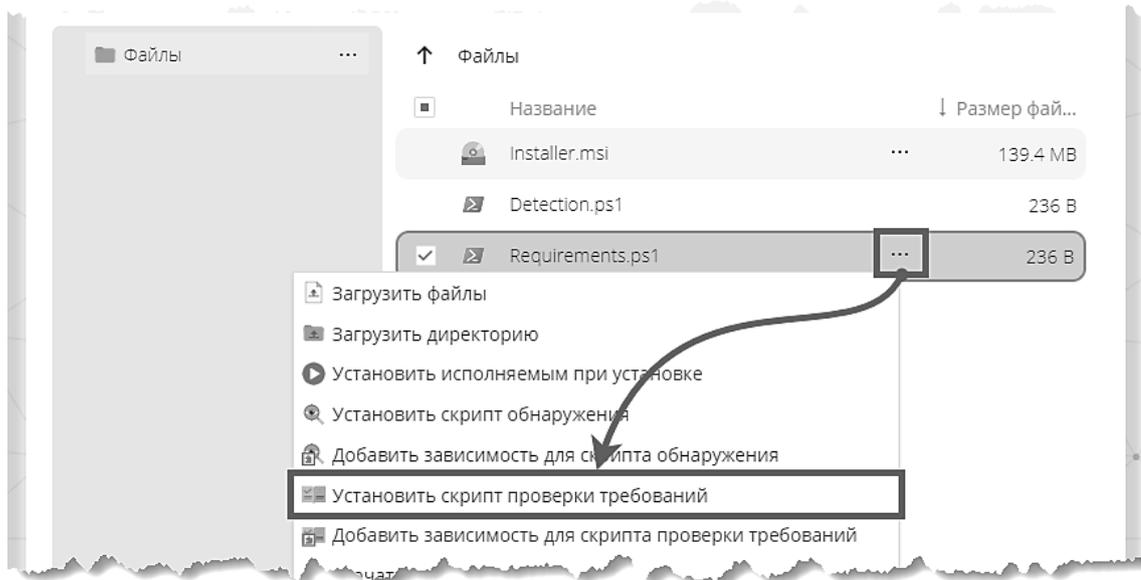


Рисунок 3.114 Установка файла пакета в качестве скрипта проверки требований

При установке **скрипта проверки требований** из контекстного меню для популярных типов исполняемых файлов, таких как .msi, .ps1, .bat, .cmd, .rpm, .deb, .sh, автоматически также будет сформирован типовой набор аргументов командной строки (рис. 3.113).

### Совет

Добавление файлов-зависимостей скрипта проверки требований также можно провести из контекстного меню файлового менеджера пакета на вкладке **«Структура»** с помощью команды **«Добавить зависимость для скрипта проверки требований»**.

### 3.16.6 Финализация пакета развертывания

Для использования пакета развертывания в **директиве развертывания** пакет должен быть **финализирован**, как показано на рисунке 3.115.

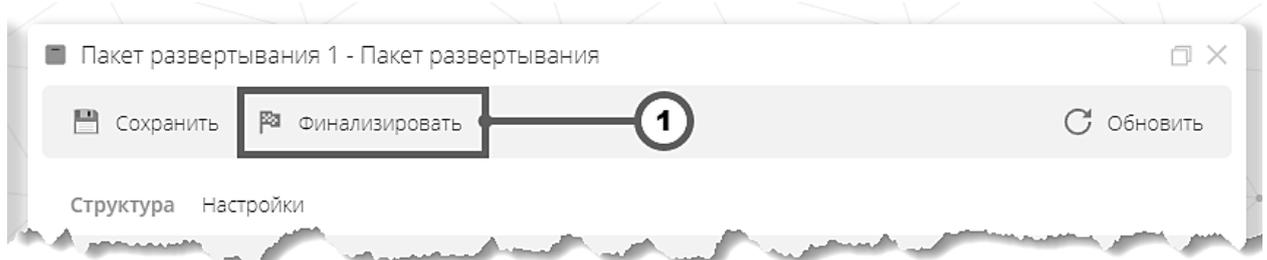


Рисунок 3.115 Финализация пакета развертывания

1. Кнопка **«Финализировать»** – финализирует пакет развертывания.

### Осторожно

Финализированные пакеты развертывания **недоступны для изменения**.

### 3.17 Каталог ПО

Каталог ПО предназначен для интерактивного просмотра результатов инвентаризации программного обеспечения. Для настройки инвентаризации ПО воспользуйтесь элементом директивы «Инвентаризация ПО», описанном в разделе 3.4.10.

Внешний вид основного окна режима «Каталог ПО» приведен на рисунке 3.116:

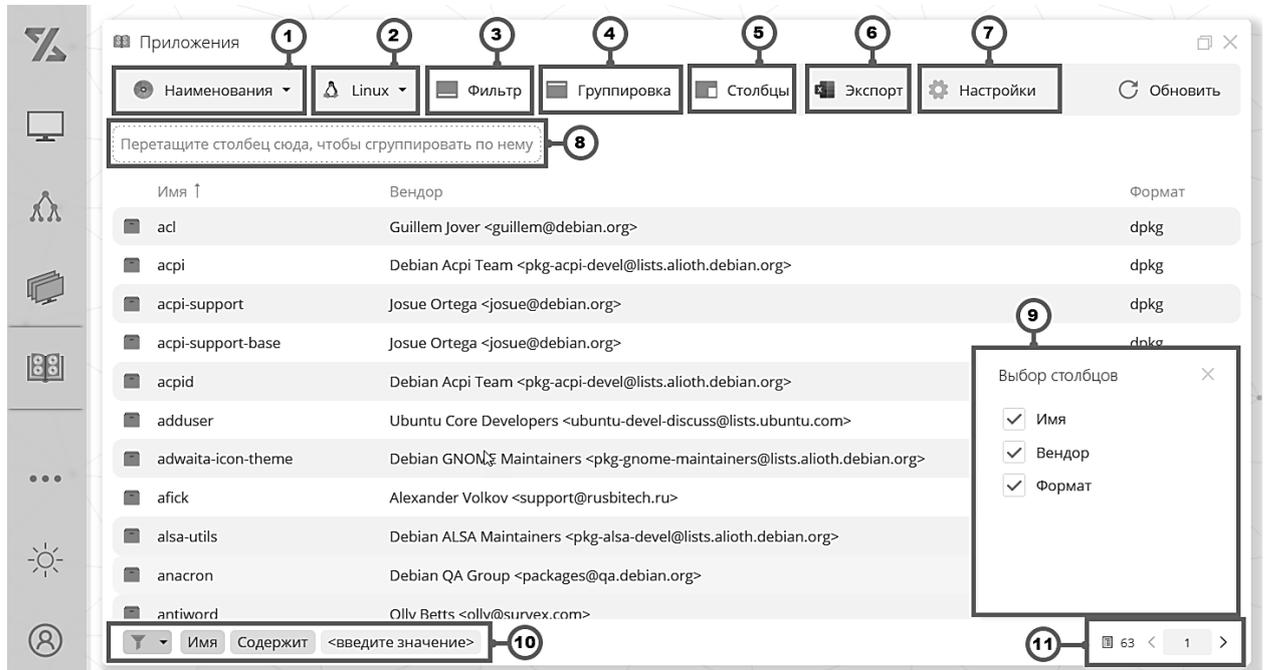


Рисунок 3.116 Основное окно режима «Каталог ПО»

1. Кнопка **выбора подробности** - предназначена для переключения подробности отображаемых табличных данных. Поддерживаются следующие уровни подробности:
  - a. «Наименования» - отображаются только уникальные наименования ПО, обнаруженные по результатам инвентаризации. В отображении не участвуют версии и компьютеры, на которых было обнаружено ПО;
  - b. «Версии» - отображаются уникальные наименования и версии ПО. В отображении не участвуют компьютеры, на которых было обнаружено ПО;
  - c. «Инсталляции» - отображаются данные инвентаризации по всем компьютерам. Наиболее подробный уровень.
2. Кнопка выбора семейства ОС – предназначена для переключения отображения результатов инвентаризации ПО для компьютеров под управлением ОС семейства Windows и ОС семейства Linux.
3. Кнопка **«Группировка»** – предназначена для включения/выключения видимости области группировки, находящейся над таблицей с данными (8).
4. Кнопка **«Фильтр»** - предназначена для включения/выключения видимости конструктора фильтра, находящегося под таблицей с данными (10).
5. Кнопка **«Столбцы»** – предназначена для включения/выключения видимости окна выбора столбцов (9).

6. Кнопка **«Экспорт»** - предназначена для сохранения данных в файле формата OpenXML Spreadsheet (\*.xlsx).
7. Кнопка **«Настройки»** - предназначена для открытия окна настроек каталога ПО. Работа с данным окном описана в разделе 3.17.2.
8. Область **группировки** – перетаскивание заголовка столбца из таблицы на эту область приводит к группировке таблицы по этому столбцу.
9. Окно **«Выбор столбцов»** - предназначено для включения/выключения видимости столбцов в таблице.
10. **Конструктор фильтра** - предназначен для настройки визуализации таблицы результатов инвентаризации ПО. Использование конструктора фильтра описано в разделе 2.2.4.
11. **Пейджер** – предназначен для переключения между страницами табличного представления и отображения номера текущей страницы.

Основные сценарии работы с основным окном режима «Каталог ПО» состоят:

- в выборе требуемых семейства ОС и подробности данных;
- фильтрации или группировке полученных данных (например, по имени ПО);
- экспорте полученного результата или переходу к карточке ПО (описана в разделе 3.17.1).

---

### **Примечание**

Уровни подробности **«Наименования»** и **«Версии»** отображают данные инвентаризации, собираемые по накопительному принципу. Таким образом, часть наименований ПО и их версий, отображаемых на данных уровнях подробности, могут не присутствовать ни на одном компьютере в последних поступивших данных инвентаризации ПО.

Каталог ПО выполняет автоматическую очистку устаревших наименований и версий ПО на регулярной основе. Период актуальности данных, в течение которого наименования и версии, более не представленные ни на одном компьютере, не удаляются из представлений **«Наименования»** и **«Версии»**, регулируется соответствующей настройкой, описанной в разделе 3.17.2.

---

### **Совет**

Для работы с максимально актуальными данными (по результатам последних поступивших результатов инвентаризации ПО с каждого компьютера, не включающих устаревшие наименования и версии) воспользуйтесь уровнем подробности **«Инсталляции»** или режимом **«Отчеты ПО»**.

---

### 3.17.1 Карточка ПО

Карточка программного обеспечения предназначена для просмотра расширенной информации по конкретному наименованию ПО, включая все его обнаруженные версии.

Первоначальный вид карточки ПО для Linux-пакета *apt* приведен на рисунке 3.117:

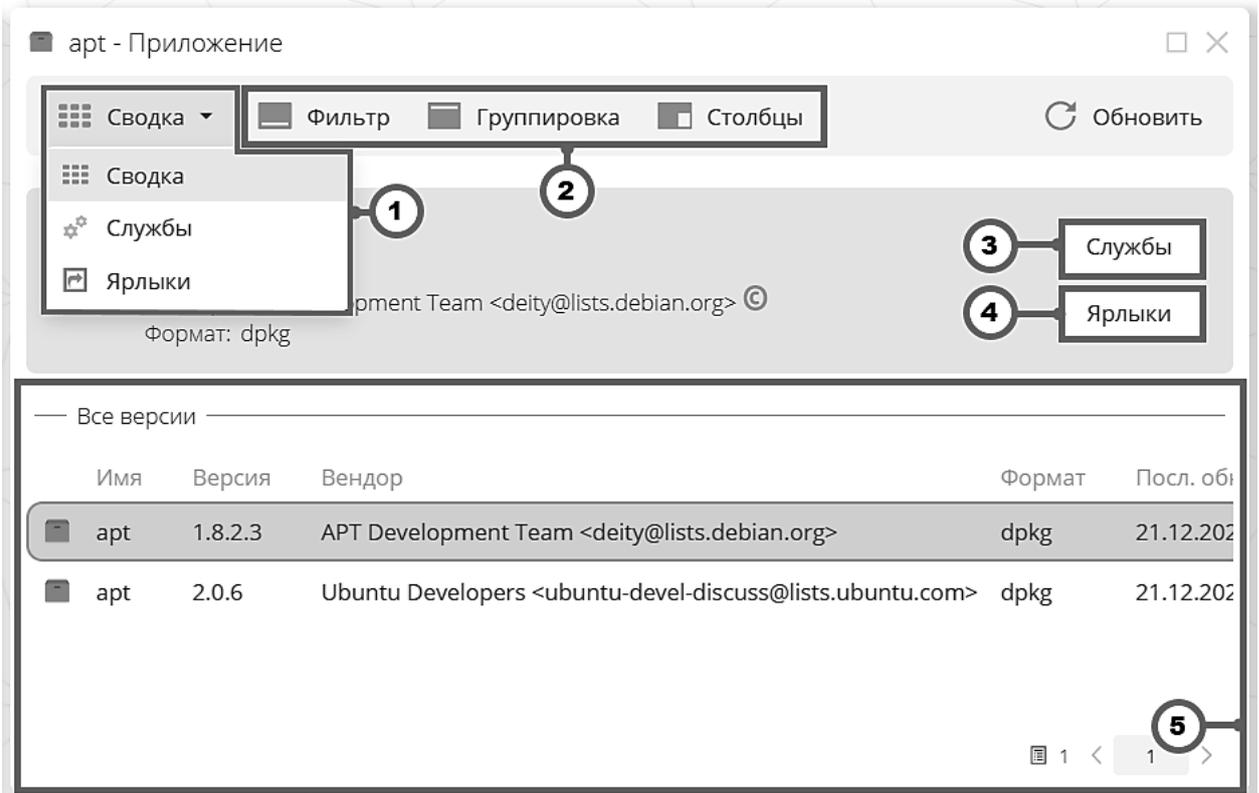


Рисунок 3.117 Представление «Сводка» карточки ПО Linux-пакета *apt*

1. Кнопка **выбора представления** - предназначена для переключения данных, отображаемых в области данных (5):
  - а. «Сводка» - отображаются все обнаруженные версии ПО;
  - б. «Службы» - отображаются службы, установленные в рамках инсталляционных процедур конкретной версии ПО;
  - в. «Ярлыки» - отображаются ярлыки (\*.lnk и \*.desktop файлы), установленные в рамках инсталляционных процедур конкретной версии ПО.
2. Кнопки управления данными – предназначены для типовых операций по фильтрации, группировки и регулированию состава столбцов в области данных (5).
3. Кнопка **«Службы»** – предназначена для быстрого перехода к представлению «Службы». Эквивалентна выбору представления «Службы» при помощи кнопки выбора представления (1).
4. Кнопка **«Ярлыки»** - предназначена для быстрого перехода к представлению «Ярлыки». Эквивалентна выбору представления «Ярлыки» при помощи кнопки выбора представления (1).
5. Область данных – предназначена для отображения данных соответствующих выбранному представлению.

## Примечание

Область данных в представлении **«Сводка»** отображает данные инвентаризации, собираемые по накопительному принципу. Таким образом, часть версий ПО, отображаемых в данной области, могут не присутствовать ни на одном компьютере в последних поступивших данных инвентаризации ПО.

Каталог ПО выполняет автоматическую очистку устаревших наименований и версий ПО на регулярной основе. Период актуальности данных, в течение которого наименования и версии, более не представленные ни на одном компьютере, не удаляются из представлений **«Наименования»** и **«Версии»**, регулируется соответствующей настройкой, описанной в разделе 3.17.2.

## Совет

Обратите внимание на колонку **«Посл. обнаружение»** области данных в представлении **«Сводка»**. Данная колонка отображает дату и время последнего обнаружения соответствующей версии в поступивших результатах инвентаризации ПО.

Используйте данную колонку для оценки актуальности присутствия данной версии на управляемых компьютерах.

Внешний вид представления сводка для Windows-приложения *7-Zip* приведен на рисунке 3.118:

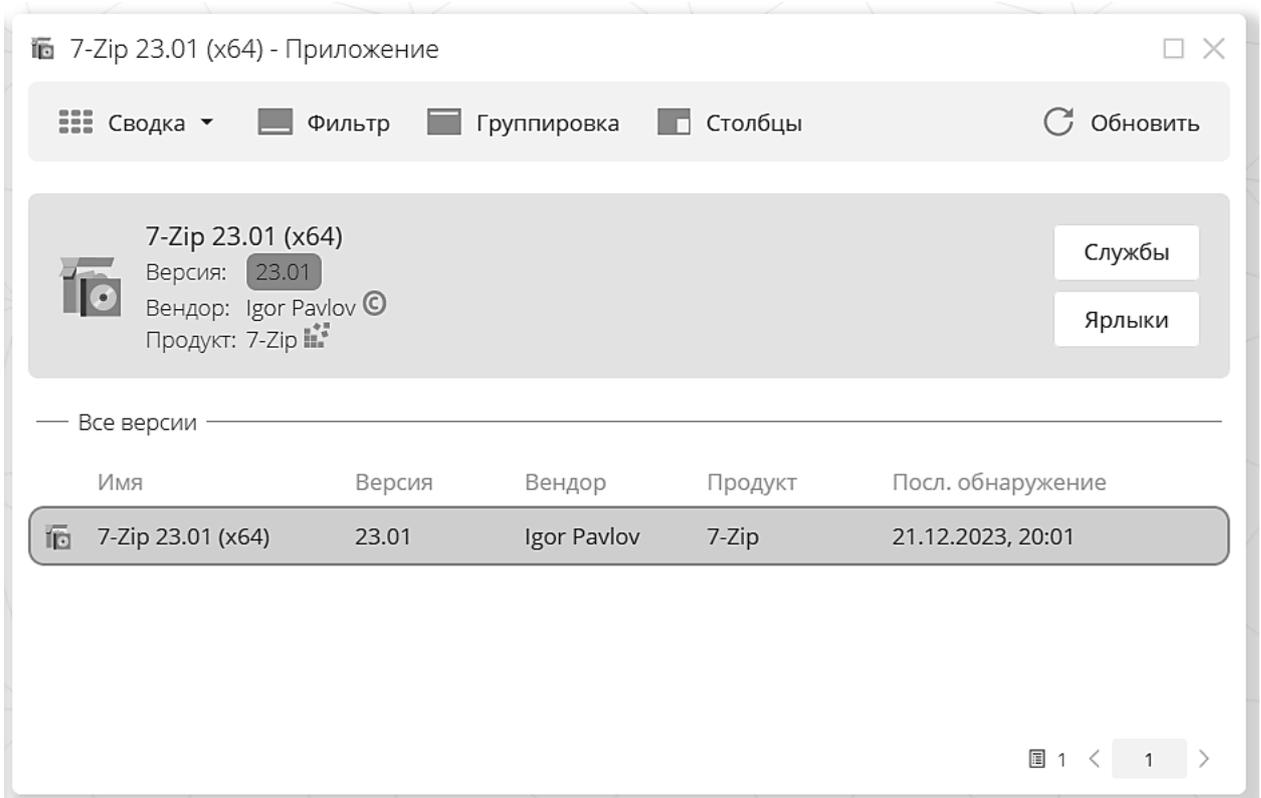


Рисунок 3.118 Представление «Сводка» карточки ПО Windows-приложения 7-Zip

Состав столбцов области данных представления «Сводка» карточки ПО для Windows-приложений и Linux-пакетов отличается. Отличия и описание каждого столбца приведены в таблице 3.3:

Столбец			Описание
Имя	+	+	Отображаемое имя ПО для ОС семейства Windows. Соответствует параметру <code>DisplayName</code> ключа реестра для ПО. Имя пакета для ОС семейства Linux.
Версия	+	+	Отображаемое имя ПО для ОС семейства Windows. Соответствует параметру <code>DisplayVersion</code> ключа реестра для ПО. Версия пакета для ОС семейства Linux.
Вендор	+	+	Производитель ПО для ОС семейства Windows. Соответствует параметру <code>Publisher</code> ключа реестра для ПО. Атрибут <code>maintainer</code> для *.deb пакетов. Атрибут <code>vendor</code> для *.rpm пакетов.
Формат	-	+	Формат пакета: <b>dpkg</b> или <b>rpm</b> .
Продукт	+	-	Имя ключа реестра ПО для ОС семейства Windows. Для инсталляторов ПО в формате *.msi соответствует коду продукта MSI. Для других форматов инсталляторов – произвольная строка.
Посл. обнаружение	+	+	Дата и время последнего обнаружения записи о версии ПО в результатах инвентаризации ПО, поступивших с управляемых компьютеров.

Таблица 3.3. Столбцы представления «Сводка» карточки ПО



### Совет

Обратите внимание, что при переключении представления в значения «**Службы**» и «**Ярлыки**», соответствующие данные по обнаруженным службам и ярлыкам отображаются для версии ПО, выбранной в представлении «**Сводка**».

Внешний вид представления «Службы» карточки ПО для Linux-пакета *ntp* приведен на рисунке 3.119:

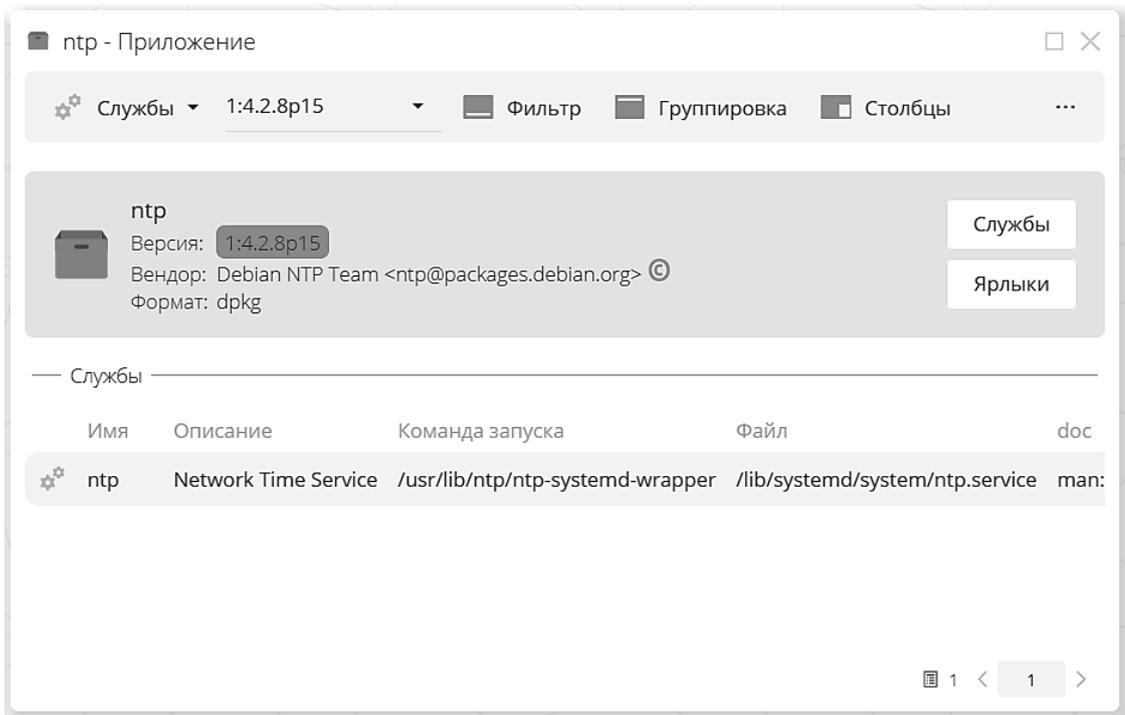


Рисунок 3.119 Представление «Службы» карточки ПО Linux-пакета *ntp*

Внешний вид представления «Службы» карточки ПО для Windows-приложения *PostgreSQL 14* приведен на рисунке 3.120:

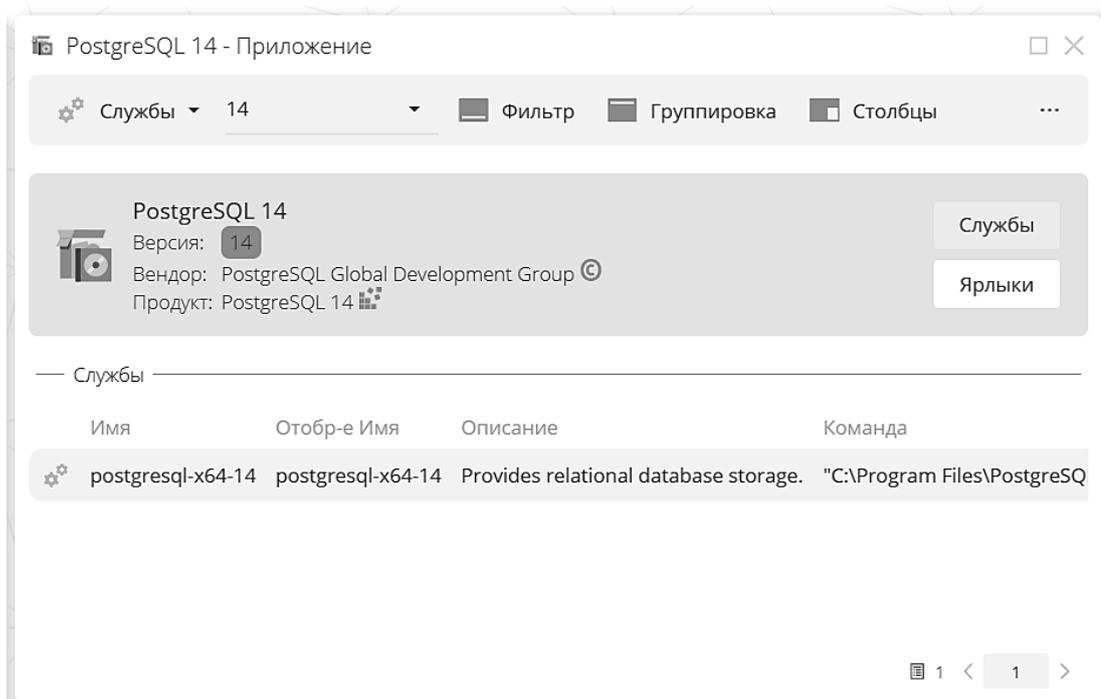


Рисунок 3.120 Представление «Службы» карточки ПО Windows-приложения *PostgreSQL 14*

Состав столбцов области данных представления «Службы» карточки ПО для Windows-приложений и Linux-пакетов отличается. Отличия и описание каждого столбца приведены в таблице 3.4:

Столбец			Описание
Имя	+	+	Системное имя службы. Соответствует имени Windows-службы для ОС семейства Windows. Соответствует имени *.service-файла службы для ОС семейства Linux.
Отобр-е имя	+	-	Отображаемое имя службы. Соответствует отображаемому имени Windows-службы для ОС семейства Windows.
Описание	+	+	Описание службы. Соответствует описанию Windows-службы для ОС семейства Windows. Соответствует параметру <code>Description</code> секции <code>[Unit] *.service-</code> файла службы для ОС семейства Linux.
Команда (запуска)	+	+	Команда запуска службы. Соответствует команде запуска Windows-службы для ОС семейства Windows. Соответствует параметру <code>ExecStart</code> секции <code>[Service] *.service-</code> файла службы для ОС семейства Linux.
Файл	-	+	Полный путь до *.service-файла службы для ОС семейства Linux.
doc	-	+	Соответствует параметру <code>Documentation</code> секции <code>[Unit] *.service-</code> файла службы для ОС семейства Linux.

Таблица 3.4. Столбцы представления «Службы» карточки ПО

### Примечание

Сбор информации о службах, соответствующих Windows-приложениям и Linux-пакетам осуществляется для каждого семейства ОС по эвристическому алгоритму.

Ни одно из семейств ОС не предоставляет гарантированной информации о связи между службами и установленными приложениями/пакетами.

Поэтому информация в представлении «Службы» карточки ПО носит вспомогательный характер и для ряда видов ПО может отсутствовать.

Внешний вид представления «Ярлыки» карточки ПО для Linux-пакета *firefox* приведен на рисунке 3.121:

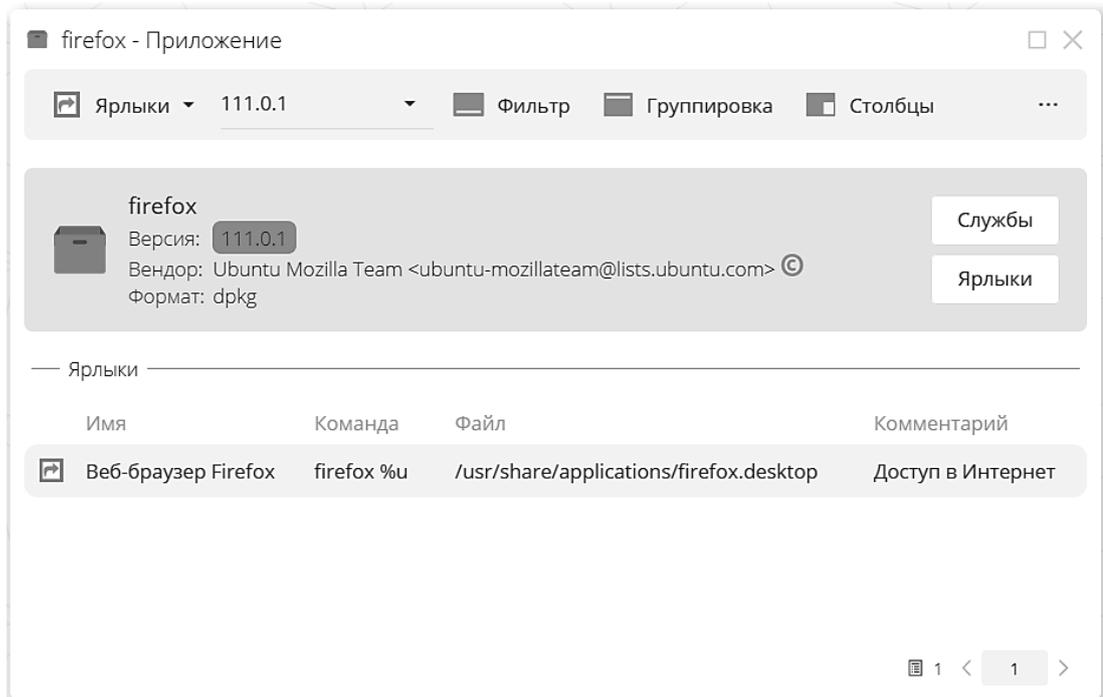


Рисунок 3.121 Представление «Ярлыки» карточки ПО Linux-пакета *firefox*

Внешний вид представления «Ярлыки» карточки ПО для Windows-приложения *Firefox* приведен на рисунке 3.122:

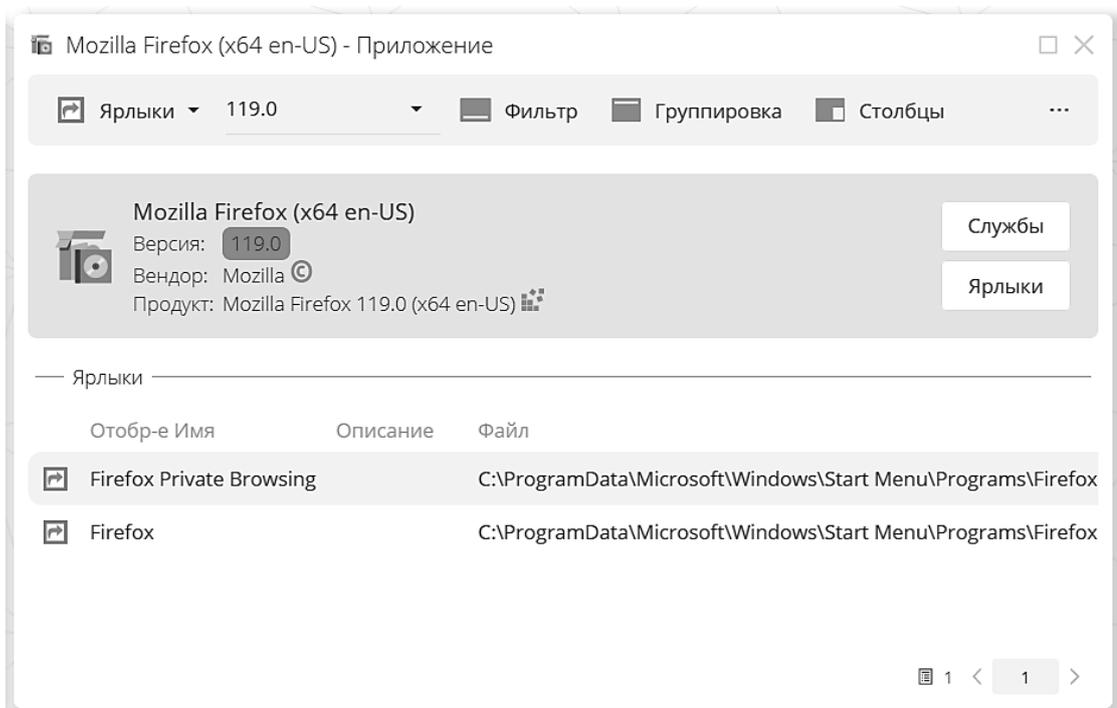


Рисунок 3.122 Представление «Ярлыки» карточки ПО Windows-приложения *Firefox*

Состав столбцов области данных представления «Ярлыки» карточки ПО для Windows-приложений и Linux-пакетов отличается. Отличия и описание каждого столбца приведены в таблице 3.5:

Столбец			Описание
Имя	-	+	Имя ярлыка. Соответствует параметру <code>Name [ru]</code> или <code>Name</code> секции <code>[Desktop Entry]</code> *.desktop-файла для ОС семейства Linux.
Отобр-е имя	+	-	Отображаемое имя ярлыка. Соответствует имени *.lnk-файла для ОС семейства Windows.
Описание	+	-	Описание ярлыка. Соответствует описанию *.lnk-файла для ОС семейства Windows.
Команда	+	+	Команда запуска ярлыка. Соответствует команде запуска *.lnk-файла для ОС семейства Windows. Соответствует параметру <code>Exec</code> секции <code>[Desktop Entry]</code> *.desktop-файла для ОС семейства Linux.
Файл	+	-	Полный путь до *.lnk-файла для ОС семейства Windows.
Комментарий	-	+	Комментарий ярлыка. Соответствует параметру <code>Comment [ru]</code> или <code>Comment</code> секции <code>[Desktop Entry]</code> *.desktop-файла для ОС семейства Linux.

Таблица 3.5. Столбцы представления «Ярлыки» карточки ПО

### Примечание

Сбор информации о ярлыках, соответствующих Windows-приложениям и Linux-пакетам осуществляется для каждого семейства ОС по эвристическому алгоритму.

Ни одно из семейств ОС не предоставляет гарантированной информации о связи между ярлыками и установленными приложениями/пакетами.

Поэтому информация в представлении «Ярлыки» карточки ПО носит вспомогательный характер и для ряда видов ПО может отсутствовать.

### 3.17.2 Настройки каталога ПО

Окно настроек каталога ПО предназначено для модификации настроек алгоритма автоматической очистки устаревших наименований и версий ПО, а также для выполнения ручной очистки всего каталога.

Внешний вид окна настроек каталога ПО приведен на рисунке 3.123:

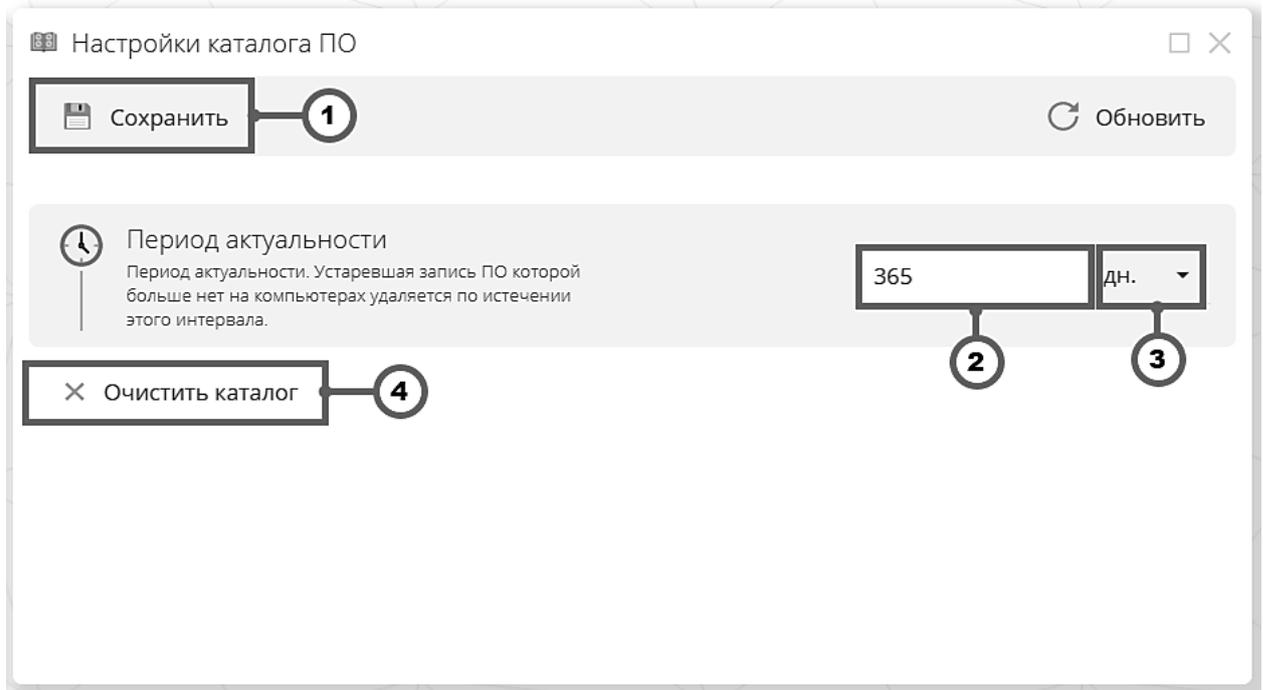


Рисунок 3.123 Окно настроек каталога ПО

1. Кнопка **«Сохранить»** - предназначена для сохранения изменений значения периода актуальности (2-3).
2. **Поле числового ввода** – предназначено для ввода числового значения периода актуальности.
3. **Переключатель единиц измерения** – предназначен для переключения единиц измерения значения в поле числового ввода (2).
4. Кнопка **«Очистить каталог»** - предназначена выполнения **полной** очистки данных в каталоге ПО.

#### **!** Осторожно

Нажатие на кнопку «Очистить каталог» приведет к полному удалению всех накопленных данных. Повторное наполнение каталога будет происходить постепенно, по мере поступления новых данных инвентаризации ПО с управляемых компьютеров.

В большинстве случаев использование данной кнопки ограничивается пилотными и тестовыми средами.

### 3.18 Отчеты ПО

Режим «Отчеты ПО» предназначен для построения табличных иерархических отчетов по результатам инвентаризации программного обеспечения. Для настройки инвентаризации ПО воспользуйтесь элементом директивы «Инвентаризация ПО», описанном в разделе 3.4.10.

Внешний вид окна проводника по отчетам приведен на рисунке 3.124:

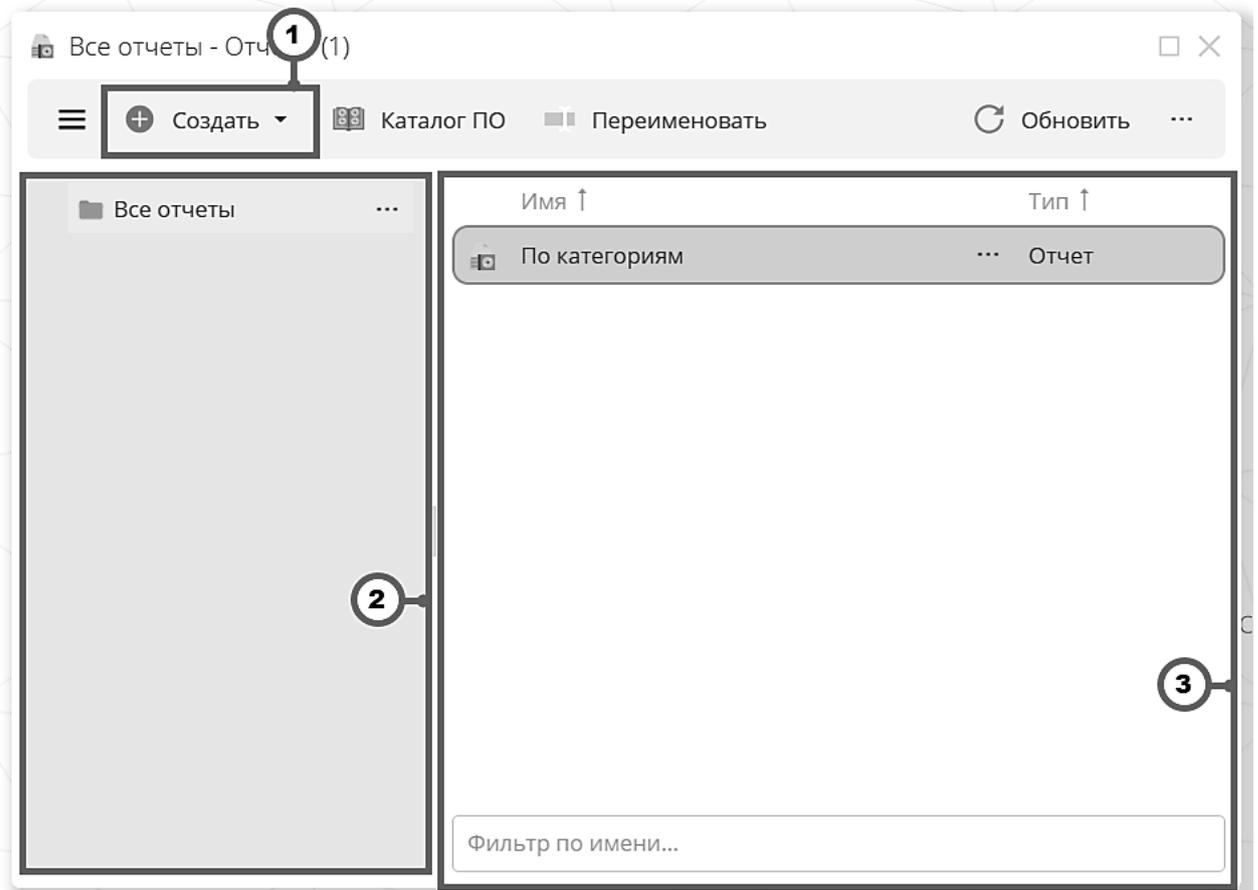


Рисунок 3.124 Окно проводника по отчетам

Основными элементами окна проводника по отчетам являются:

1. Кнопка **«Создать»** - предназначена для создания новых отчетов и папок в текущей просматриваемой папке.
2. **Область иерархии папок** – предназначена для навигации по папкам отчетов с учетом их вложенности.
3. **Область просмотра содержимого папки** – предназначена для визуализации и выбора отчетов и подпапок для текущей отображаемой папки.

Основными сценариями работы с окном проводника по отчетам являются:

- поиск требуемого существующего отчета или создание нового;
- редактирование/удаление/открытие существующего отчета.

## Совет

Для модификации, переименования, удаления существующего отчета воспользуйтесь контекстным меню соответствующего элемента области просмотра содержимого папки или пунктом инструментальной панели окна.

Для отображения дополнительных пунктов инструментальной панели окна воспользуйтесь кнопкой «...».

### 3.18.1 Создание и редактирование отчета

Создание и редактирование отчета производится при помощи единого **мастера редактирования отчета**. Внешний вид вкладки «**Правила**» окна мастера редактирования отчетов приведен на рисунке 3.125:

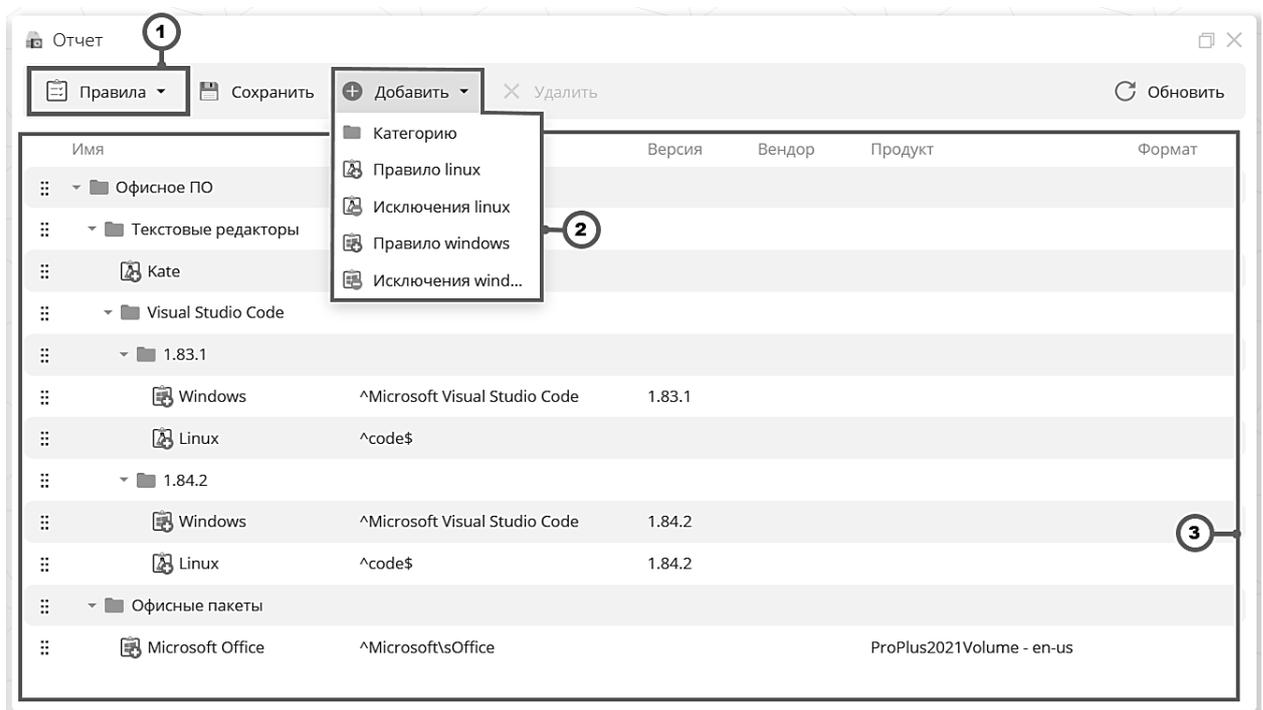


Рисунок 3.125 Вкладка «Правила» окна мастера редактирования отчетов

Основными элементами вкладки «Правила» являются:

1. **Переключатель вкладок** - предназначен для переключения между вкладками «Правила» и «Группы».
2. **Кнопка «Добавить»** – предназначена добавления в отчет категорий, правил и исключений.
3. **Область просмотра структуры отчета** – предназначена для визуализации и модификации иерархической структуры отчета.

Ключевыми понятиями отчетов ПО являются:

- **Категория** – предназначена для группировки правил и исключений. Для каждой категории при расчете отчета выполняется подсчет агрегатов (количество инсталляций, количество компьютеров). Используйте категории для получения в отчете дополнительных срезов по видам ПО.
- **Правило (Windows или Linux)** – предназначены для формулирования правил отбора программного обеспечения, участвующего в отчете.
- **Исключение (Windows или Linux)** – предназначены для формулирования правил исключения программного обеспечения, участвующего в отчете. Исключения применяются только к программному обеспечению, отобранному в соответствии с правилами, расположенными в той же категории, что и исключения.

---

### **Примечание**

Типичная структура отчета ПО приведена на рисунке 3.125. В рамках приведенной структуры организована корневая категория «Офисное ПО», состоящая из подкатегорий «Текстовые редакторы» и «Офисные пакеты». Соответствующие подкатегории в свою очередь наполнены правилами отбора интересующего ПО, как учитывающими версии, так и игнорирующими их.

При просмотре результатов расчета подобного отчета созданная структура позволит получить информацию как о количестве компьютеров, на которых установлен хотя бы один вид офисного ПО, так и путем дальнейшего раскрытия иерархии – разбиение по различным подкатегориям и конкретным видам ПО.

---

Для добавления категории выберите соответствующий элемент выпадающего меню кнопки «Добавить». Для создания категории требуется задать лишь ее имя.

Для добавления правила или исключения воспользуйтесь соответствующим элементом выпадающего меню кнопки «Добавить». Создание правил и исключений происходит в отдельном окне редактирования.

---

### **Примечание**

Несмотря на наличие четырех отдельных видов окон редактирования (правило Windows, правило Linux, исключение Windows, исключение Linux) принципы работы с данными окнами общие.

Различия между упомянутыми окнами состоят в перечне тестовых данных для проверки правила или исключения:

- для Windows- правил и исключений – это каталог Windows ПО,
- для Linux- правил и исключений – это каталог пакетов Linux,

а также в полях правил, на которые можно задать условия.

---

Внешний вид окна редактирования Windows-правила приведен на рисунке 3.126:

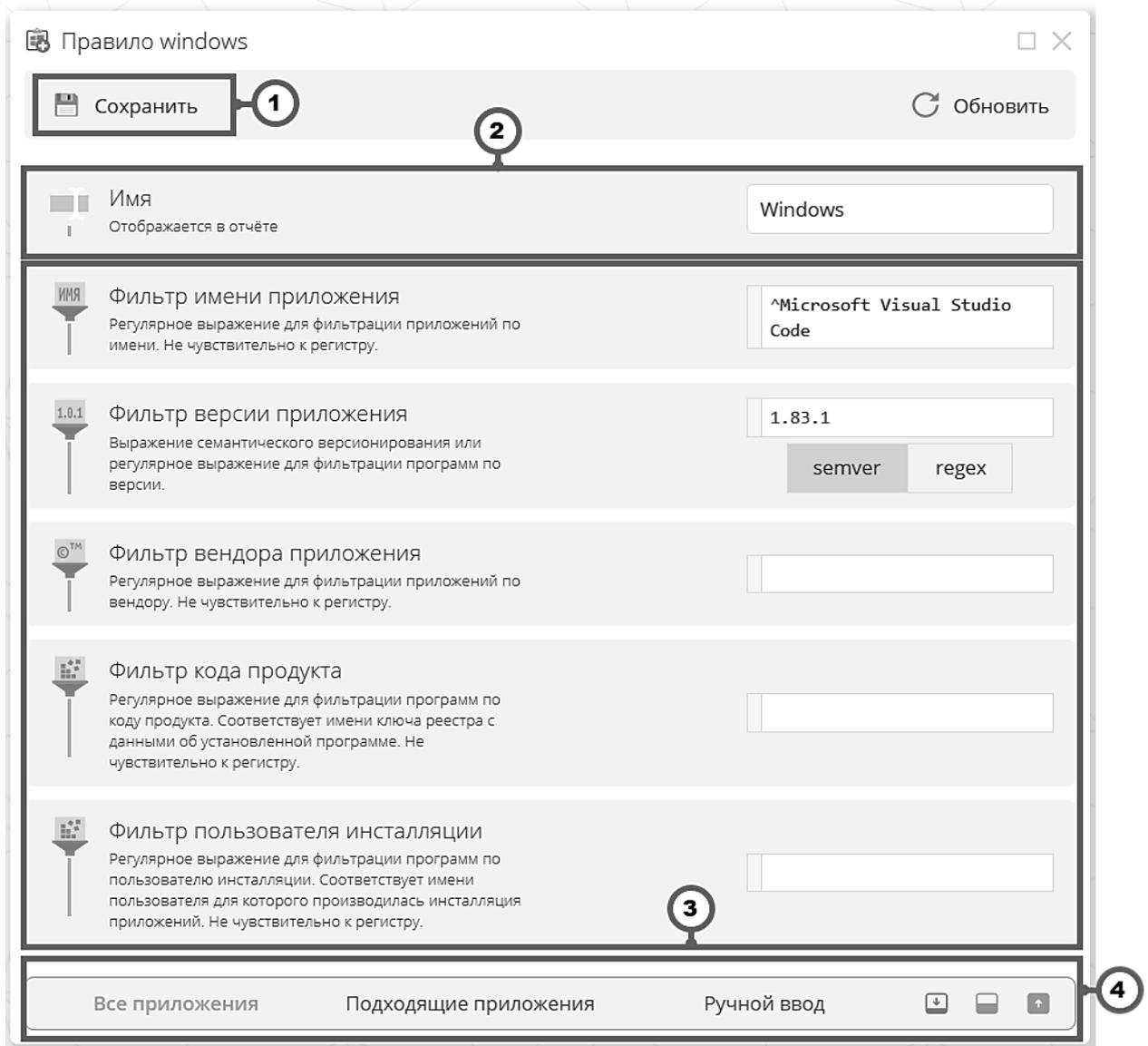


Рисунок 3.126 Окно редактирования Windows-правила отчета ПО

Основными элементами окна редактирования Windows-правила являются:

1. Кнопка **«Сохранить»** - предназначена для сохранения изменений, внесенных в правило.
2. **Редактор имени правила** – предназначен для задания имени правила, отображаемого в отчете.
3. **Редакторы фильтров** – предназначены для задания условий отбора программного обеспечения.
4. **Заголовок области тестирования** – предназначен для выбора режима отображения области тестирования правила (скрыта, 50% высоты окна, 100% высоты окна), а также выбора соответствующей вкладки тестирования («Все приложения», «Подходящие приложения», «Ручной ввод»).

Введите в редакторах фильтров выражения фильтрации, соответствующие желаемому отбору программного обеспечения. Отсутствие значения в редакторе того или иного фильтра соответствует отсутствию фильтрации по данному полю.

---

### Совет 1

Во всех редакторах фильтров поддерживается фильтрация при помощи **регулярных выражений** (англ. *Regular Expressions, Regex*). Для облегчения задачи написания подобных выражений все редакторы фильтра сопровождаются интерактивным полем ввода, поддерживающим подсветку синтаксиса и всплывающие подсказки.

Для получения информации об интерпретации того или иного символа с точки зрения синтаксиса регулярных выражений наведите курсор мыши на интересующий символ. Примеры всплывающих окон с подсказками приведены на рисунке 3.127:

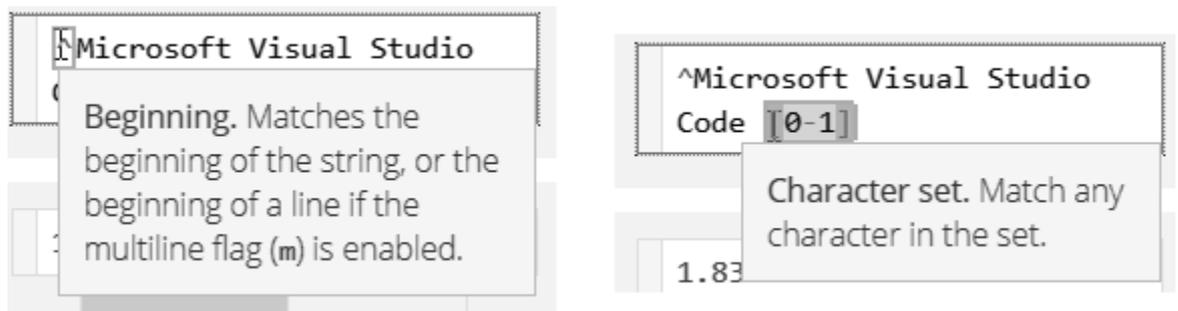


Рисунок 3.127 Примеры всплывающих подсказок редакторов регулярных выражений

---

### Совет 2

Наиболее типовым подходом является наложение фильтров на **имя и версию** приложений.

---

По окончании ввода выражений фильтрации на требуемые поля выполните проверку правила при помощи **области тестирования**. Для открытия области тестирования нажмите на название любой вкладки в ее заголовке (элемент 3 на рисунке 3.126).

Вкладка «**Все приложения**» области тестирования предназначена для оценки применимости наложенных фильтров для захвата интересующих приложений. Внешний вид вкладки «Все приложения» приведен на рисунке 3.128:

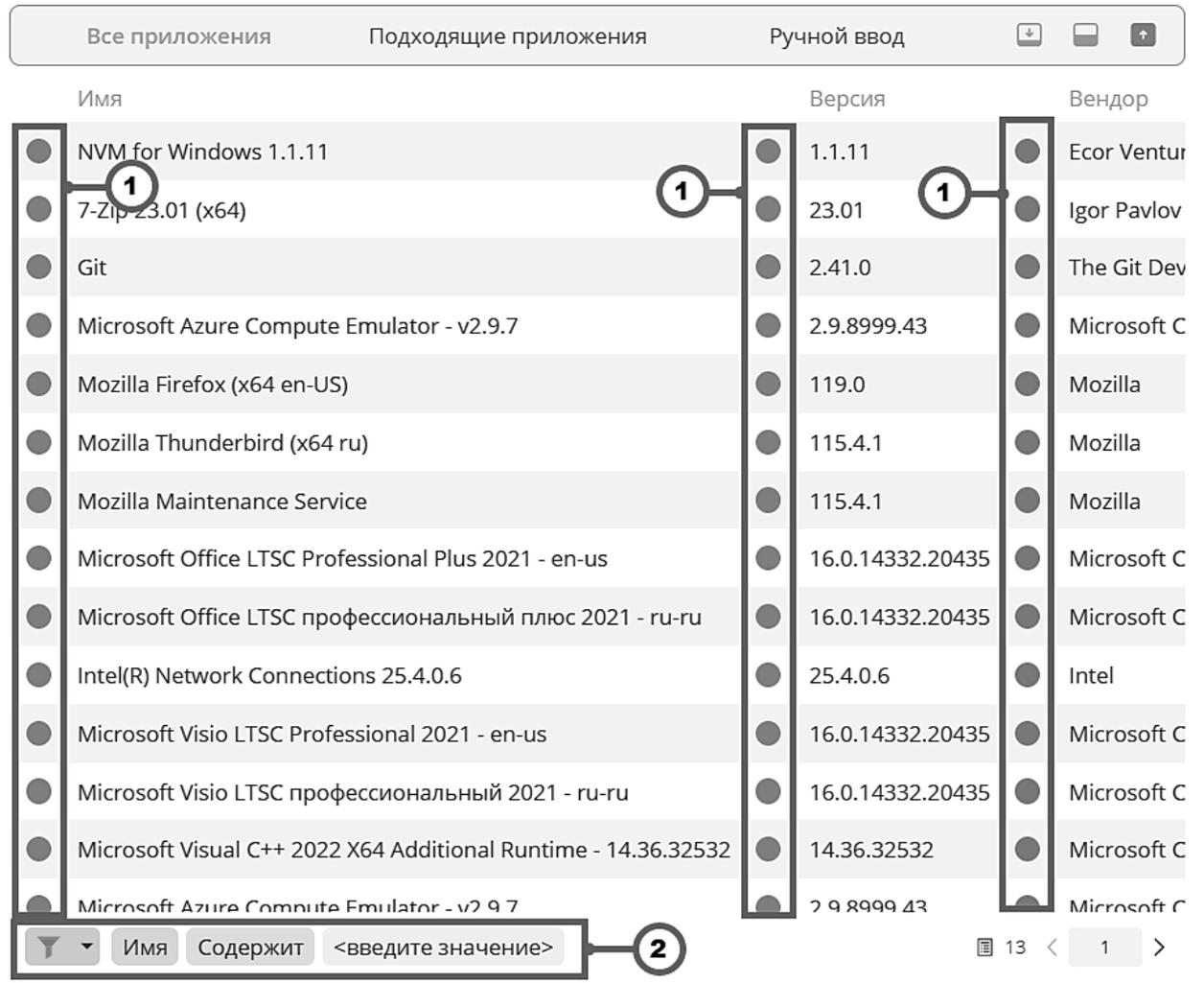


Рисунок 3.128 Вкладка «Все приложения» области тестирования редактора правила Windows отчета ПО

Список версий ПО, отображаемый в данной вкладке, сопровождается цветовым индикатором в каждой колонке (1):

- *серый* цвет свидетельствует о том, что значение в данной колонке **не соответствует** введенному выражению фильтрации;
- *зеленый* цвет свидетельствует о том, что значение в данной колонке **соответствует** введенному выражению фильтрации.

В перечень ПО, отбираемый правилом, входят только те строки, у которых индикатор соответствия окрашен зеленым цветом **во всех колонках**.

Пересчет индикаторов осуществляется автоматически в реальном времени при изменении выражений фильтрации.

## Совет

Разумеется, ручная оценка применимости правила путем проверки всех индикаторов для всех видов ПО, отображаемых в списке, является слишком трудоемкой, особенно при значительном многообразии видов ПО в результатах инвентаризации.

Использование вкладки «Все приложения» предполагается совместно с областью фильтрации каталога ПО (элемент 2 на рисунке 3.128).

Типовым сценарием использования данной вкладки является:

- отбор интересующего ПО при помощи области фильтрации;
- проверка, что цветовой индикатор окрашен в зеленый цвет для всех отображаемых результатов;
- отбор потенциально ошибочного ПО при помощи области фильтрации (например, с похожим названием или версией);
- проверка, что цветовой индикатор окрашен в серый цвет для всех отображаемых результатов;

Вкладка **«Подходящие приложения»** предназначена для визуализации только тех приложений и их версий, которые соответствуют правилу. Подобный расчет может занимать значительное время, поэтому для его выполнения необходимо нажать на кнопку «Рассчитать», отображаемую при первом переходе в данную вкладку.

Внешний вид вкладки «Подходящие приложения» по окончании расчета приведен на рисунке 3.129:

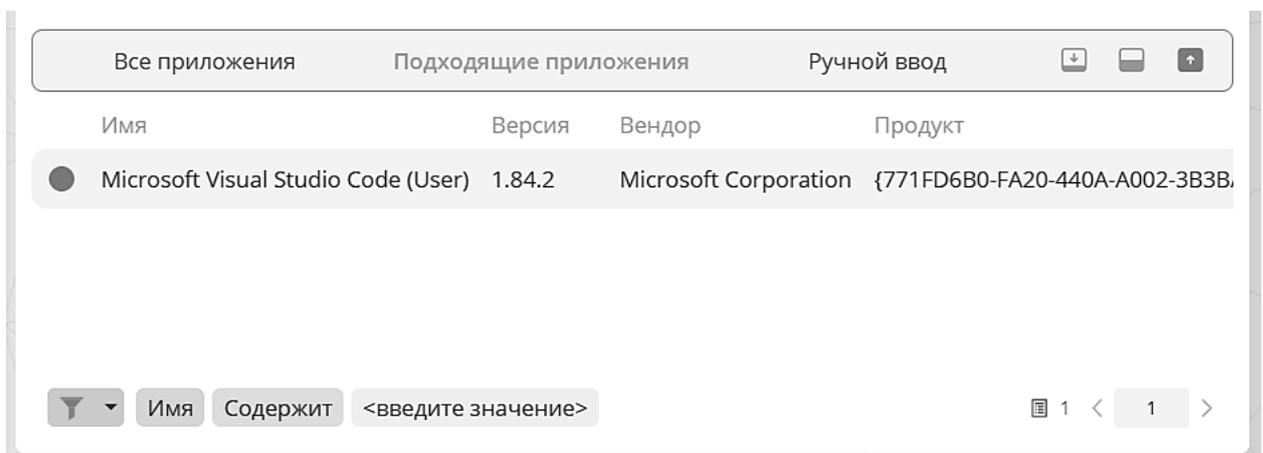


Рисунок 3.129 Вкладка «Подходящие приложения» области тестирования редактора правила Windows отчета ПО

## Примечание

При внесении изменений в любое выражение фильтра правила необходимо выполнить расчет подходящих приложений повторно.

Вкладка «**Ручной ввод**» предназначена для тестирования правила на данных о ПО, введенных вручную. Внешний вид данной вкладки приведен на рисунке 3.130:

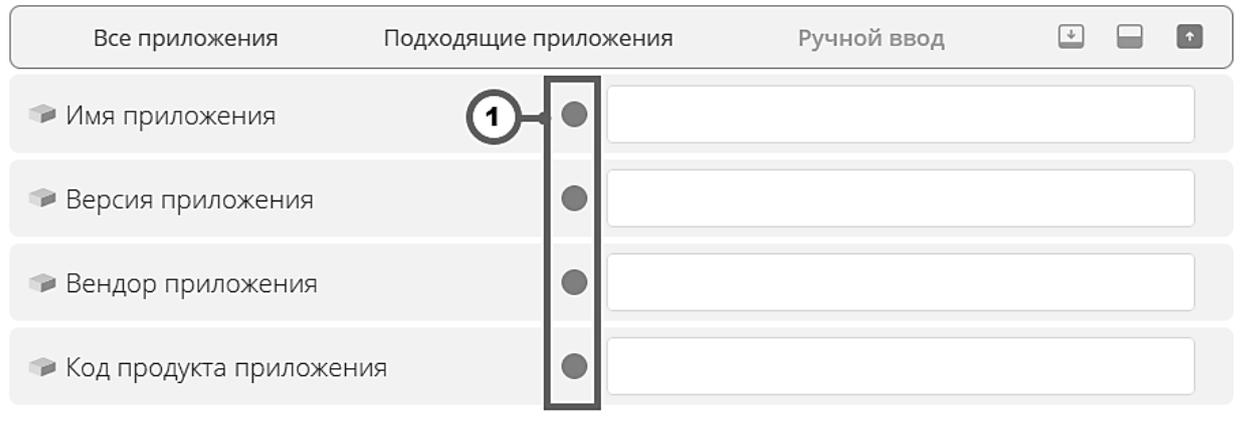


Рисунок 3.130 Вкладка «Ручной ввод» области тестирования редактора правила Windows отчета ПО

По мере ввода значений в поля вкладки цветковые индикаторы (1) будут окрашиваться в зеленый или серый цвет, свидетельствующий о соответствии введенного значения правилу.

### Совет

Используйте вкладку «Ручной ввод» для оценки применимости правила для будущих видов и версий ПО, еще не представленных в каталоге ПО и во вкладках «Все приложения» и «Подходящие приложения».

## 3.18.2 Расчет отчета

Созданный отчет может быть рассчитан в любой момент времени для актуальных данных инвентаризации, поступивших на серверы Зодиак.АйТиЭм. Для выполнения расчета воспользуйтесь пунктом контекстного меню «Рассчитать» в проводнике отчетов ПО:

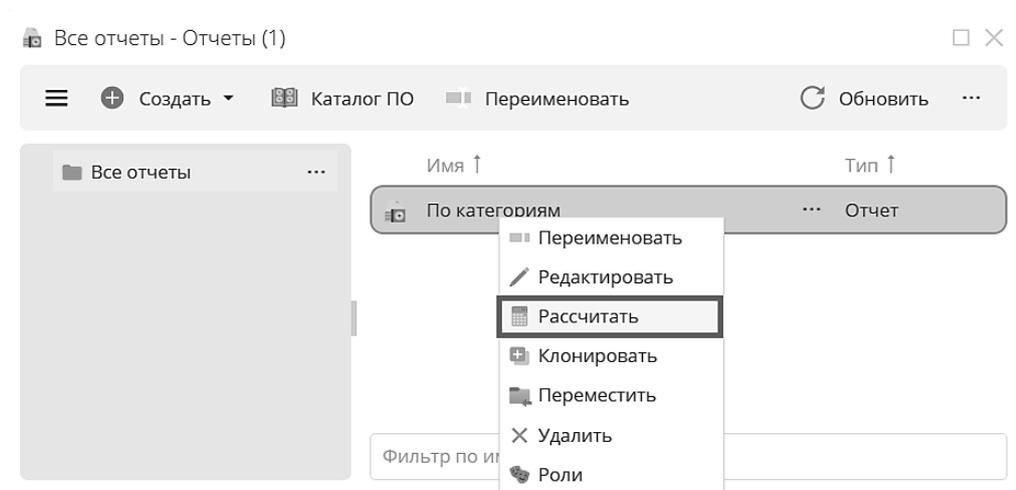


Рисунок 3.131 Пункт контекстного меню «Рассчитать» в проводнике отчетов ПО

### 3.18.3 Просмотр всех результатов расчета

Режим «Отчеты ПО» позволяет хранить для каждого отчета произвольное количество прошлых расчетов. Для просмотра результатов расчета выполните двойной клик левой клавишей мыши на интересующем отчете в проводнике отчетов ПО. В результате будет открыто окно результатов отчета, приведенное на рисунке 3.132:

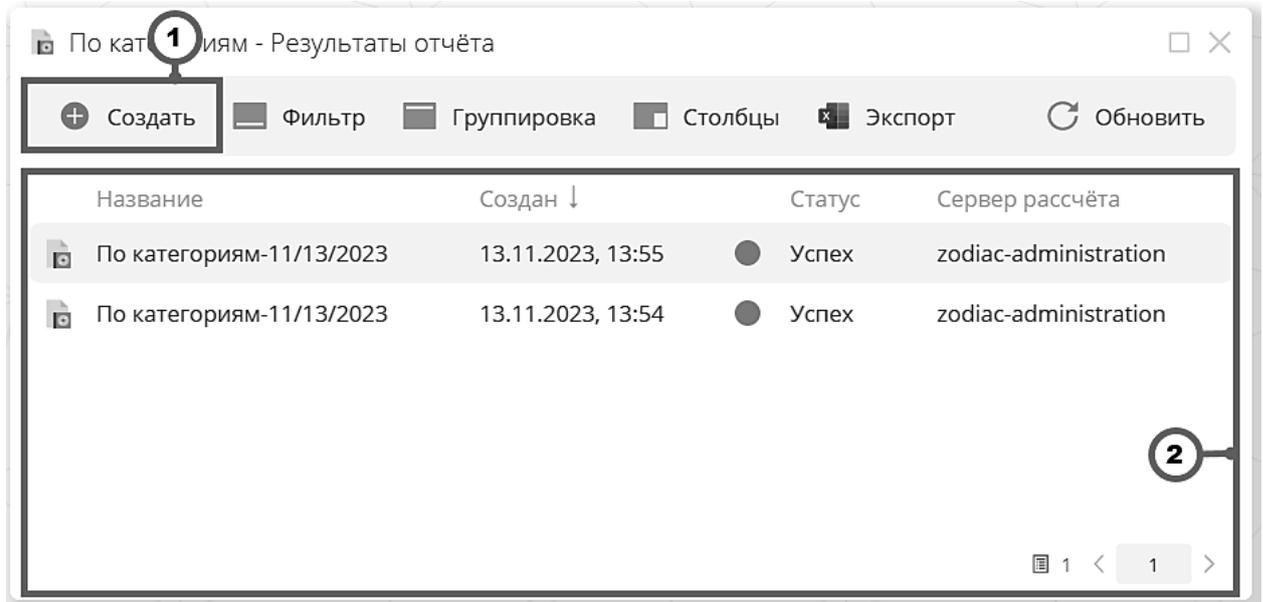


Рисунок 3.132 Окно просмотра всех результатов отчета

Основными элементами окна результатов расчета являются:

1. Кнопка **«Создать»** - предназначена для создания нового расчета отчета с возможностью выбора имени.
2. **Область визуализации расчетов** – предназначена для отображения и манипуляции расчетами отчета, сохраненными в Системе.

### 3.18.4 Просмотр конкретного результата расчета

Для просмотра конкретного результата расчета отчета выполните двойной клик на интересующем элементе окна просмотра всех результатов расчета, в следствие чего будет открыто окно конкретного результата отчета, приведенное на рисунке 3.133:

Имя	Компьютеры			Инсталляции		
	Lin	Win	Все	Lin	Win	Все
Офисное ПО	1	1	2	2	2	4
Текстовые редакторы	1	1	2	2	1	3
Kate	1	0	1	1	0	1
Visual Studio Code	1	1	2	1	1	2
1.83.1	1	0	1	1	0	1
Windows	0	0	0	0	0	0
Linux	1	0	1	1	0	1
1.84.2	0	1	1	0	1	1
Windows	0	1	1	0	1	1
Linux	0	0	0	0	0	0
Офисные пакеты	0	1	1	0	1	1
Microsoft Office	0	1	1	0	1	1

Рисунок 3.133 Окно просмотра конкретного результата отчета

Основными элементами окна результатов расчета являются:

1. Переключатель **отображаемых элементов** - предназначен для выбора элементов иерархии, отображаемых в отчете:
  - все – отображаются все виды элементов: категории, правила, исключения;
  - правила – отображаются только категории и правила;
  - категории – отображаются только категории.
2. **Область визуализации расчета** – предназначена для отображения статистических данных, полученных в ходе расчета.

## Примечание

Обратите внимание на то, что агрегаты по инсталляциям для каждого уровня иерархии представляют собой сумму по всем более низким уровням иерархии.

При этом агрегаты по компьютерам могут не быть связаны строгой формулой, поскольку рассчитываются индивидуально для каждого уровня иерархии как количество уникальных компьютеров.

## Совет

Обратите внимание, что большинство значений, отображаемых в области визуализации расчета являются интерактивными. Одиночное нажатие левой кнопкой мыши по интересующему значению приведет к открытию списка компьютеров или списка инсталляций, соответствующего выбранному значению:

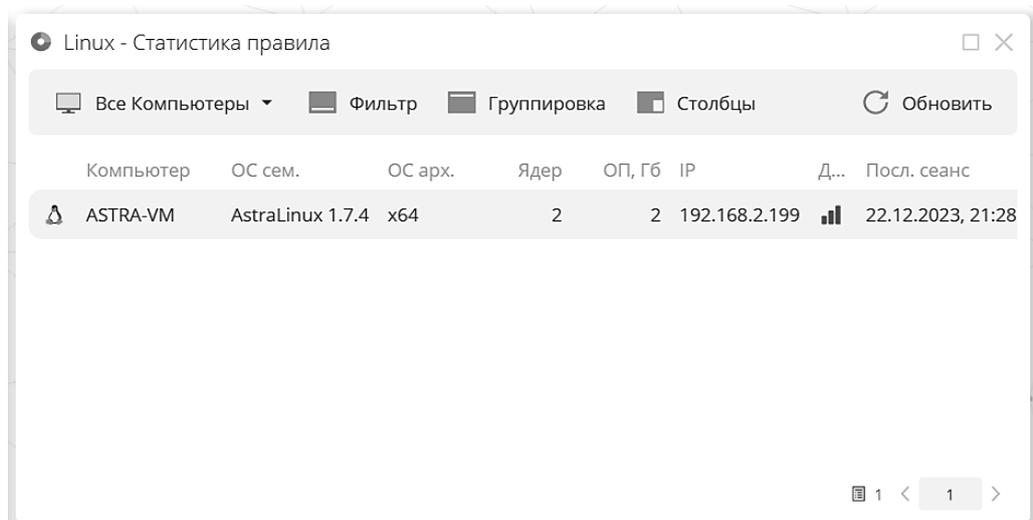


Рисунок 3.134 Окно списка компьютеров для выбранного уровня иерархии

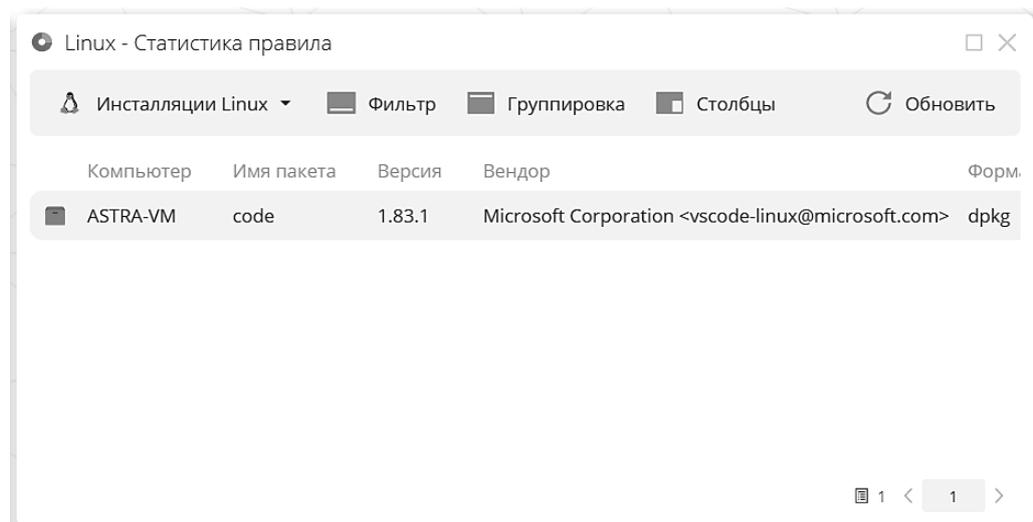


Рисунок 3.135 Окно списка инсталляций для выбранного уровня иерархии

### 3.19 Лицензия

Лицензия Зодиак.АйТиЭм выпускается на уникальный идентификатор, который создается в процессе установки системы Зодиак.АйТиЭм, и следовательно, различается для различных установок (рисунок 3.136).

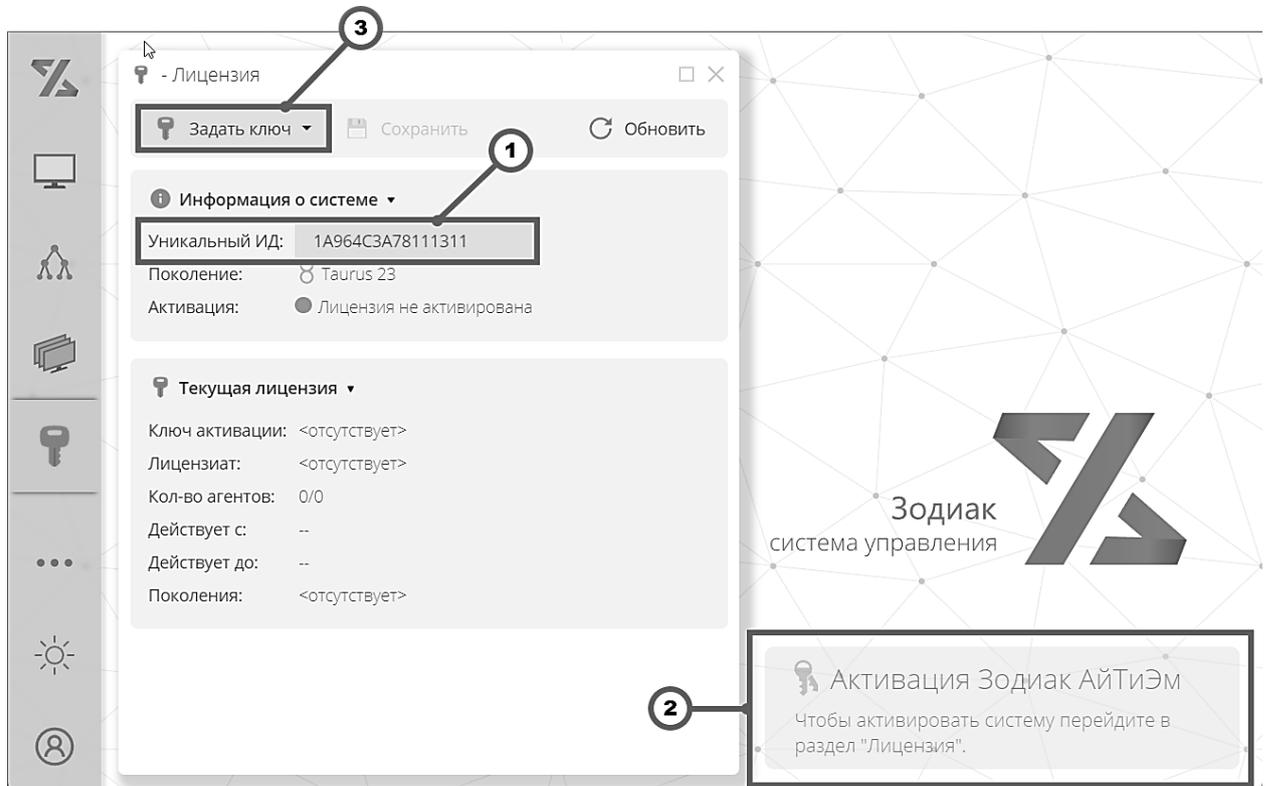


Рисунок 3.136 Окно управления лицензией

1. Уникальный **идентификатор установки**, на который выпускается лицензия.
2. **Водяной знак**, появляющийся в случае отсутствия активированной лицензии или превышения количества УК.
3. Кнопка **«Задать ключ»** вызывает диалог ввода ключа активации путем загрузки ключа из файла или копирования через буфер обмена.

#### **i** Примечание

Превышение количества УК ведет к отказу новым УК в коммуникации с сервером. Старые УК продолжают работать в штатном режиме.

Кроме того, поскольку определение выведенных из эксплуатации УК не происходит в реальном времени, система позволяет на некоторое время превысить количество УК с сохранением полной работоспособности.

## 4. ПОМОЩНИК АДМИНИСТРАТОРА

*Помощник Администратора Зодиак.АйТиЭм* (англ. Admin Aide) – опциональное приложение для рабочего стола ОС семейств Windows и Linux, расширяющее функциональные возможности веб-консоли Зодиак.АйТиЭм. Весь функционал веб-консоли, описанный в разделе 3, доступен без использования *Помощника*. Текущий раздел 4 содержит сведения о расширенном функционале, предоставляемым *Помощником*.

### 4.1 Общая схема работы

Схема работы *Помощника* основывается на возможностях современных браузеров, а именно: страница, загруженная с некоторого ресурса ([http\(s\)://mydomain](http(s)://mydomain)), может осуществлять коммуникации с веб-серверами после окончания своей загрузки, причем как с сервером, с которого она была загружена, так и с другими серверами.

*Помощник Администратора Зодиак.АйТиЭм* представляет собой локальный веб-сервер, запускаемый на том же АРМ, где расположен браузер, используемый для доступа к веб-консоли Зодиак.АйТиЭм. После своей загрузки страница веб-консоли Зодиак.АйТиЭм осуществляет коммуникации как с серверами Администрирования Зодиак.АйТиЭм (например, при переходах между режимами веб-консоли, осуществлении поиска, фильтрации и т.д.), так и с локальным сервером *Помощника* (при его наличии). Взаимодействие с серверами Администрирования Зодиак.АйТиЭм в промышленной конфигурации осуществляется по протоколу HTTPS, а с локальным сервером *Помощника* по протоколу HTTP. Для обеспечения дополнительной безопасности все содержимое таких HTTP-коммуникаций шифруется стойким криптографическим алгоритмом семейства AES-GCM. Поскольку веб-сервер *Помощника* является локальным, взаимодействие с ним за пределами АРМ, на котором он установлен, не предусматривается.

Само приложение *Помощника* представляет собой оболочку, содержащую логику локального веб-сервера и инфраструктуру **плагинов**. Весь расширенный функционал, предоставляемый *Помощником*, реализуется в виде отдельных отключаемых плагинов. Стандартный установочный дистрибутив *Помощника* уже содержит набор плагинов, которые не требуют дополнительной установки. Однако, этот набор может быть расширен дополнительными плагинами, не входящими в стандартный установочный дистрибутив. Такие дополнительные плагины могут предоставляться вендором, партнерами или быть созданы самостоятельно.

Описанная схема работы представлена в схематичном виде на рисунке 4.1.

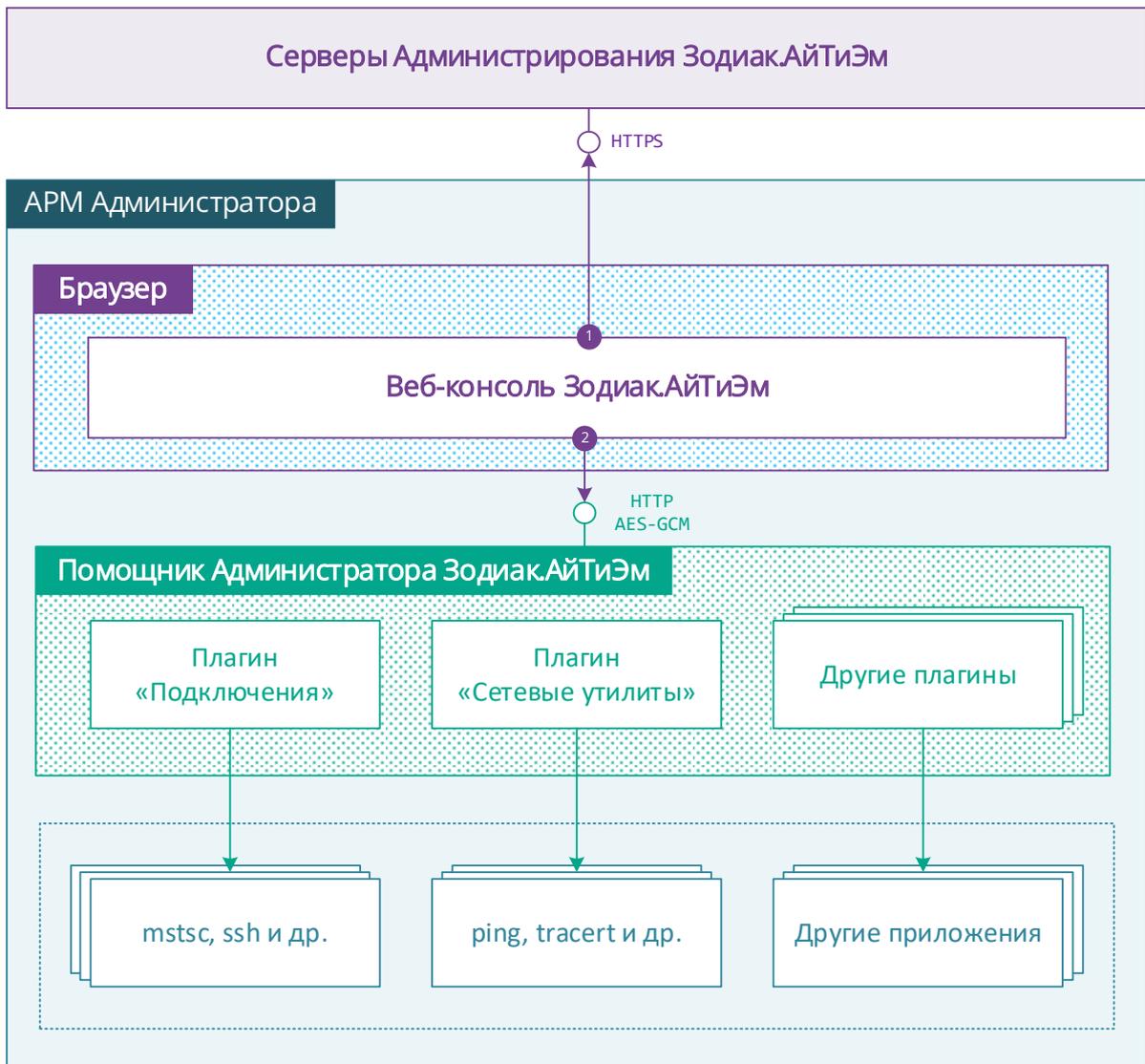


Рисунок 4.1 Общая схема работы Помощника Администратора

## 4.2 Установка приложения Помощника

Приложение Помощника Администратора Зодиак.АйТиЭм предоставляется в виде инсталляционных дистрибутивов для ОС семейства Windows (в формате \*.msi) и ОС семейства Linux (в форматах \*.rpm и \*.deb). Инсталляционные дистрибутивы содержат в своем составе все необходимые для поддерживаемых ОС зависимости. Установка дополнительных компонентов для работы Помощника не требуется.

---

### Примечание

Поскольку само приложение Помощника является оболочкой, указанное отсутствие зависимостей справедливо именно для него.

Плагины, как входящие в состав инсталляционного дистрибутива, так и поставляемые отдельно, могут предъявлять дополнительные требования к составу ПО и библиотек, установленных на АРМ.

Более подробная информация о необходимых зависимостях содержится в документации каждого плагина.

---

Инсталляционные дистрибутивы Помощника поставляются как отдельно, так и в составе серверов Администрирования Зодиак.АйТиЭм. Дистрибутивы, включенные в состав серверов Администрирования, доступны для скачивания из веб-консоли Зодиак.АйТиЭм.

---

### Осторожно

Для появления обновленных дистрибутивов Помощника в веб-консоли Зодиак.АйТиЭм требуется выполнить обновление серверов Администрирования. Такое обновление в крупных организациях зачастую запаздывает или вовсе опускается при незначительных изменениях.

При необходимости использования последней версии Помощника уточняйте актуальность дистрибутивов Помощника в веб-консоли или используйте более новую версию дистрибутивов Помощника, поставляемую отдельно от серверов Администрирования.

---

Скачивание дистрибутивов, включенных в состав серверов Администрирования, осуществляется через режим веб-консоли «Помощник».



Рисунок 4.2 Ярлык режима «Помощник» в перечне всех доступных режимов

---

**i** **Примечание**

Доступность режима «Помощник» регулируется ролевой моделью Зодиак.АйТиЭм. При отсутствии соответствующего режима в списке доступных обратитесь к Администратору ролей Зодиак.АйТиЭм для получения соответствующего доступа.

---

Внешний вид основного окна режима «Помощник» приведен на рисунке 4.3. Для скачивания инсталляционного дистрибутива нажмите на выпадающее меню «Скачать» (1) и выберите соответствующий вашей ОС вариант инсталляционного дистрибутива (2).

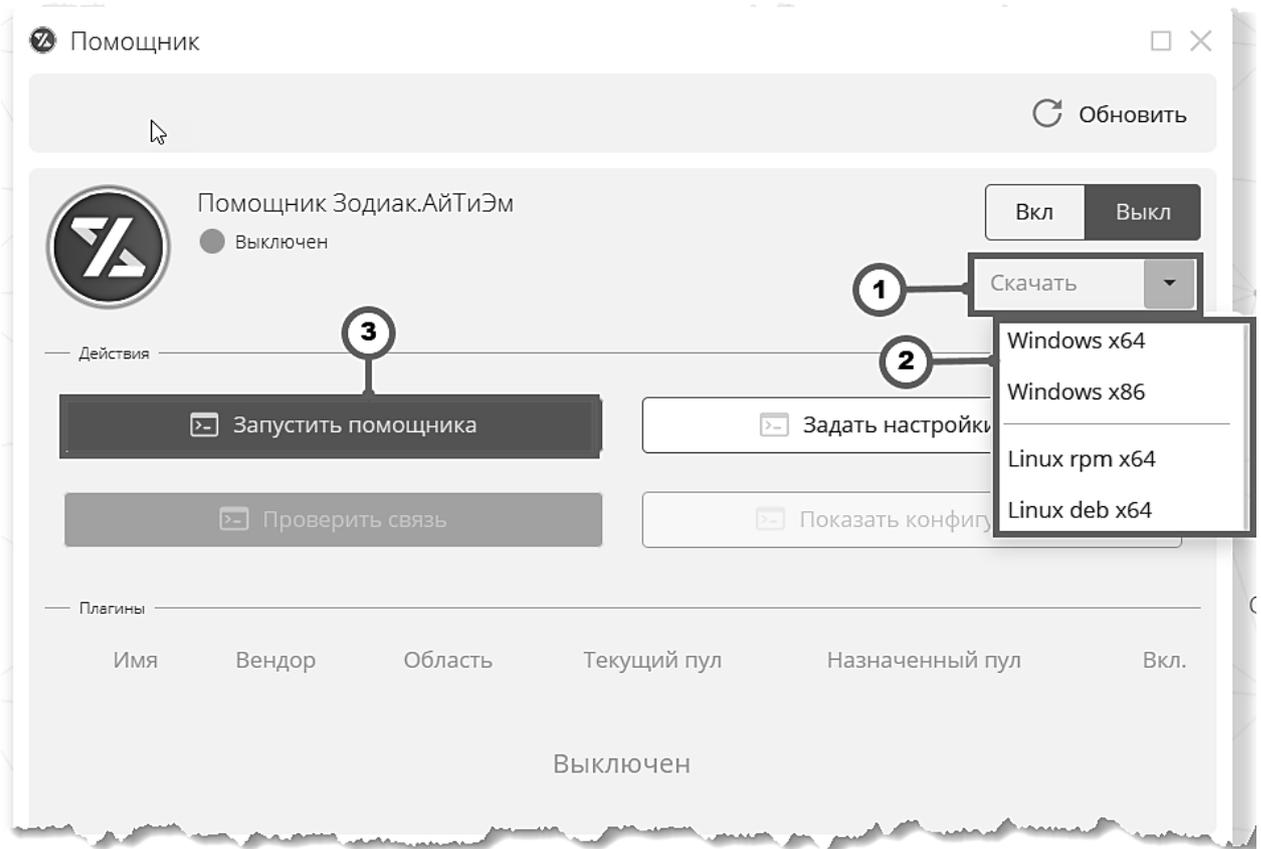


Рисунок 4.3 Внешний вид режима «Помощник» до выполнения настройки связи

### Совет

Скаченные посредством веб-браузеров файлы с расширением \*.msi по-умолчанию считаются недоверенными в ОС семейства Windows. Запуск такого инсталляционного файла сразу после скачивания может быть заблокирован.

Разблокируйте скаченный инсталляционный файл перед запуском через стандартное окно свойств файла ОС семейства Windows или воспользуйтесь альтернативным способом получения инсталляционного дистрибутива Помощника.

### 4.3 Начальная настройка приложения Помощника

После скачивания и установки дистрибутива Помощника Администратора Зодиак.АйТиЭм необходимо выполнить настройку его связи с веб-консолью Зодиак.АйТиЭм и включить требуемые плагины.

---

#### Примечание

Настройка связи приложения Помощника с веб-консолью Зодиак.АйТиЭм сохраняется в веб-браузере в контексте используемого адреса веб-консоли. При смене веб-браузера, смене адреса веб-консоли, очистке сохраненных данных в веб-браузере требуется произвести настройку связи повторно.

Настройка включенных плагинов Помощника сохраняется в приложении Помощника в контексте используемой УЗ пользователя. При смене УЗ или очистке данных помощника требуется произвести включение плагинов повторно.

---

Начальная настройка приложения Помощника состоит из следующих действий:

1. Запустите приложение Помощника одним из следующих способов:
  - a. посредством ярлыка Рабочего стола;
  - b. посредством ярлыка в меню «Пуск» или меню приложений ОС;
  - c. посредством кнопки «Запустить помощника» режима «Помощник» веб-консоли (см. элемент 3 на рисунке 4.3).

---

#### Примечание

Доступность запуска внешних приложений посредством веб-браузера (с) зависит от используемого браузера и ОС. При недоступности данного функционала воспользуйтесь альтернативными способами запуска (a, b).

---

2. Задайте настройки связи посредством нажатия кнопки «Задать настройки связи» (элемент 1 на рисунке 4.4) режима «Помощник» веб-консоли Зодиак.АйТиЭм и выбора файла настроек связи, созданного Помощником при первом запуске:

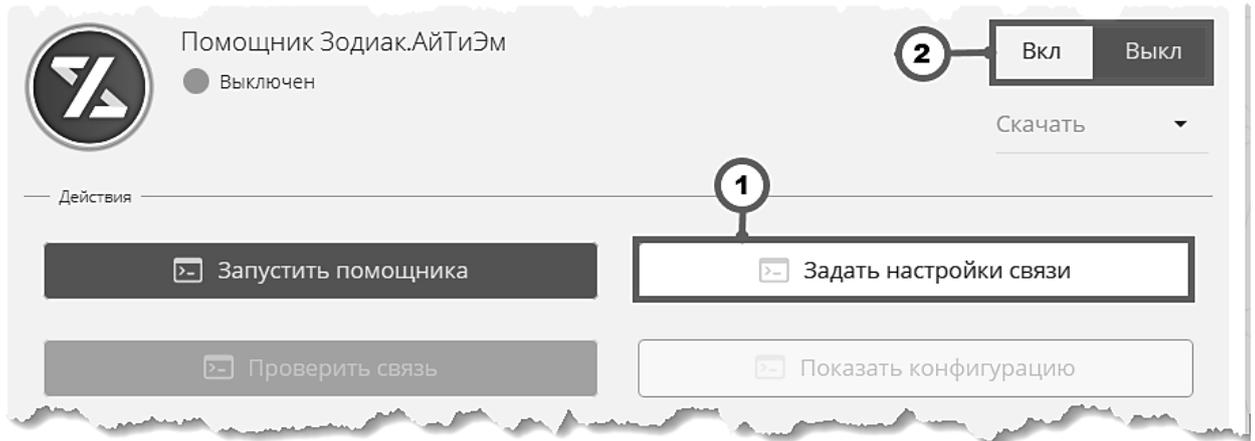


Рисунок 4.4 Внешний вид режима «Помощник» до выполнения настройки связи

Расположение файла настроек связи по-умолчанию для ОС семейства Windows:

```
%LocalAppData%\Zodiac\AdminAide\link.settings.ini
```

Расположение файла настроек связи по-умолчанию для ОС семейства Linux:

```
~/zodiac-admin-aide/link.settings.ini
```

### Совет

Для быстрого открытия визуального проводника файлов в папке расположения файла настроек связи воспользуйтесь пунктом «Показать настройки связи» контекстного меню иконки Помощника в системном древе (элемент 1 на рисунке 4.5).

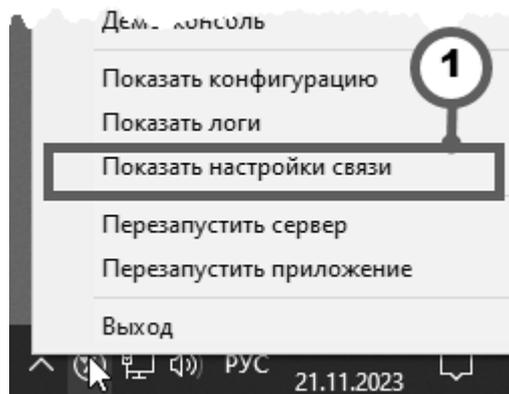


Рисунок 4.5 Меню иконки в системном древе приложения «Помощник Администратора»

3. (При необходимости) Включите связь с Помощником путем перевода переключателя включения функционала Помощника в положение «Вкл» (элемент 1 на рисунке 4.6). Внешний вид основного окна режима «Помощник» веб-консоли Зодиак.АйТиЭм после настройки связи приведен на рисунке 4.6:

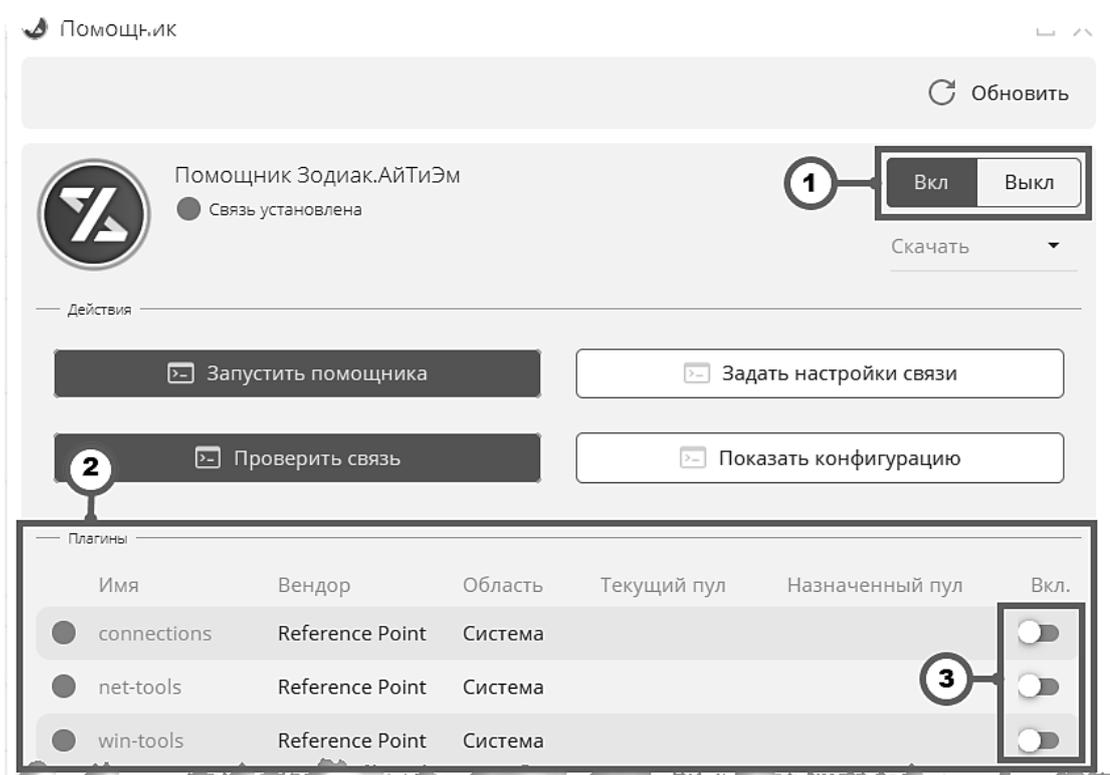


Рисунок 4.6 Внешний вид режима «Помощник» после выполнения настройки связи

На приведенном рисунке использованы следующие обозначения:

1. **Переключатель включения функционала Помощника** - для корректной работы расширенного функционала, предоставляемого Помощником, данный переключатель должен находиться в положении «Вкл».
2. **Список доступных плагинов** – содержит информацию об установленных и доступных к использованию плагинах.
3. **Переключатели включения плагинов** – отображают текущее состояние включения плагинов и позволяют управлять им.
4. Включите необходимые плагины путем перевода переключателей включения во включенное положение. Для применения изменений и загрузки включенных плагинов требуется перезапустить приложение Помощника, о чем сигнализирует информационное сообщение, отображаемое в списке доступных плагинов. Внешний вид списка доступных плагинов после включения всех или части из них приведен на рисунке 4.7:

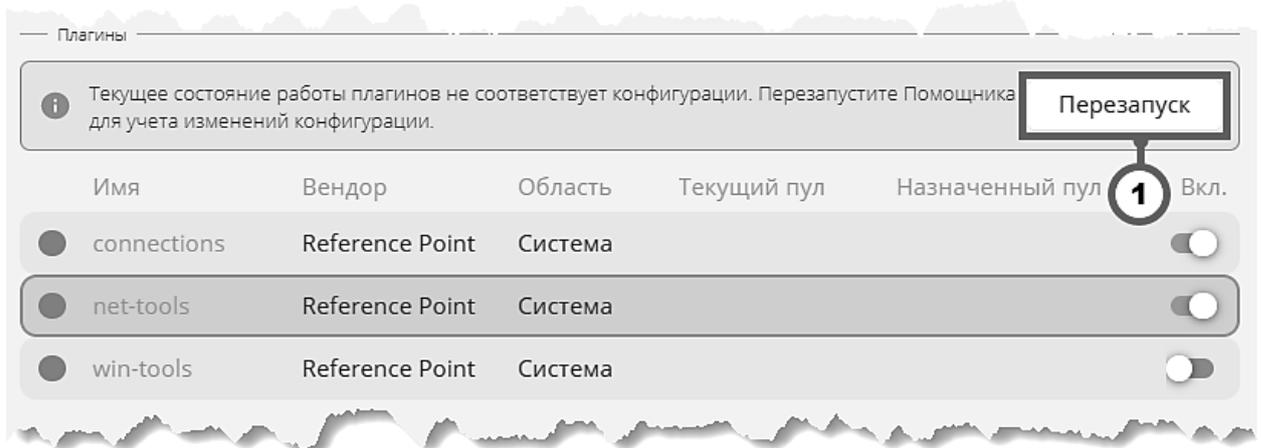


Рисунок 4.7 Внешний вид списка доступных плагинов после включения плагинов

5. Выполните перезапуск приложения Помощника одним из следующих способов:
- посредством нажатия кнопки «Перезапуск» информационного сообщения в списке доступных плагинов (элемент 1 на рисунке 4.7);
  - посредством выбора пункта «Перезапустить приложение» контекстного меню иконки Помощника в системном трее (элемент 1 на рисунке 4.8);
  - посредством последовательного завершения работы приложения Помощника путем выбора пункта «Выход» контекстного меню иконки Помощника в системном трее (элемент 2 на рисунке 4.8) и последующего повторного запуска приложения Помощника одним из способов, приведенных в пункте 1.

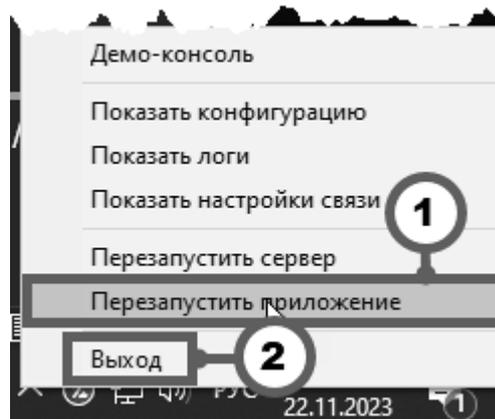


Рисунок 4.8 Меню иконки в системном трее приложения «Помощник Администратора»

6. Выполните обновление основного окна режима «Помощник» путем нажатия на кнопку «Обновить» (элемент 1 на рисунке 4.9). Внешний вид окна после корректного выполнения всех описанных действий приведен на рисунке 4.9:

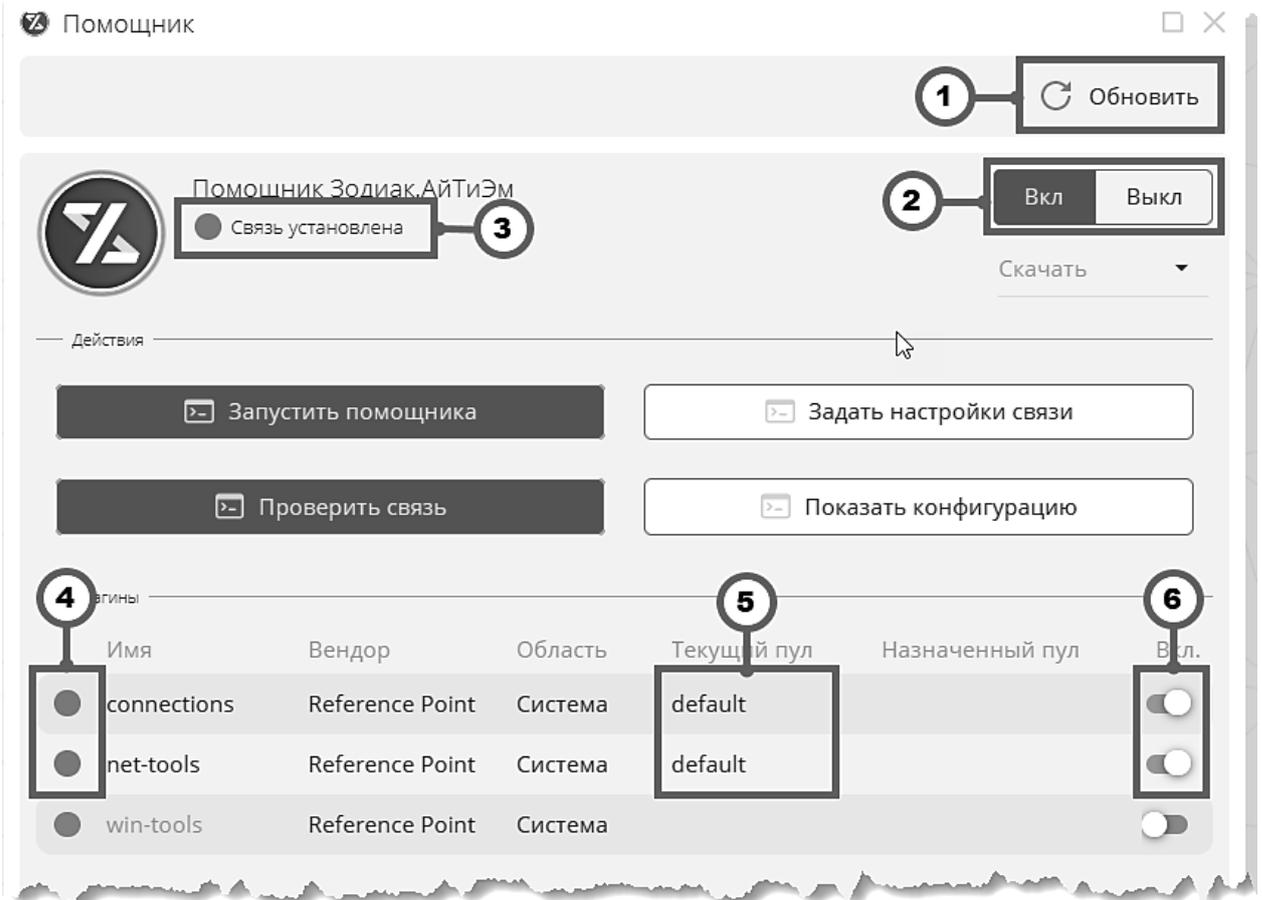


Рисунок 4.9 Меню иконки в системном tree приложения «Помощник Администратора»

На приведенном рисунке отмечены следующие маркеры, свидетельствующие о корректном завершении настройки связи:

2. Переключатель включения функционала Помощника находится в положении «Вкл».
3. Индикатор связи с приложением Помощника окрашен в зеленый цвет. Отображается сообщение «Связь установлена».
4. Индикатор работы для включенных плагинов окрашен в зеленый цвет.
5. Текущий пул для включенных плагинов не пуст.
6. Переключатель включения для включенных плагинов находится в положении «Вкл».

## 4.4 Общие принципы размещения функционала Помощника в веб-консоли

Поскольку приложение Помощника Администратора Зодиак.АйТиЭм является модульным и предполагает наполнение его функционала при помощи плагинов, в веб-консоли Зодиак.АйТиЭм используются строгие правила размещения такого функционала, позволяющие структурировать функции плагинов и обеспечить единообразный подход к их вызову из веб-консоли.

Основными способами вызова функционала Помощника из веб-консоли, приведенными на рисунке 4.10, являются:

1. Контекстное меню списков и таблиц;
2. Меню «Помощник» в панели инструментов окон.

Оба способа являются эквивалентными и предоставляют доступ к одному и тому же функционалу.

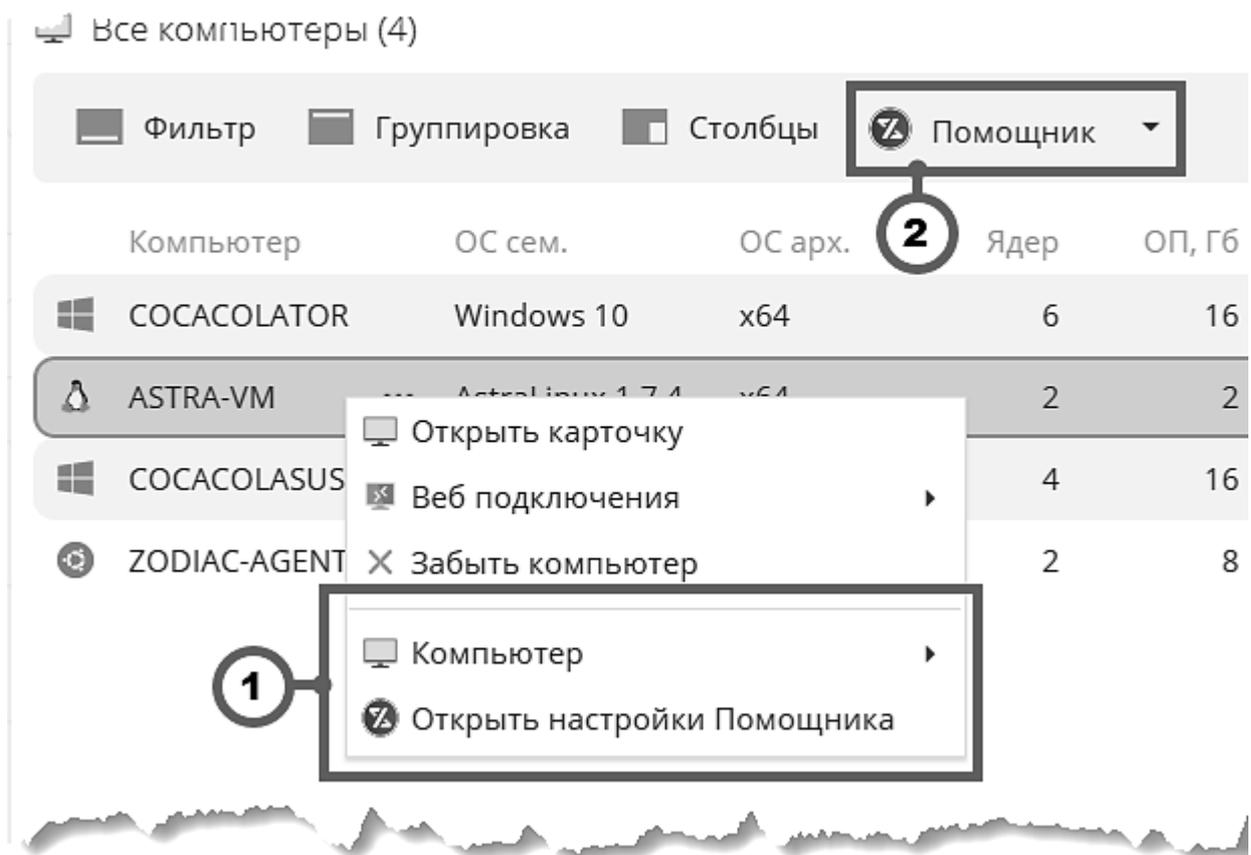


Рисунок 4.10 Способы вызова функционала Помощника на примере окна режима «Все компьютеры»

### **i** Примечание

Перечисленные способы вызова Помощника доступны не из всех окон и не из всех режимов веб-консоли Зодиак.АйТиЭм.

Каждый включенный плагин публикует набор пунктов меню для элементов списков / строк таблиц. Такие пункты меню автоматически собираются в трехуровневое дерево:

```
<КОРЕНЬ>
|
|-- Контекст 1
|   |
|   |-- Категория 1.1
|   |   |
|   |   |-- Элемент меню 1.1.1
|   |   |-- Элемент меню 1.1.2
|   |
|   |-- Категория 1.2
|   |   |
|   |   |-- Элемент меню 1.2.1
|   |   |-- Элемент меню 1.2.2
|   |   |-- Элемент меню 1.2.3
|
|-- Контекст 2
|   |
|   |-- Категория 2.1
|   |   |
|   |   |-- Элемент меню 1.1
|   |
|   |-- Категория 2.2
|   |   |
|   |   |-- Элемент меню 2.2.1
|   |   |-- Элемент меню 2.2.2
```

**Контекст** – определяет к чему применяются пункты меню. Например, элемент списка компьютеров может поддерживать единственный контекст: «Компьютер», а элемент списка установленных на компьютер пакетов – два контекста: «Компьютер» и «Пакет». Таким образом, элементы меню в контексте «Компьютер» будут применяться к компьютеру, на который установлен пакет, а элементы меню в контексте «Пакет» - к пакету.

**Категория** – строковая характеристика пунктов меню, позволяющая сгруппировать их по папкам для облегчения навигации.

**Элемент меню** – потенциальное действие, которое может исполнить Помощник при его вызове.

---

### Примечание

В большинстве случаев элементы списков / таблиц в веб-консоли Зодиак.АйТиЭм поддерживают единственный контекст.

---

Внешний вид контекстного меню элемента списка режима «Все компьютеры» приведен на рисунке 4.11. На данном рисунке отмечены: контекст «Компьютер» (1), категории контекста «Компьютер» (2), пункты категории «Сеть» (3).

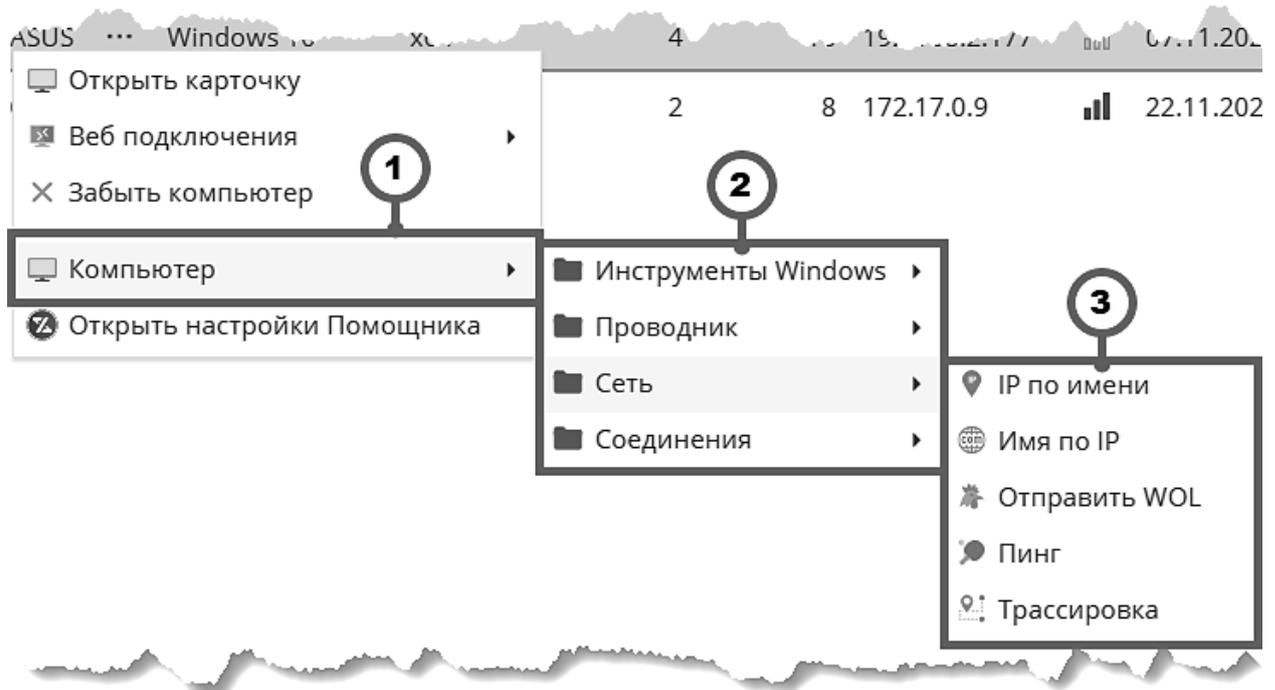


Рисунок 4.11 Внешний вид контекстного меню в контексте «Компьютер»

Внешний вид меню «Помощник» в панели инструментов основного окна режима «Все компьютеры» приведен на рисунке 4.12. На данном рисунке отмечены те же самые уровни дерева, что и на рисунке 4.11: контекст (1), категории (2), пункты меню (3).

### Совет

Ключевым отличием вызова функционала Помощника из контекстного меню и меню «Помощник» в области инструментов окна является возможность поиска по пунктам меню в последнем.

Используйте вызов функционала Помощника из области инструментов окна в тех случаях, когда Вы не уверены в точном расположении требуемого пункта меню, например, при изучении функциональных возможностей, предоставляемых Помощником.

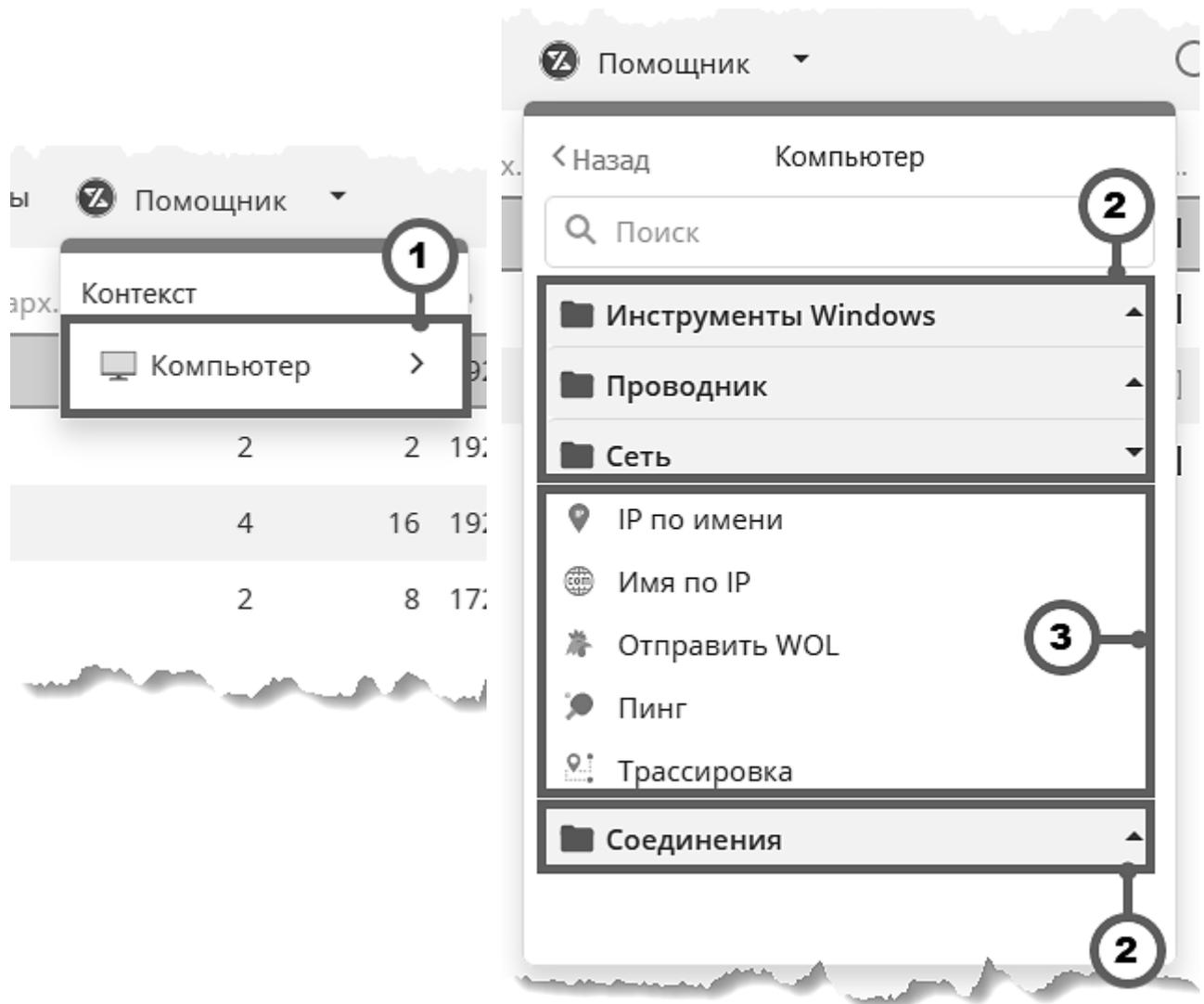


Рисунок 4.12 Внешний вид меню «Помощник» в панели инструментов окна в контексте «Компьютер»

### ! Предупреждение

Каждый плагин Помощника Администратора Зодиак.АйТиЭм выполняет публикацию пунктов меню в момент открытия меню, на основании данных об элементе, для которого оно было открыто.

Набор пунктов меню не фиксирован и может отличаться в зависимости от множества факторов: семейство ОС АРМ Администратора, семейство ОС управляемого компьютера, доступность того или иного ПО на АРМ Администратора, и других.

Например, функционал вызова штатных оснасток управления Windows доступен только для управляемых компьютеров под управлением ОС семейства Windows и при вызове меню с АРМ Администратора под управлением ОС семейства Windows.

## 4.5 Конфигурация приложения Помощника

Приложение Помощника является модульным и предполагает наполнение предполагает наполнение его функционала при помощи плагинов. Само приложение Помощника представляет собой оболочку, обеспечивающую общие подходы к размещению и вызову функционала плагинов, и не содержит бизнес-логику.

Таким образом, конфигурирование приложения Помощника предназначено для изменения общих инфраструктурных аспектов его работы. Конфигурирование аспектов работы бизнес-функционала осуществляется через конфигурацию плагинов, описанную в разделе 4.11.

Приложение Помощника поддерживает конфигурирование при помощи двух конфигурационных файлов:

- системного конфигурационного файла;
- пользовательского конфигурационного файла.

Данные конфигурационные файлы совпадают<sup>1</sup> по набору параметров. Пользовательский конфигурационный файл имеет более высокий приоритет. При запуске приложения Помощника осуществляется **чтение и слияние** обоих конфигурационных файлов, а затем проверка результата слияния на полноту.

### Примечание

Системный конфигурационный файл после установки дистрибутива Помощника содержит все поддерживаемые конфигурационные параметры. Любой<sup>1</sup> из этих параметров может быть переопределен как путем изменения системного конфигурационного файла, так и путем переопределения через пользовательский конфигурационный файл.

Расположения конфигурационных файлов по-умолчанию приведены в таблице 4.6:

Конфиг. файл	ОС	Расположение плагинов
Системный	Windows	C:\Program Files\Zodiac\AdminAide\aide.ini
	Linux	/opt/zodiac/admin-aide/aide.ini
Пользовательский	Windows	%LocalAppData%\Zodiac\AdminAide\aide.ini
	Linux	~/zodiac-admin-aide/aide.ini

Таблица 4.6. Расположения папок хранения данных плагинов Помощника

<sup>1</sup> Единственной секцией системного конфигурационного файла, не поддерживающей изменение через пользовательский конфигурационный файл, является секция [User]. Данная секция определяет местоположение папки Помощника для каждой УЗ (и содержащегося в ней пользовательского конфигурационного файла) и файла блокировки.

Наиболее частым и рекомендованным сценарием является изменение конфигурации приложения Помощника при помощи пользовательского конфигурационного файла. Данный способ не требует административных полномочий, не затрагивает конфигурацию для других УЗ на АРМ и сохраняет изменения при обновлении приложения Помощника.

При запуске от имени конкретной УЗ приложение Помощника проверяет наличие пользовательского конфигурационного файла в профиле данной УЗ и при его отсутствии выполняет автоматическое создание по шаблону. Созданный по шаблону пользовательский конфигурационный файл содержит только закомментированные строки и, тем самым, не переопределяет параметры системного конфигурационного файла. Закомментированные строки содержат описание и примеры секций и конфигурационных параметров.

Расположения шаблона пользовательских конфигурационных файлов приведены в таблице 4.7:

ОС	Расположение плагинов
Windows	C:\Program Files\Zodiac\AdminAide\config.template.ini
Linux	/opt/zodiac/admin-aide/config.template.ini

Таблица 4.7. Расположения шаблонов пользовательских конфигурационных файлов

Таким образом, рекомендованное изменение конфигурации приложения Помощника сводится к открытию созданного при первом запуске пользовательского конфигурационного файла и раскомментированию секций и параметров, требующих изменения.

---

### Примечание 1

Во избежание расхождения с конкретными версиями приложения Помощника, данный раздел руководства Администратора не содержит описание секций и конфигурационных параметров. Для получения актуальной информации о поддерживаемых секциях и параметрах обратитесь к создаваемому по шаблону пользовательскому конфигурационному файлу.

---

### Примечание 2

При обновлении приложения Помощника **производится перезапись системного конфигурационного файла и не производится перезапись пользовательских конфигурационных файлов**. Новые версии Помощника могут поддерживать расширенный набор секций и их параметров.

За счет использования слияния системного и пользовательских конфигурационных файлов, новые секции и параметры будут иметь значения по-умолчанию, заданные в системном конфигурационном файле, создаваемом при установке новой версии Помощника.

### Совет 1

При необходимости возврата к значениям по-умолчанию выполните переименование или удаление пользовательского конфигурационного файла и перезапустите приложение Помощника. Данное действие приведет к созданию нового пользовательского конфигурационного файла по шаблону.

---

### Совет 2

При необходимости получения информации о поддерживаемых конфигурационных параметрах (например, после обновления приложения Помощника) без удаления/переименования уже созданного пользовательского конфигурационного файла обратитесь к файлу шаблона напрямую по путям, приведенным в таблице 4.7.

---

### Совет 3

Для быстрого открытия пользовательского конфигурационного файла воспользуйтесь пунктом «Показать конфигурацию» контекстного меню Помощника в системном трее (элемент 1 на рисунке 4.13).

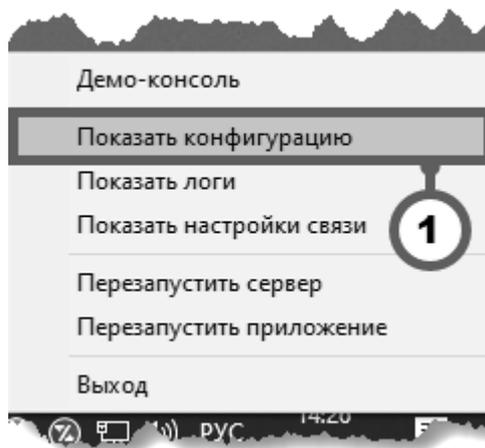


Рисунок 4.13 Пункт «Показать конфигурацию» контекстного меню Помощника в системном трее

---

## 4.6 Журналы приложения Помощника

Приложение Помощника Администратора Зодиак.АйТиЭм осуществляет журналирование (логирование) различных аспектов своей работы в текстовые журналы. Журналирование осуществляется в контексте УЗ, от имени которой было запущено приложение Помощника, т.е. для каждой УЗ ведутся независимые текстовые журналы.

Расположение текстовых журналов по-умолчанию приведено в таблице 4.8:

ОС	Расположение плагинов
Windows	%LocalAppData%\Zodiac\AdminAide\log\
Linux	~/zodiac-admin-aide/log/

Таблица 4.8. Расположения текстовых журналов приложения Помощника



### Совет

Для быстрого открытия папки текстовых журналов воспользуйтесь пунктом «Показать логи» контекстного меню Помощника в системном трее (элемент 1 на рисунке 4.14).

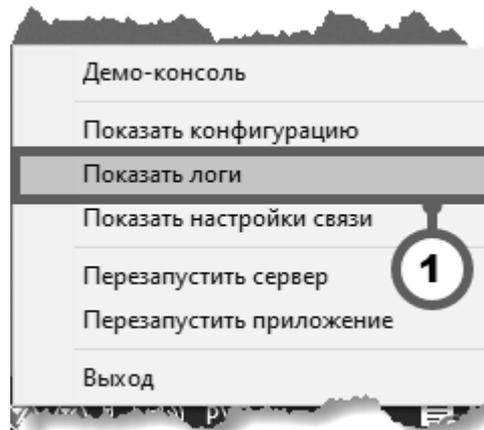


Рисунок 4.14 Пункт «Показать логи» контекстного меню Помощника в системном трее

Для создаваемых текстовых журналов осуществляется поддержание предельного размера каждого журнала и предельного количества файлов журналов. При достижении предельного количества файлов журналов осуществляется автоматическая **ротация**, т.е. перезапись самого старого файла журнала целиком.

Регулирование подробности ведения журналов, предельного размера и предельного количества файлов журнала осуществляется через конфигурацию приложения Помощника. Особенности конфигурирования приложения Помощника описаны в разделе 4.5.

Пример секции [Log], задающей различные аспекты логирования приведен ниже:

```
[Log]
dir=log
files=4
fileSize=1048576
level=info
```

Описание секции [Log], содержащееся в шаблоне пользовательского файла конфигурации приведено ниже:

```
[Log]

dir=      Путь до папки логов.
          Разрешается относительно пользовательской папки Помощника.

files=    Максимальное количество файлов логов до ротации.

fileSize= Максимальный размер каждого файла в байтах до ротации.

level=    Подробность логов.

error     - только ошибки
warn      - ошибки и предупреждения
info      - ошибки, предупреждения и информационные сообщения
verbose   - ошибки, предупреждения, информационные и избыточные сообщения
debug     - ошибки, предупреждения, информационные, избыточные и отладочные сообщения
```



### Совет

Для определения файла журнала, содержащего последние сообщения, воспользуйтесь сортировкой по дате модификации при просмотре папки журналов.

---

## 4.7 Изменение настроек связи приложения Помощника и веб-консоли

Приложение Помощника и вкладка браузера, в которой открыта веб-консоль Зодиак.АйТиЭм, осуществляют сетевое взаимодействие по протоколу HTTP. Несмотря на то, что такое взаимодействие происходит исключительно в контуре одного APM по *localhost*-интерфейсу и считается безопасным и разрешенным в современных браузерах, приложение Помощника и веб-консоль Зодиак.АйТиЭм осуществляют дополнительное шифрование передаваемых данных стойким криптографическим алгоритмом семейства AES-GCM.

Помимо этого, приложение Помощника поддерживает работу нескольких своих копий от имени разных УЗ на одном и том же APM (например, в разных сессиях на одном VDI-сервере).

Таким образом, для обеспечения связи между веб-консолью Зодиак.АйТиЭм и приложением Помощника необходимо сообщить веб-консоли Зодиак.АйТиЭм следующие сведения о приложении Помощника:

- **ключ шифрования**, используемый приложением Помощника;
- **порт**, на котором запущен локальный сервер Помощника.

Эти сведения содержатся в едином **файле настроек связи** и уникальны для каждой УЗ каждого компьютера, на котором установлено приложение Помощника.

Начальная настройка связи между веб-консолью и приложением Помощника описана в разделе 4.3. Единоразово настроенная связь между веб-консолью (в конкретном браузере) и приложением Помощника остается рабочей до очистки данных браузера или изменения настроек связи в приложении Помощника.

---

### **Внимание**

Для повышения уровня безопасности и защиты от перебора портов и ключей шифрования рекомендуется регулярно менять настройки связи приложения Помощника.

---

Приложение Помощника оснащено функционалом напоминаний о необходимости смены настроек связи. Напоминания организованы посредством механизма уведомлений ОС. Примеры соответствующего уведомления приведены на рисунках 4.15 и 4.16:

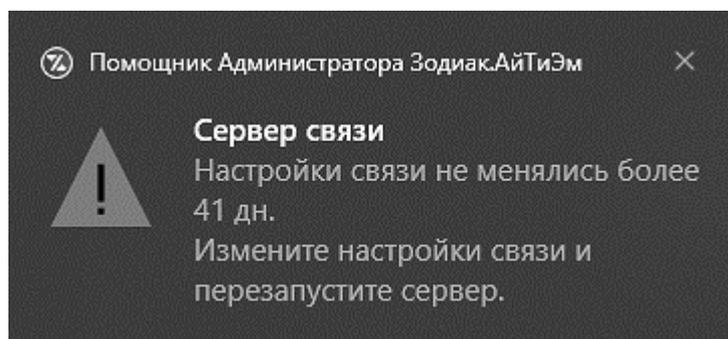


Рисунок 4.15 Уведомление о необходимости смены настроек связи в ОС семейства Windows

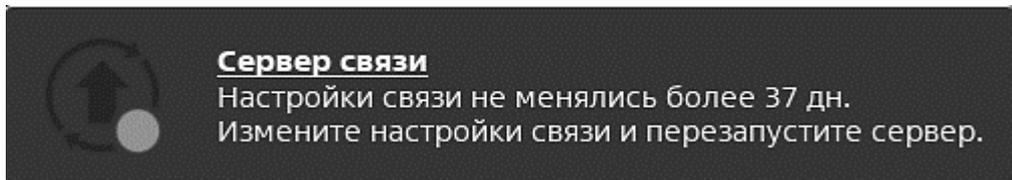


Рисунок 4.16 Уведомление о необходимости смены настроек связи в ОС семейства Linux

Уведомление о необходимости смены настроек связи формируется при каждом запуске приложения Помощника при выявлении факта превышения допустимого времени работы с одними и теми же настройками связи. Определение времени работы с настройками связи происходит на основе атрибута «Время изменения» (*англ. Date Modified, mtime*) для файла настроек связи. Регулирование допустимого времени работы с настройками связи осуществляется через параметр `prompt` секции `[Key]` конфигурации приложения Помощника. Пример секции `[Key]` приведен ниже:

```
[Key]
length=128
prompt=30.00:00:00
```

---

### ! Внимание

Обратите внимание на синтаксис интервала времени параметра `prompt`:

<ДНЕЙ> . <ЧАСОВ> : <МИНУТ> : <СЕКУНД>

Для исключения разночтения задаваемого интервала рекомендуется указывать **все его составные части**. При необходимости задания короткого интервала используйте значение `0` для мажорных частей, например `0.12:00:00`.

---

### ! Внимание

Параметр `length` секции `[Key]` задает требуемую длину ключа шифрования в битах. При изменении данного параметра требуется скорректировать существующие настройки связи для использования ключа соответствующей длины.

---

Изменение настроек связи возможно двумя способами:

1. Ручной модификацией файла настроек связи;
2. Консольной командой приложения Помощника.

## **Внимание**

Приложение Помощника осуществляет чтение настроек связи при своем запуске и перезапуске локального веб-сервера. Для применения измененных настроек связи требуется выполнить перезапуск приложения Помощника или его локального веб-сервера. После применения новых настроек связи в приложении Помощника требуется повторно выполнить процедуру задания настроек связи в веб-консоли Зодиак.АйТиЭм, описанную в разделе 4.3.

---

### 4.7.1 Ручная модификация файла настроек связи

Файл настроек связи представляет собой текстовый файл в формате *\*.ini*. Его ручное изменение производится при помощи произвольного текстового редактора.

Расположение файла настроек связи по-умолчанию для ОС семейства Windows:

```
%LocalAppData%\Zodiac\AdminAide\link.settings.ini
```

Расположение файла настроек связи по-умолчанию для ОС семейства Linux:

```
~/zodiac-admin-aide/link.settings.ini
```

Пример файла настроек связи:

```
key=ab6bc336ac4eddd1fdb53ff621f47374  
port=39407
```

Файл настроек связи поддерживает два корневых параметра `key` и `port` и не предполагает использование секций. Параметр `key` задает используемый ключ шифрования в HEX-формате. Параметр `port` задает порт локального веб-сервера приложения Помощника.

---

## **Совет**

Для повышения уровня безопасности и защиты от перебора рекомендуется регулярно изменять оба параметра: и `key`, и `port`.

---

## **Внимание**

При ручном изменении параметра `port` необходимо убедиться в том, что задаваемый порт свободен на текущем АРМ. Для уменьшения вероятности пересечения с другими приложениями рекомендуется использовать порты в диапазоне 6,000 – 65,535.

---

## **Внимание**

При ручном изменении параметра `key` необходимо убедиться в том, что задаваемое значение является корректным ключом для алгоритма AES-GCM и его длина соответствует параметру `length` секции `[Key]` конфигурации приложения Помощника.

---

#### 4.7.2 Консольные команды изменения настроек связи

Приложение Помощника предоставляет консольную команду для изменения файла настроек связи. В рамках исполнения данной команды приложение Помощника выполняет:

- автоматическую генерацию **ключа шифрования** с длиной, соответствующей текущей конфигурации приложения Помощника;
- выбор случайного **порта** в диапазоне 6,000 – 65,535 с проверкой, что выбранный порт свободен в момент совершения выбора.

---

#### **Внимание**

Проверка, что выбранный случайным образом порт свободен, осуществляется в момент исполнения команды изменения настроек связи. Если на АРМ в момент исполнения команды не было запущено другое приложение, использующее тот же самый порт, то данная проверка завершится успехом, а конфликт по портам может проявиться позднее.

Если на АРМ представлены приложения с нерегулярной работой, но использующие строго определенный порт – убедитесь, что выбранный случайным образом порт не конфликтует с другими приложениями.

---

Изменение настроек связи производится при помощи следующей команды:

```
> zoaide link settings reset
```

Вызов данной команды приводит к созданию **нового** ключа шифрования **и сохранению** используемого порта.

Для изменения как ключа шифрования, так и выбора нового свободного порта предназначен флаг команды `--port` со значением `random`:

```
> zoaide link settings reset --port random
```

Вызов данной команды приводит к созданию **нового** ключа шифрования **и выбору нового** свободного порта.

Для вывода синописа команды изменения настроек связи предназначен флаг `--help`:

```
> zoaide link settings reset --help
```

## 4.8 Установка дополнительных плагинов

Приложение Помощника Администратора Зодиак.АйТиЭм содержит в своем инсталляционном дистрибутиве набор плагинов, который может быть расширен или сужен после его установки.

Любой плагин может быть установлен в следующие области (*англ. scope*):

- **глобальная область** (*англ. global scope*) – содержит плагины, предназначенные для использования всеми УЗ на АРМ;
- **пользовательская область** (*англ. user scope*) – содержит плагины, предназначенные для использования конкретной УЗ на АРМ.

### Примечание

Пользовательская область имеет более высокий приоритет, чем глобальная. Если плагин с одним и тем же именем установлен как в пользовательскую область, так и в глобальную – при загрузке приложения Помощника будет загружена единственная копия плагина из пользовательской области.

Расположения устанавливаемых плагинов по-умолчанию приведены в таблице 4.9:

Область	ОС	Расположение плагинов
Глобальная	Windows	C:\Program Files\Zodiac\AdminAide\plugins
	Linux	/opt/zodiac/admin-aide/plugins
Пользовательская	Windows	%LocalAppData%\Zodiac\AdminAide\plugins
	Linux	~/zodiac-admin-aide/plugins

Таблица 4.9. Расположения установки плагинов Помощника

### Предупреждение

Установка плагинов в глобальную область в большинстве конфигураций требует административных полномочий как для ОС семейства Windows, так и для ОС семейства Linux.

Для установки плагинов предназначены дистрибутивы плагинов – файлы с расширением «\*.zaap» (*англ. Zodiac Admin Aide Plugin*). Структурно такие файлы представляют собой архивы в формате «\*.tar.gz». При установке дистрибутива плагина происходит его распаковка в одно из расположений, указанных в таблице 4.9.

## ! Предупреждение

Не рекомендуется выполнять «ручную» распаковку дистрибутива плагина в указанные расположения, минуя описанные ниже штатные способы установки. При штатных способах установки выполняется контроль целостности дистрибутива плагина и коррекция прав доступа к извлекаемым файлам в зависимости от области установки.

Штатная установка плагина может быть выполнена одним из следующих способов:

- посредством консольной команды;
- посредством файловой ассоциации с расширением «\*.zaap».

### 4.8.1 Установка посредством консольной команды

Данный способ позволяет выполнить установку плагина как в пользовательскую, так и в глобальную область, а также задать дополнительные опции производимой установки. Установка плагина производится при помощи следующей команды:

```
> zoaide plugin install <путь_до_файла_заар>
```

Синописис команды можно получить, передав опцию **--help**:

```
> zoaide plugin install --help
Usage: zoaide plugin install [options] <file>

install plugin from file

Arguments:
  file          file to install

Options:
  -g, --global  install to global scope (default: false)
  --clear      clear old plugin data before installation (default: false)
  -y, --yes    proceed with installation without confirmation (default: false)
  -h, --help   display help for command
```

Команда поддерживает передачу следующих опций:

```
-g, --global
```

Задаёт глобальную область установки плагина. При отсутствии данной опции установка будет производиться в пользовательскую область текущей УЗ.

```
--clear
```

Задаёт признак очистки сохранённых данных прошлой установки плагина. Используйте данную опцию для получения заведомо «чистой» установки плагина.

```
-y, --yes
```

Задаёт режим «тихой» установки без дополнительного интерактивного подтверждения. Используйте данную опцию при автоматизации вызова команды.

## ! Предупреждение

Обращайте внимание на УЗ, от имени которой вызывается команда установки, и указываемую область установки.

Типовые **корректные** комбинации:

- a) установка от имени УЗ пользователя в пользовательскую область;
- b) установка от имени УЗ «root» в глобальную область;

Типовые **ошибочные** комбинации:

- a) установка от имени УЗ пользователя в глобальную область  
(недостаточно прав для записи в папку);
- b) установка от имени УЗ «root» в пользовательскую область  
(установленный плагин будет доступен только в сессии от имени УЗ «root»)

При вызове команды установки плагина (при условии отсутствия установленного плагина с тем же именем в той же области) без указания опции «тихой» установки в консоль будет выведено сообщение, показанное на рисунке 4.17. На приведенном рисунке отмечены следующие ключевые части сообщения:

### 1. Область установки

Значение «*user*» соответствует установке в пользовательскую область, значение «*global*» – в глобальную область.

### 2. Метаданные

Описывают имя, производителя и версию плагина.

### 3. Размер всех файлов без сжатия

Предназначен для оценки наличия достаточного свободного пространства для установки плагина.

```

astra-admin@astra-vm:~/Desktop/test$ zoaide plugin install ./test.zaap
Reading plugin installer: /home/astra-admin/Desktops/Desktop1/test/test.zaap... Done.
Searching for previously installed plugin version... Done.

The following plugin will be installed to user scope:
Name:      test
Vendor:    Unknown
Version:    1.0.0

After this operation, 3.4 kB of additional disk space will be used.
Proceed with installation [yes/no]?
    
```

Рисунок 4.17 Сообщение команды установки плагина при отсутствии другой установленной версии плагина

После вывода описанного сообщения команда установки ожидает подтверждение дальнейших действий от пользователя в виде ввода допустимого значения в область ввода (элемент 4 на рисунке 4.17). Для продолжения установки необходимо ввести значение «yes», для отмены установки – значение «no». Для подтверждения введенного значения используется клавиша «Ввод» (Enter).



### Совет

Поле ввода поддерживает функцию автоматического дополнения вводимых значений. Вместо ввода полного текста значения достаточно ввести его первую букву и вызвать автоматическое дополнение нажатием клавиши табуляции (Tab).

---

#### 4.8.2 Установка посредством файловой ассоциации

Данный способ установки предназначен для быстрой установки плагина в пользовательскую область путем двойного щелчка мышью в визуальном файловом проводнике на дистрибутив плагина Помощника с расширением «\*.zaar».

#### **!** Предупреждение

Данный способ не предназначен для установки плагинов в глобальную область. Для установки плагина в глобальную область воспользуйтесь альтернативным способом установки посредством консольной команды, описанным в разделе 4.8.1.

#### **i** Примечание

Фактически данный способ эквивалентен запуску консольной команды установки, описанной в разделе 4.8.1, без указания дополнительных опций.

При выполнении двойного щелчка мышью на файле с расширением «\*.zaar» будет выведено окно терминала по-умолчанию с сообщением, показанным на рисунке 4.18:

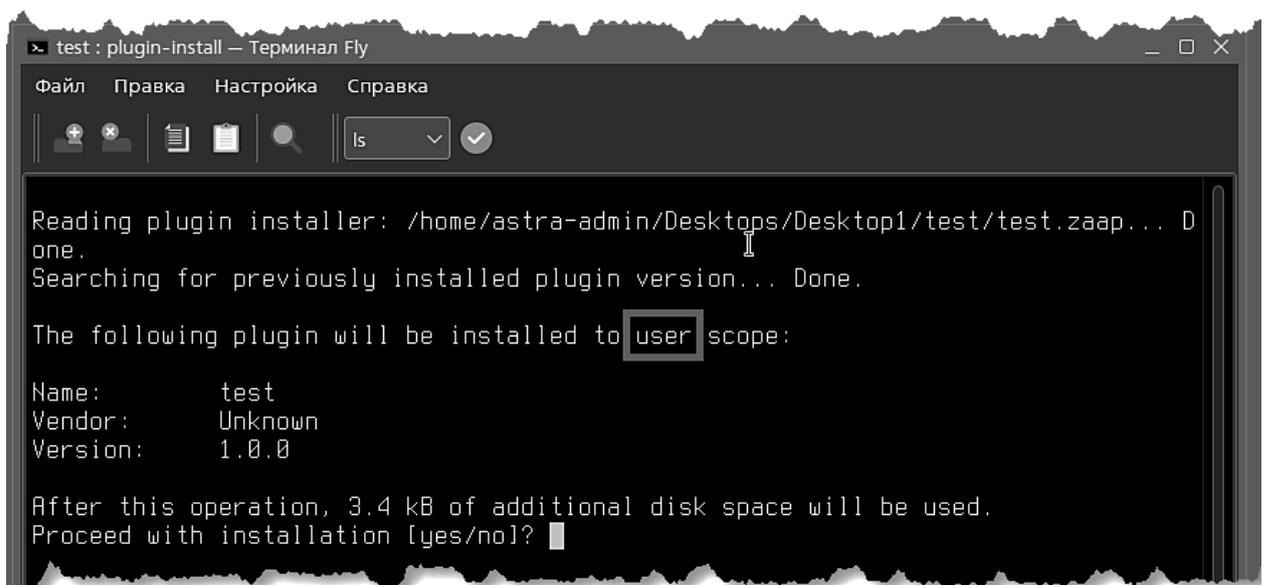


Рисунок 4.18 Окно терминала при установке плагина при отсутствии другой установленной версии плагина

Подтверждение дальнейших действий по установке полностью эквивалентно подтверждению консольной команды установки, описанной в разделе 4.8.1.

## 4.9 Изоляция плагинов

Приложение Помощника Администратора Зодиак.АйТиЭм является модульным и предполагает наполнение его функционала при помощи плагинов. Плагины, загружаемые в приложение Помощника, могут быть разработаны разными вендорами или Администратором самостоятельно. Эксплуатационные характеристики (стабильность, зрелость, отсутствие ошибок) разных плагинов могут различаться.

В процессе своей работы приложение Помощника опрашивает все плагины на предмет подходящих пунктов меню и выполняет агрегацию таких пунктов меню в единое дерево. В ходе такого опроса приложение Помощника выполняет автоматическую обработку ошибок, которой в большинстве случаев достаточно для сохранения стабильности работы. В случае, если какой-то плагин формирует ошибку – его пункты меню не будут отображаться в итоговом дереве пунктов меню. Описанное поведение выполняется автоматически и не требует дополнительной настройки.

Тем не менее, приложение Администратора технологически не может ограничить разработчиков плагинов в используемых подходах к разработке и системных вызовах, которые они выполняют из своих плагинов. Поэтому возможны ситуации, при которых отдельный плагин может дестабилизировать работу других плагинов и осложнить или сделать невозможным использование всего функционала Помощника.

---

### Пример 1

После установки нового или обновления существующего плагина наблюдаются задержки и нестабильность в появлении дерева пунктов меню. Длительное время отображается индикатор загрузки пунктов меню. Повторные вызовы меню Помощника для одного и того же элемента приводят к отображению различного состава пунктов меню.

---

### Пример 2

После установки нового или обновлении существующего плагина наблюдается нестабильность в появлении дерева пунктов меню. После перезапуска приложения Помощника в течение некоторого времени отображается корректное дерево пунктов меню, однако по прошествии времени перестают отображаться все пункты меню.

---

Для выявления проблемных плагинов, диагностики их работы и предотвращения дестабилизации всего функционала Помощника предназначен функционал **изоляции плагинов**.

Данный функционал позволяет создавать **зоны изоляции**, предназначенные для размещения в них одного или нескольких плагинов. Влияние плагинов, загруженных в разные зоны изоляции, друг на друга и на приложение Помощника минимизируется. Такие зоны изоляции в терминологии Помощника именуется **пулами плагинов** (*англ. plugin pool*).

---

### Предупреждение

Поддержание каждого дополнительного **пула плагинов** сопровождается увеличением нагрузки на ЦП и объема занимаемой ОП (~9 МБ на каждый пул).

---

Помощник Администратора Зодиак.АйТиЭм поддерживает как ручное управление количеством пулов и размещением в них плагинов, так и автоматическое (на основе политики).

### 4.9.1 Политика размещения плагинов в пулах

Политика размещения плагинов в пулах предназначена для задания общего принципа выделения пулов и размещения в них плагинов. Принцип размещения, диктуемый политикой, имеет более низкий приоритет и может быть переопределен при помощи ручного размещения плагинов в пулах, описанного в разделе 4.9.2. Таким образом, политика применяется только к тем плагинам, для которых не выполнено ручное присвоение конкретному пулу.

Задание политики производится путем изменения конфигурационного параметра приложения Помощника:

```
[Pool]
policy = single | vendor | each
```

Конфигурационный параметр *«policy»* поддерживает следующие значения:

- **single** – (по-умолчанию) размещать все плагины в единственном пуле;
- **vendor** – размещать плагины от разных вендоров в разные пулы;
- **each** – размещать каждый плагин в отдельном пуле.

---

#### Примечание

Для применения новой политики размещения плагинов в пулах необходимо выполнить перезапуск приложения Помощника.

---

### 4.9.2 Ручное размещение плагинов в пулах

Ручное размещение плагинов в пулах позволяет Администратору присвоить каждому плагину имя пула, в котором он должен размещаться. Приложение Помощника выполняет автоматический анализ заданных значений, создание нужного количества пулов и размещение в них плагинов.

---

#### Примечание

Если для разных плагинов *P1* и *P2* установлено одно и то же имя пула *N*, то приложение Помощника создаст один пул *N*, в который разместит оба плагина *P1* и *P2*.

---

Присвоить имя пула конкретному плагину возможно двумя способами:

1. При помощи режима «Помощник» веб-консоли Зодиак.АйТиЭм;
2. При помощи консольных команд.

### Присвоение имени пула при помощи режима «Помощник» веб-консоли

Данный способ позволяет выполнить присвоение или сброс имени пула при помощи визуального интерфейса веб-консоли Администратора Зодиак.АйТиЭм.



#### Совет

Присвоение имени пула при помощи веб-консоли применимо только при настроенной связи между веб-консолью и Помощником в соответствии с разделом 4.3. При отсутствии доступа к веб-консоли или ненастроенной связи с Помощником используйте консольные команды для присвоения имени пула.

Для получения информации о текущих пулах плагинов и ручного присвоения имен пулов предназначены колонки «Текущий пул» и «Назначенный пул» списка плагинов в режиме «Помощник» веб-консоли Зодиак.АйТиЭм, изображенные на рисунке 4.19:

Имя	Вендор	Область	Текущий пул	Назначенный пул	Вкл.
connections	Reference Point	Система	my1	my1	<input type="checkbox"/>
net-tools	Reference Point	Система	my1	my1	<input type="checkbox"/>
win-tools	Reference Point	Система	my2	my2	<input type="checkbox"/>
remote-assist-vnc	Reference Point	Пользователь	default		<input type="checkbox"/>
remote-script	Reference Point	Пользователь	default		<input type="checkbox"/>

Рисунок 4.19 Колонки «Текущий пул» и «Назначенный пул» списка плагинов

Колонка «Текущий пул» предназначена для отображения имени пула, в который в текущий момент времени загружен плагин. Данная колонка доступна только для чтения и носит информационный характер.

Колонка «Назначенный пул» предназначена для отображения и изменения имени пула, в который должен быть загружен плагин. Данная колонка доступна как для чтения, так и для изменения.

Пустое значение в колонке «Текущий пул» свидетельствует об одном из следующего:

- плагин выключен и не был загружен ни в один пул при запуске Помощника;
- произошла ошибка при загрузке плагина, и он не был загружен ни в один пул.

Пустое значение в колонке «*Назначенный пул*» свидетельствует о том, что для данного плагина не выполнено ручное присвоение имени пула и выбор пула для его размещения определяется политикой, описанной в разделе 4.9.1. Имя пула, выбранного в соответствии с политикой, отображается в колонке «*Текущий пул*».

---

### **Примечание**

Размещение плагинов в пулах производится приложением Помощника при запуске, поэтому для применения изменений в колонке «*Назначенный пул*» необходимо выполнить перезапуск приложения Помощника.

---

При выявлении несоответствия значений в колонках «*Текущий пул*» и «*Назначенный пул*» для включенных плагинов режим «Помощник» веб-консоли Зодиак.АйТиЭм отображает информационное сообщение о необходимости перезапуска приложения Помощника для применения изменений, изображенное на рисунке 4.20:

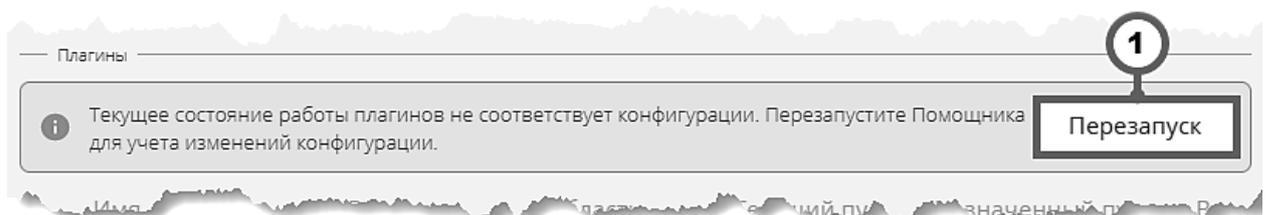


Рисунок 4.20 Информационное сообщение о необходимости перезапуска приложения Помощника

Для выполнения перезапуска нажмите на кнопку «*Перезапуск*» в области информационного сообщения (элемент 1 на рисунке 4.20) или воспользуйтесь пунктом контекстного меню «*Перезапустить приложение*» приложения Помощника в системном трее, (элемент 1 на рисунке 4.8).

---

### **Совет**

Для обновления информации о текущем состоянии плагинов после перезапуска приложения Помощника воспользуйтесь кнопкой «*Обновить*» в области инструментов основного окна режима «Помощник» (элемент 1 на рисунке 4.9).

---

## Присвоение имени пула при помощи консольных команд

Данный способ позволяет выполнить присвоение или сброс имени пула при помощи консольных команд приложения Помощника:

- присвоение имени пула для плагина в пользовательской области:

```
> zoaide plugin pool assign <имя_плагина> <имя_пула>
```

- присвоение имени пула для плагина в пользовательской области:

```
> zoaide plugin pool assign -g <имя_плагина> <имя_пула>
```

- сброс имени пула для плагина в пользовательской области:

```
> zoaide plugin pool clear <имя_плагина>
```

- сброс имени пула для плагина в глобальной области:

```
> zoaide plugin pool clear -g <имя_плагина>
```

Для вывода синописа указанных команд предназначена опция «*--help*», доступная для указания как для группы «*plugin pool*», так и для команд «*clear*» и «*assign*»:

```
> zoaide plugin pool --help  
> zoaide plugin pool assign --help  
> zoaide plugin pool clear -help
```

---

### Примечание

Размещение плагинов в пулах производится приложением Помощника при запуске, поэтому для применения изменений в колонке «Назначенный пул» необходимо выполнить перезапуск приложения Помощника.

Для выполнения перезапуска приложения Помощника, воспользуйтесь одним из визуальных способов перезапуска, указанных в прошлом разделе, или консольной командой:

```
> zoaide restart
```

## 4.10 Справка по плагинам

Плагины Помощника Администратора Зодиак.АйТиЭм могут содержать в своем составе справку, описывающую назначение, функциональность, особенности конфигурации и другие аспекты работы с плагином.

---

### Примечание

Наличие или отсутствие справки по плагину определяется исключительно разработчиком плагина.

Ведение справки плагинов осуществляется в формате Markdown (\*.md). Данный формат является текстовым с поддержкой разметки в виде дополнительных печатных символов непосредственно в тексте. Документы в данном формате остаются читаемыми при открытии

в любом текстовом редакторе / просмотрщике, однако более удобны для чтения при использовании специализированных средств, поддерживающих данный формат, например Sublime Text, Notepad++ или Visual Studio Code.

Имя файла справки плагина фиксированное:

```
<расположение_установки_плагинов>/<имя_плагина>/readme.md
```

Расположения установки плагинов по-умолчанию в зависимости от семейства ОС и области установки приведены в таблице 4.9.

Для быстрой навигации к файлу справки конкретного плагина предназначен пункт контекстного меню списка плагинов «Показать файл readme», приведенный на рисунке 4.21:

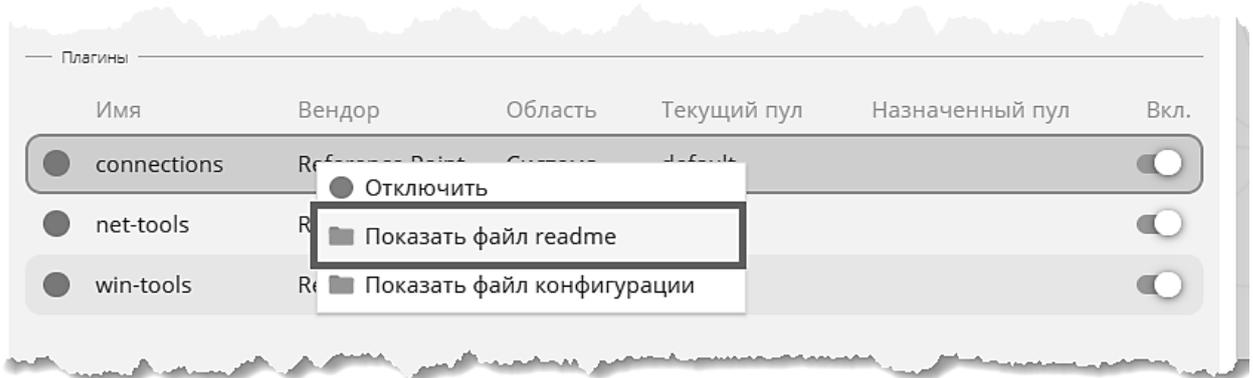


Рисунок 4.21 Пункт контекстного меню для быстрой навигации к файлу справки плагина

Вызов данного пункта меню приводит к открытию родительской папки файла справки «*readme.md*» выбранного плагина в визуальном файловом проводнике ОС. Открытие файла в требуемом редакторе необходимо выполнить вручную.

---

### Примечание

Если разработчиком плагина не предусмотрена справка, пункт контекстного меню «Показать файл readme» будет отсутствовать.

---

## 4.11 Конфигурация плагинов

Плагины Помощника Администратора Зодиак.АйТиЭм могут поддерживать конфигурирование параметров своей работы.

Инфраструктура плагинов Помощника предоставляет для разработчиков плагинов удобный и обобщенный механизм для работы с конфигурацией. Данный механизм выполняет:

- автоматическое создание **папки хранения данных** для каждого плагина;
- автоматическое создание **конфигурационного файла плагина** из **шаблона**, созданного разработчиком;
- функции по чтению и сохранению конфигурационного файла из плагина.

### Примечание

Описываемый механизм конфигурирования носит рекомендательный характер. Разработчик плагина может использовать при реализации своего плагина дополнительные или альтернативные способы конфигурации.

**Папка хранения данных** – это папка на файловой системе компьютера, предназначенная для использования плагином в процессе своей работы. Эта папка может содержать конфигурационный файл плагина, временные файлы плагина, создаваемые в процессе его работы, и другие элементы, предусмотренные разработчиком плагина. Папка хранения данных плагина создается для каждой УЗ компьютера: таким образом, разные пользователи одного и того же компьютера при использовании одного и того же плагина могут задать различные конфигурационные параметры, а различные копии плагина (работающие от имени разных УЗ) не войдут в конфликт. Папка хранения данных плагина по-умолчанию не удаляется при удалении и обновлении плагина: таким образом, созданная пользователем конфигурация сохраняется и переиспользуется в процессе жизненного цикла плагина.

Расположения папок хранения данных плагинов по-умолчанию приведены в таблице 4.10:

Область	ОС	Папка хранения данных плагина
Глобальная	Windows	%LocalAppData%\Zodiac\AdminAide\data\global\ <имя_плагина>
	Linux	~/zodiac-admin-aide/data/user/<имя_плагина>
Пользовательская	Windows	%LocalAppData%\Zodiac\AdminAide\data\user\ <имя_плагина>
	Linux	~/zodiac-admin-aide/data/user/<имя_плагина>

Таблица 4.10. Расположения папок хранения данных плагинов Помощника

### Совет

В случае выявления проблем в работе плагина, особенно в случаях недавнего обновления его версии, рекомендуется удалить его папку хранения данных и перезапустить приложение Помощника. Это действие приведет к началу работы плагина в режиме «первой установки».

Удаление папки хранения данных плагина можно произвести как вручную, так и при помощи консольной команды очистки данных плагина:

для **пользовательской** области:

```
> zoaide plugin clear <имя_плагина>
```

для **глобальной** области:

```
> zoaide plugin clear -g <имя_плагина>
```

---

**Конфигурационный файл плагина** – это файл в формате *\*.ini*, расположенный в папке данных плагина. Имя файла фиксировано в значении «*config.ini*». Конфигурационные файлы плагинов поддерживают секции в формате:

```
[<имя_секции>]
<имя_параметра_1> = <значение_параметра_1>
<имя_параметра_2> = <значение_параметра_2>
```

и комментарии в формате:

```
# Комментарий
# <закомментированный_параметр> = <значение>
```

**Шаблон конфигурационного файла плагина** – это файл в формате *\*.ini*, расположенный в папке установки плагина. Имя файла фиксировано в значении «*config.template.ini*». При запуске любого плагина и отсутствии конфигурационного файла в папке хранения данных этого плагина Помощник выполняет копирование шаблона конфигурационного файла в папку хранения данных под именем «*config.ini*».

### Совет

В случае необходимости возврата конфигурационного файла плагина к значениям по умолчанию рекомендуется удалить или переименовать существующий файл конфигурации и перезапустить приложение Помощника. Это действие приведет к созданию нового файла конфигурации из шаблона. Данное действие может быть полезно при обновлении версии плагина и изменении (расширении) конфигурационных параметров, поддерживаемых новой версией.

---

Для быстрой навигации к файлу конфигурации конкретного плагина предназначен пункт контекстного меню списка плагинов «Показать файл конфигурации», приведенный на рисунке 4.22:

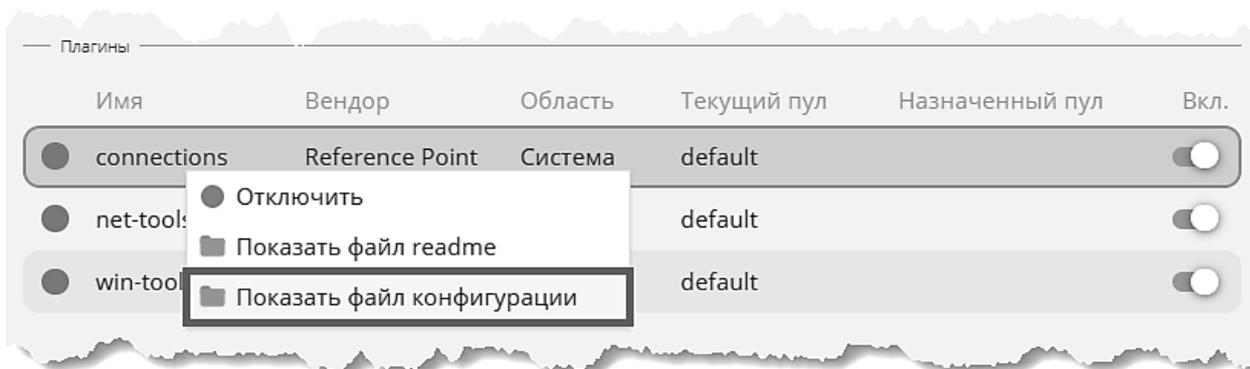


Рисунок 4.22 Пункт контекстного меню для быстрой навигации к файлу конфигурации плагина

### Примечание

Если разработчиком плагина не предусмотрено конфигурирование параметров работы, пункт контекстного меню «Показать файл конфигурации» будет отсутствовать.

Альтернативным способом навигации к конфигурационному файлу плагина является консольная команда Помощника:

- для плагина в пользовательской области:

```
> zoaide plugin configure <имя_плагина>
```

- для плагина в глобальной области:

```
> zoaide plugin configure -g <имя_плагина>
```

Вызов указанных команд и пункт контекстного меню приводит к открытию папки расположения конфигурационного файла плагина в визуальном файловом проводнике ОС. Открытие файла в требуемом редакторе необходимо выполнить вручную.

## 4.12 Журналы плагинов

Плагины Помощника Администратора Зодиак.АйТиЭм могут поддерживать журналирование (логирование) различных аспектов своей работы.

Инфраструктура плагинов Помощника предоставляет для разработчиков плагинов удобный и обобщенный механизм журналирования.

---

### Примечание

Описываемый механизм журналирования носит рекомендательный характер. Разработчик плагина может использовать при реализации своего плагина дополнительные или альтернативные способы журналирования.

---

Механизм журналирования плагинов осуществляет запись сообщений от плагинов в те же самые текстовые журналы, что и само приложение Помощника (подробности журналирования приложения Помощника приведены в разделе 4.6). Для облегчения поиска и улучшения дифференциации сообщений от различных плагинов и сообщений от приложения Помощника осуществляется автоматическое добавление префикса с именем и версией плагина ко всем сообщениям, формируемым плагинами:

```
2023-12-01T11:31:14.016Z INFO [21852] [connections @ 1.6.5]: Refresh clients
2023-12-07T10:03:07.939Z INFO [25316] [remote-assist-vnc @ 1.6.4]: Success on refresh
```

Поскольку журналирование сообщений от плагинов осуществляется в текстовые журналы приложения Помощника, то и регулирование подробности ведения журналов, предельного размера и предельного количества файлов журнала осуществляется той же самой конфигурационной секцией [Log], описанной в разделе 4.6.

## 4.13 Создание плагинов

Приложение Помощника разработано с использованием *NodeJS* – открытой кроссплатформенной среды исполнения для языка *ECMAScript* (более широко известного под названием *JavaScript*). Создание плагинов для приложения Помощника также производится на языке *ECMAScript*. Данный язык является интерпретируемым (скриптовым, не требующим компиляции).

---

### Примечание

Для языка *ECMAScript* существует большое количество производных языков, использующих отличный синтаксис и предоставляющих дополнительные удобства в процессе разработки: *TypeScript*, *CoffeeScript*, *PureScript* и др. Создание плагинов для приложения Помощника может производиться с использованием перечисленных языков.

Однако, инструментарий и процедура настройки подобной разработки значительно сложнее в освоении, требует навыков разработчика и, зачастую, избыточна для функционала, реализуемого в рамках плагинов.

Общей рекомендацией является начало разработки любого плагина с использованием языка *ECMAScript* и переход к использованию отличных языков только при накоплении существенной сложности и объема кодовой базы.

---

В своей сути создание плагина представляет собой написание *ECMAScript*-скрипта и его размещение в предусмотренной Помощником папке. Изменение и отладка плагина в простейшем случае представляют собой изменение текста скрипта и перезапуск приложения Помощника для применения изменений.

Расширенные сценарии, встречающиеся при создании плагинов, включают в себя добавление иконок плагина, включение в состав плагина скриптов на других языках (например, *BASH* или *PowerShell*), запарковка плагина в дистрибутив плагина (\*.zaap).

Инструментарий для создания плагинов Помощника Администратора Зодиак.АйТиЭм включен непосредственно в инсталляционный дистрибутив Помощника. Единственным дополнительным компонентом, требующимся для создания плагинов, является текстовый редактор.

---

### Совет

Несмотря на возможность использования произвольного текстового редактора для создания и редактирования плагинов, рекомендуется использовать специализированные редакторы исходных кодов, предоставляющие расширенные возможности по редактированию текстов на языке *ECMAScript*: подсветку синтаксиса, подсветку ошибок, интерактивные подсказки и документацию.

Наиболее мощным по функциям является кроссплатформенный редактор с открытым исходным кодом *VisualStudio Code*. Этот редактор также разработан на языке *ECMAScript* с использованием фактически совпадающего с Помощником технологического стека. Редакторы, обладающие меньшими возможностями, но тем не менее ориентированные на изменение текстов на языке *ECMAScript*, включают *SublimeText* и *Notepad++*.

---

#### 4.13.1 Размещение проекта плагина

Разработка плагина производится в *проекте плагина* – папке, содержащей составные части будущего плагина. Проект плагина может быть размещен:

- непосредственно в папке области `user` или `global`;
- в произвольной папке на файловой системе АРМ.



#### Совет

При выборе места размещения проекта плагина следует учитывать следующие факторы:

- при ведении разработки в произвольной папке удлиняется процесс тестирования и отладки плагина, поскольку после внесения изменений требуются дополнительные действия по переносу разрабатываемого плагина в папку области `user` или `global` для его загрузки в приложение Помощника;
- при ведении разработки в папке одной из областей усложняется процедура выпуска инсталляционного дистрибутива плагина (\*.заар) и его тестирования.

Общими советами по выбору места разработки являются:

- если плагин будет использоваться только на том АРМ, где он и разрабатывается, и только одной УЗ – вести разработку в папке области `user`;
- если плагин будет использоваться только на том АРМ, где он и разрабатывается, но разными УЗ – вести разработку в папке области `global`;
- если плагин будет выпускаться в виде инсталляционного дистрибутива плагина (\*.заар) и тиражироваться на разные АРМ – вести разработку в изолированной папке.

---

Для создания плагина в текущей папке предназначена команда:

```
> zoaide plugin create --name <имя_плагина>
```

Для создания плагина в заданной папке предназначена команда:

```
> zoaide plugin create --name <имя_плагина> --dir <путь_до_папки>
```

Для создания плагина в области `user` предназначена команда:

```
> zoaide plugin create --name <имя_плагина> --scope user
```

Для создания плагина в области `global` предназначена команда:

```
> zoaide plugin create --name <имя_плагина> --scope global
```

### **Внимание**

В именах плагинов допускается использование только строчных и заглавных латинских букв, цифр и символов – (дефис) и \_ (подчеркивание). Первым символом в имени плагина должна выступить строчная или заглавная латинская буква.

---

### **Совет 1**

В любой копии приложения Помощника имена плагинов должны быть уникальны. Таким образом, не поддерживается установка и одновременная работа разных плагинов с совпадающим именем.

Во избежание пересечения по имени с другими плагинами рекомендуется дополнять имя плагина уникальным префиксом, например, кратким наименованием Вашей организации (`contoso-имя_плагина`).

---

### **Совет 2**

При вызове команды создания плагина в текущей или заданной папке параметр `--name <имя_плагина>` может быть опущен (отсутствовать). В таком случае, в качестве имени создаваемого плагина будет использовано имя папки создания плагина.

---

### **Совет 3**

Для открытия папки созданного плагина в визуальном файловом проводнике сразу после его создания используйте флаг `--reveal`.

---

Все перечисленные варианты вызова команды `zoaide plugin create` приводят к созданию в папке начальной структуры проекта плагина. Отличие между вариантами состоит в выборе целевой папки.

### 4.13.2 Структура проекта плагина

Начальная структура проекта плагина состоит из следующих файлов и папок:

```
<КОРЕНЬ_ПРОЕКТА>           [ Папка ]
|
|-- plugin.json              [ Файл ]
|-- plugin.mjs               [ Файл ]
|
|-- config.template.ini      [ Файл ]
|-- readme.md                [ Файл ]
|
|-- .zapexec                 [ Файл ]
|-- .zapignore               [ Файл ]
|
|-- tsconfig.json            [ Файл ]
|-- @typings                  [ Папка ]
```

#### Файл plugin.json

Файл метаданных плагина в формате JSON. Предназначен для задания и модификации описательных характеристик плагина:

- *имени* – свойство "name";
- *версии* – свойство "version";
- *производителя* – свойство "vendor";
- *описания* – свойство "description".

#### Файл plugin.mjs

Основной скрипт плагина на языке ECMAScript. Предназначен для реализации логики работы плагина. В терминах среды NodeJS является ES6-модулем, о чем свидетельствует расширение \*.mjs.

#### Файл config.template.ini

Шаблон конфигурационного файла плагина в формате INI. Предназначен для задания общей структуры и значений по-умолчанию для конфигурационных параметров плагина. Более подробно особенности конфигурации плагинов описаны в разделе 4.11.

#### Файл readme.md

Файл справки плагина в формате Markdown. Предназначен для описания назначения и особенностей функционирования плагина. Более подробно особенности работы со справкой плагинов описаны в разделе 4.10.

#### Файл .zapexec

Файл маркировки файлов плагина как исполняемых для ОС семейства Linux в формате gitignore. Предназначен для указания файлов из состава проекта плагина, которые должны быть помечены как исполняемые (*file mode: executable*) на целевой АРМ после установки плагина. Используется только при упаковке плагина в инсталляционный дистрибутив

(\*.**заар**). При разработке и запуске плагина непосредственно из папки области плагинов задание соответствующего признака должно производиться вручную.

### Файл **.заарignore**

Файл игнорирования файлов при создании инсталляционного дистрибутива плагина (\*.**заар**) в формате `gitignore`. Предназначен для указания файлов из состава плагина, которые не должны включаться в инсталляционный дистрибутив плагина, например вспомогательных и временных файлов, используемых только на этапе разработки.

### Файл **tsconfig.json**

Файл настройки диалекта ECMAScript для ряда текстовых редакторов исходных кодов, в частности Visual Studio Code. Предназначен для указания редактору на особенности и вариации языковых конструкций и расположение папки тайпингов модулей, предоставляемых плагину.

### Папка **@typings**

Папка тайпингов модулей, доступных любому плагину, для ряда текстовых редакторов исходных кодов, в частности Visual Studio Code. Содержит документацию и правила использования таких модулей, оформленные специализированным образом, который поддерживают редакторы исходных кодов.

---

## **Примечание**

Из перечисленных файлов и папок, входящих в проект плагина, часть используется и в работе плагина, и при его разработке, а другая часть – только при его разработке.

Файлы, используемые и при разработке, и при работе плагина:

- **plugin.json**
- **plugin.mjs**
- **config.template.ini**
- **readme.md**

Файлы и папки, используемые только при разработке:

- **.заарехес**
- **.заарignore**
- **tsconfig.json**
- **@typings**

При создании инсталляционного дистрибутива плагина файлы **.заарехес** и **.заарignore** не включаются в него автоматически, а файл **tsconfig.json** и папка **@typings** по-умолчанию добавлены в файл **.заарignore**.

---

## Совет

В ходе разработки плагина предполагается изменение разных файлов из состава структуры проекта. Поэтому в используемом текстовом редакторе рекомендуется использовать режим открытия папки, а не отдельных файлов.

---

### 4.13.3 Работа с проектом плагина в редакторе Visual Studio Code

Для открытия папки проекта плагина в редакторе Visual Studio Code воспользуйтесь пунктом меню «Файл» → «Открыть папку» (англ. «File» → «Open Folder») как показано на рисунке 4.23:

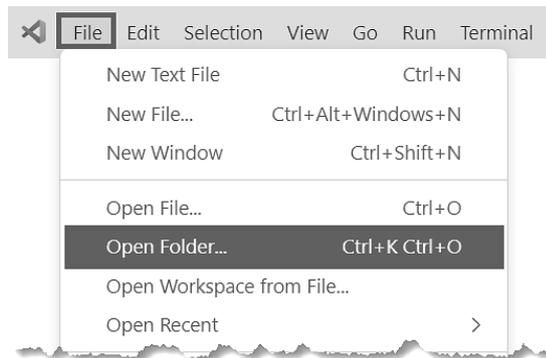


Рисунок 4.23 Пункт меню открытия папки в редакторе Visual Studio Code

Альтернативным способом открытия папки в редакторе Visual Studio Code является консольная команда `code`, поддерживающая указание пути до папки:

```
/home/my-plugin> code .  
/other-directory> code /home/my-plugin
```

Внешний вид основного окна редактора с открытой папкой проекта плагина приведен на рисунке 4.24:

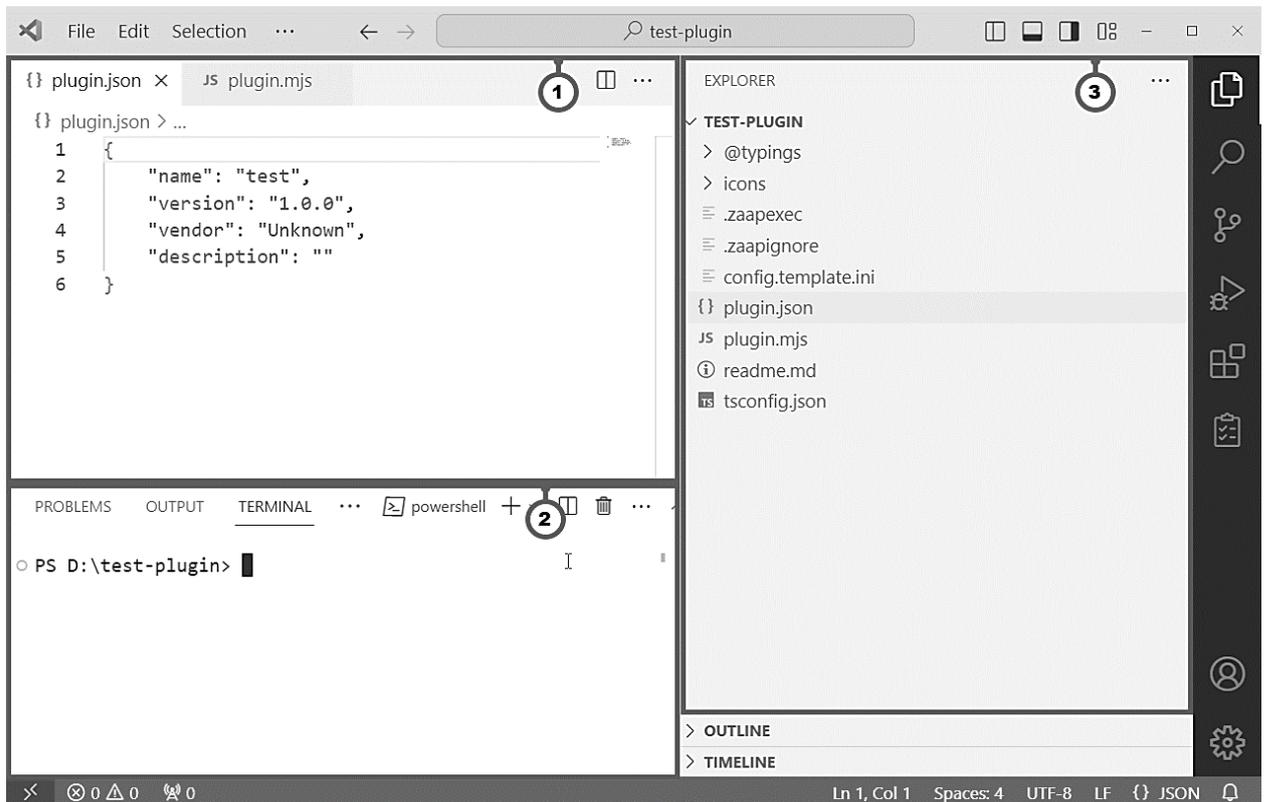


Рисунок 4.24 Внешний вид основного окна редактора Visual Studio Code с открытой папкой проекта плагина

На приведенном рисунке 4.24 обозначены следующие ключевые элементы управления:

### 1. Область редактирования файлов

Предназначена для отображения открытого (-ых) файла (-ов). В зависимости от расширения открываемого файла отображается содержимое файла в текстовом или визуальном виде. Используется для внесения изменений в файлы проекта.

### 2. Область терминала (консоли)

Предназначена для вызова консольных команд при работе с проектом. По-умолчанию в качестве рабочей папки используется папка проекта. Для открытия области терминала воспользуйтесь пунктом меню «Терминал» → «Новый Терминал» (англ. «Terminal» → «New Terminal»).

### 3. Область содержимого открытой папки

Предназначена для визуализации, создания, удаления и открытия файлов и папок из состава папки проекта.

Для открытия вкладки редактирования файла **без фиксации** в области редактирования файлов (1) используйте одиночное нажатие левой кнопки мыши. В таком случае при отсутствии изменений и выборе другого файла из области содержимого предыдущая вкладка будет автоматически закрыта, а вкладка выбранного файла – открыта. Данный

режим рекомендуется применять для изучения и проверки корректности содержимого файлов без внесения изменений.

Для открытия вкладки редактирования файла **с фиксацией** в области редактирования файлов (1) используйте двойное нажатие левой кнопки мыши. В таком случае даже при отсутствии изменений и выборе другого файла из области содержимого предыдущая вкладка останется открытой в области редактирования файлов.

### 4.13.4 Редактирование основных файлов плагина

Базовая разработка плагина включает в себя редактирование следующих основных файлов из состава проекта плагина:

#### 1. Файл `plugin.json`

Заполните или отредактируйте все поля, предусмотренные в файле: "name", "version", "vendor", "description". Все перечисленные поля являются текстовыми и их значение в соответствии с форматом JSON должно быть заключено в двойные кавычки: "<значение>".

#### 2. Файл `config.template.ini`

Внесите в файл конфигурационные параметры, влияющие на поведение создаваемого плагина, и их значения по-умолчанию. Для документирования параметров используйте строчные комментарии, задаваемые в соответствии с форматом INI символом #:

```
# Комментарий
[Секция]
Параметр1 = Значение1
Параметр2 = Значение2
```

При отсутствии параметров, влияющих на поведение плагина, зафиксируйте данный факт в виде строчного комментария:

```
# Данный плагин не поддерживает конфигурирование.
```

#### 3. Файл `readme.md`

Внесите в файл описание назначения плагина, особенностей его работы и конфигурации. При составлении файла для повышения читаемости используйте синтаксические конструкции формата Markdown, приведенные ниже:

```
# Заголовок уровня 1
## Заголовок уровня 2
Обычный текст, в котором используется *курсив* или **жирный текст**.
```ini
# Врезка содержимого на том или ином языке программирования/разметки
Параметр1 = Значение1
```
```

Подробное описание синтаксиса Markdown приведено на следующих интернет-ресурсах:

[Руниверсалис] <https://руни.рф/Markdown>

[Википедия] <https://ru.wikipedia.org/wiki/Markdown>

## Совет

Для открытия файла в формате Markdown в режиме визуализации в редакторе Visual Studio Code воспользуйтесь кнопкой открытия альтернативного представления (элемент 1 на рисунке 4.25) или комбинациями клавиш `Ctrl+K V` и `Ctrl+Shift+V`.



Рисунок 4.25 Кнопка открытия альтернативного представления в редакторе Visual Studio Code

Отображение файла в формате Markdown в режиме визуализации производится без вспомогательных символов и с расширенной разметкой, что облегчает его чтение и контроль вносимых изменений. Внешний вид основного окна Visual Studio Code с открытым файлом `readme.md` в режиме совмещенного просмотра приведен на рисунке 4.26 (1 – область редактирования, 2 – область визуализации):

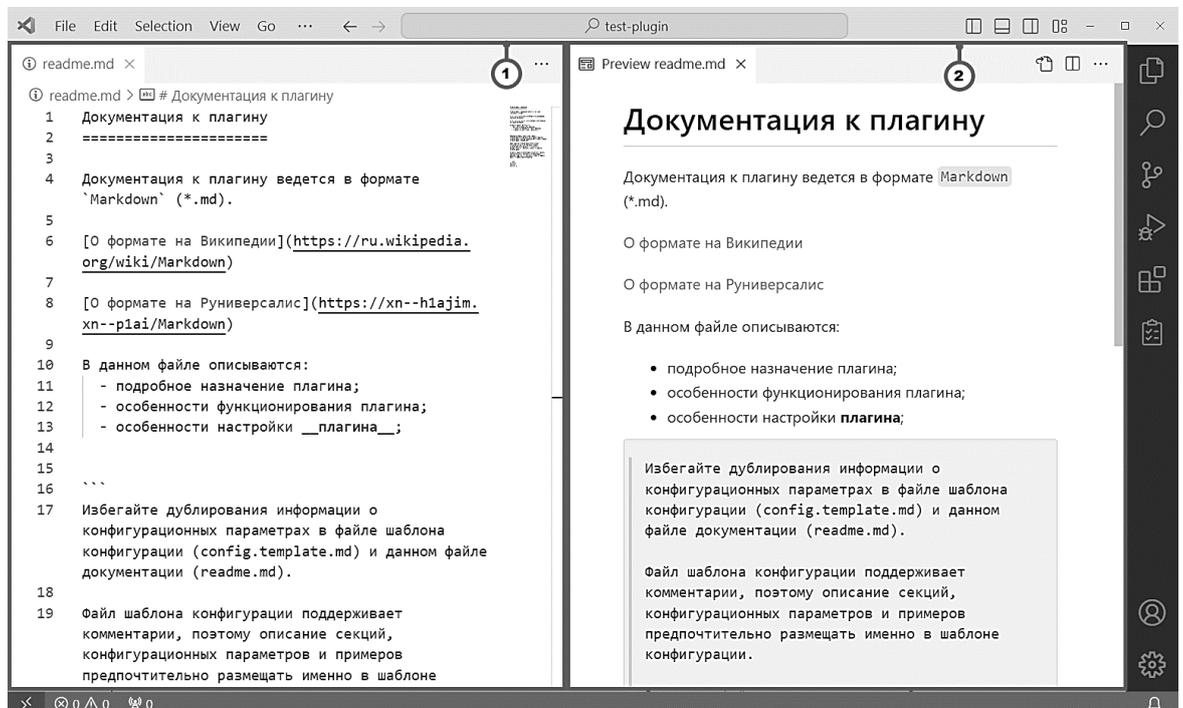


Рисунок 4.26 Отображение файла `readme.md` в совмещенном режиме редактирования/визуализации в редакторе Visual Studio Code

#### 4. Файл plugin.mjs

Реализуйте в данном файле логику формирования пунктов меню плагина и действий, выполняемых при выборе этих пунктов.

#### Примечание

Структура исходного кода плагина позволяет организовать *вариативное* появление пунктов меню плагина в зависимости от АРМ Администратора, на котором работает плагин, и от данных, для которых выполнен вызов Помощника: например, выводить пункт меню «Открыть *services.msc*» только если:

- a) плагин загружен в приложение Помощника на ОС семейства Windows  
т.к. запуск *services.msc* возможен только на ОС семейства Windows.
- b) Помощник вызван для управляемого компьютера под управлением ОС семейства Windows  
т.к. значение параметра `/computer=MyComputer1` должно соответствовать имени компьютера под управлением ОС семейства Windows.

Оформление плагина с подобной вариативностью является **предпочтительным**, т.к. позволяет избежать потенциальных ошибок при вызове заведомо неприменимых пунктов меню.

Однако, реализация описанной вариативности приводит к усложнению и увеличению объема кода плагина. Поэтому для задачи ознакомления с процедурой создания плагинов и изучения языка ECMAScript **допускается** упрощение логики плагина в части исключения вариативности. В таком случае оценка применимости пунктов меню плагина к текущему окружению и к данным выполняется пользователями такого плагина.

---

Создаваемый из шаблона проекта файл **plugin.mjs** уже содержит типовые примеры реализации логики плагина и использования библиотеки функций Помощника.

Первая строка созданного файла **plugin.mjs** содержит следующий комментарий:

```
// @ts-check
```

Как и любой другой комментарий в языке ECMAScript данный комментарий не влияет на логику работы плагина, а предназначен для сигнализации редакторам исходных кодов (в частности, Visual Studio Code), что для такого файла необходимо выполнять расширенную проверку на ошибки. Подобный комментарий должен быть **первой строкой файла**. При удалении этой строки редакторы исходных кодов выполняют лишь базовую проверку на ошибки.

Второй строкой в созданном файле является конструкция импорта функций и объектов из встроенного модуля `zodiac`:

```
import { provide, run, log } from 'zodiac'
```

Типичный синтаксис конструкции импорта приведен ниже:

```
import { <элемент 1>, <элемент 2> } from '<модуль 1>'
```

Подобные конструкции импорта могут быть множественные, причем допускается повторный импорт из одного и того же модуля:

```
import { provide, run } from 'zodiac'  
import { log } from 'zodiac'  
import { EOL } from 'os'
```

Все конструкции импорта должны предшествовать функциональным языковым элементам ECMAScript и находиться в начале файла.

Имена импортируемых элементов должны быть уникальны в рамках файла, который выполняет импорт. Для случаев, когда разные модули предоставляют элементы с одинаковыми именами, которые тем не менее необходимо различать и использовать в рамках одного и того же файла, предназначена конструкция импорта с псевдонимами:

```
import { <элемент 1> as <псевдоним 1> } from '<модуль 1>'
```

Например, при помощи этой конструкции можно организовать импорт функции `resolve` как из модуля `dns`, так и из модуля `path`.

```
import { resolve as resolveDNS } from 'dns'  
import { resolve as resolvePath } from 'path'
```

---

### Примечание 1

Задание псевдонима поддерживается для любого импортируемого элемента и может быть использовано в комбинации с использованием оригинального имени элемента, например:

```
import { resolve as resolvePath, join } from 'path'
```

---

### Примечание 2

При использовании псевдонимов в конструкциях импорта правило уникальности имен распространяется и на псевдонимы: имена всех импортируемых элементов или их псевдонимов должны быть уникальны в рамках файла.

---

### Примечание 3

В качестве имени модуля может выступать как встроенный модуль `zodiac`, так и любой из встроенных модулей среды NodeJS, так и внешние модули. Более подробная информация о встроенных и внешних модулях среды NodeJS содержится в официальной документации NodeJS уровня версии 12.x:

<https://nodejs.org/docs/latest-v12.x/api/index.html>

## Совет

Редактор Visual Studio Code поддерживает интерактивные подсказки для конструкций импорта, включающие как имена доступных модулей, так и элементы каждого из них.

Для вызова интерактивной подсказки с именами доступных модулей, добавьте заготовку конструкции импорта:

```
import {} from ''
```

Поместите курсор редактирования (например, одиночным нажатием кнопки мыши) в пространство между одинарными кавычками, в котором должно содержаться имя модуля, и нажмите комбинацию клавиш, отвечающую за вызов интерактивных подсказок `Ctrl+Space`. Пример интерактивной подсказки с именами модулей приведен на рисунке 4.27:

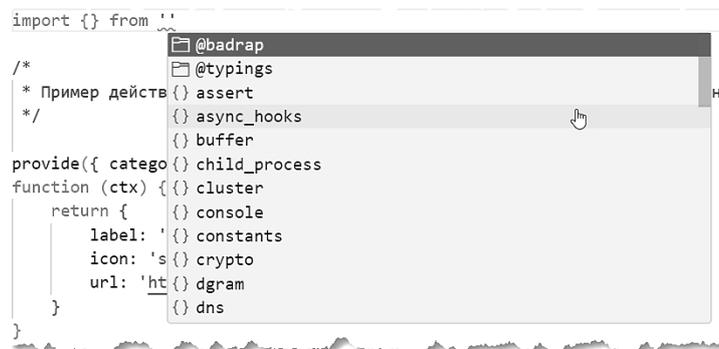


Рисунок 4.27 Интерактивная подсказка с именами модулей в редакторе Visual Studio Code

Для выбора требуемого модуля используйте одиночное нажатие левой кнопки мыши или клавишу `Ввод`.

После выбора или ввода имени модуля поместите курсор редактирования в пространство между фигурными скобками, в котором должны содержаться имена импортируемых элементов и нажмите комбинацию клавиш, отвечающую за вызов интерактивных подсказок `Ctrl+Space`. Пример интерактивной подсказки с элементами модуля `path` приведен на рисунке 4.28

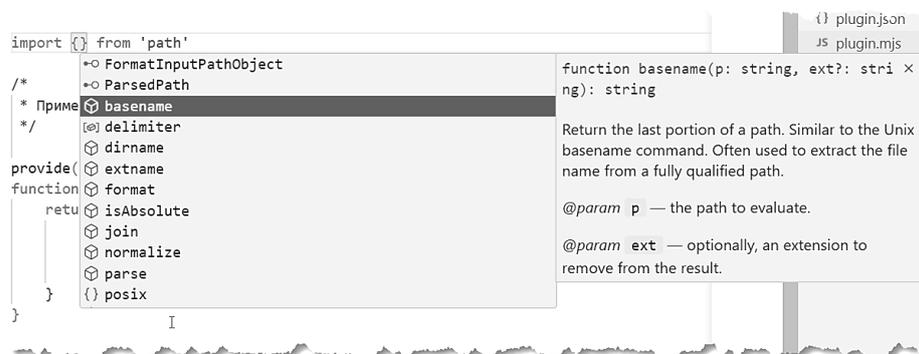


Рисунок 4.28 Интерактивная подсказка с элементами модуля `path` в редакторе Visual Studio Code

После конструкций импорта в тексте файла *plugin.mjs* следует функциональная логика плагина. Организация этой логики производится в виде функций, которым предшествует **конструкция обеспечения**:

```
provide({ category: 'hello', context: 'computer' }).items =  
function (ctx) { ... }
```

---

### Примечание

Конструкция обеспечения не является самостоятельной конструкцией языка ECMAScript. С точки зрения ECMAScript это – комбинация из вызова функции `provide`, разыменования ее результата по полю `items` и присвоение ему значения в виде новой функции.

Конструкция обеспечения описывается как отдельная конструкция в данном документе для облегчения понимания структуры файла **plugin.mjs** для разработчиков и Администраторов, не обладающих глубокими знаниями синтаксиса ECMAScript.

Для данной категории читателей приведенное выражение ECMAScript можно считать отдельной конструкцией, которую можно использовать и тиражировать в плагинах, не вдаваясь в технические особенности ее функционирования.

---

Конструкция обеспечения поддерживает изменение полей `category` и `context`:

#### Поле `category`

Предназначено для задания имени категории, в которую будут размещаться пункты меню, формируемые задаваемой функцией. Имена категорий не должны быть пусты и поддерживают произвольные символы в кодировке Unicode. Подробная информация о размещении функционала плагинов и понятии «*категория*» приведены в разделе 4.4.

#### Поле `context`

Предназначено для задания имени контекста, для которого предназначены пункты меню, формируемые задаваемой функцией. Поддерживаются значения `computer`, `neighbour` и `remote-connection`, соответствующие контекстам «*Компьютер*», «*Соседнее устройство*» и «*Сохраненное подключение*». Подробная информация о размещении функционала плагинов и понятии «*контекст*» приведены в разделе 4.4.

---

### Примечание

Перечень поддерживаемых значений поля `context` может расширяться в новых версиях приложения Помощника. Для получения актуального перечня воспользуйтесь интерактивной подсказкой редактора Visual Studio Code.

Объявление функции, следующее за конструкцией обеспечения, содержит единственный параметр `ctx` (сокр. англ. *context*). Объявленная функция будет вызываться на каждый вызов контекстного меню Помощника или меню «Помощник» из панели инструментов окна веб-консоли Зодиак.АйТиЭм. При этом в качестве параметра `ctx` в каждый вызов будет передаваться информация об элементе, для которого вызвано соответствующее меню.

---

### **Внимание**

Избегайте длительных операций в теле объявленной функции, особенно вызова функций с суффиксом `*Sync`. Чем дольше выполняется функция – тем дольше в веб-консоли отображается индикатор загрузки меню.

Если для обеспечения вариативности появления пунктов меню требуется выполнить длительную операцию (например, чтение файловой системы или сетевой запрос) – объявите корневую логическую переменную и организуйте ее регулярное обновление при помощи функции `setInterval`.

---

Основная задача объявленной функции – вернуть один или несколько пунктов меню. Простейшие варианты реализации такой функции приведены в создаваемом из шаблона проекта файле `plugin.mjs`.

Первый пример иллюстрирует функцию, возвращающую единственный пункт контекстного меню, соответствующий открытию ссылки в браузере:

```
provide({ category: 'hello', context: 'computer' }).items =
function (ctx) {
  return {
    label: 'Искать в Яндекс',
    icon: 'ssh.svg',
    url: 'https://yandex.ru/search/?text=' + ctx.hostname
  }
}
```

Пункт контекстного меню описывается объектом с полями `label`, `icon` и `url`:

#### **Поле `label`**

Соответствует имени пункта контекстного меню, отображаемого в веб-консоли.

#### **Поле `url`**

Соответствует адресу для открытия в браузере при выборе данного пункта меню.

#### **Поле `icon` (необязательное)**

Задаёт имя файла иконки, отображаемого для данного пункта меню. Файл с заданным именем должен быть размещен в папке `images` проекта плагина (отсутствует по-умолчанию).

---

### **Примечание**

При использовании пункта контекстного меню с полем `url`, открытие соответствующего адреса при выборе этого пункта меню производится автоматически, без повторного вызова плагина.

---

## Совет

Обратите внимание, что в приведенном примере в качестве поля `url` передается строка, составляемая из базовой части (`https://yandex.ru/search/?text=`) и вариативной части, меняющейся в зависимости от передаваемого параметра `ctx` (`ctx.hostname`). Для объединения строк используйте оператор `+` или шаблонные строки ECMAScript, например:

```
url: `https://yandex.ru/search/?text=${ctx.hostname}`
```

---

Второй пример иллюстрирует функцию, возвращающую единственный пункт контекстного меню, соответствующий запуску приложения `powershell.exe` с передачей в него команды печати имени компьютера, для которого было вызвано действие:

```
provide({ category: category, context: 'computer' }).items =
function (ctx) {
  return {
    label: 'Запустить PowerShell',
    action: () => {
      run(`powershell.exe -Command "Write-Host '${ctx.hostname}';
timeout /t -1"`)
    }
  }
}
```

---

## Совет

Обратите внимание, что функция `run` поддерживает вызов как с одним параметром:

```
run(
  `powershell.exe -Command "Write-Host '${ctx.hostname}'; timeout /t -1`
)
```

так и с множеством параметров:

```
run(
  'powershell.exe',
  [
    '-Command',
    `Write-Host '${ctx.hostname}'; timeout /t -1`
  ]
)
```

Оба способа позволяют достичь запуска требуемого приложения с передачей в него параметров, однако, второй способ является **рекомендованным**, поскольку позволяет избежать разночтений в разборе параметров и исключить необходимость в использовании символов `"` и `'` для обозначения границ параметров.

---

## Примечание

При использовании пункта контекстного меню с полем `action`, при выборе этого пункта меню в веб-консоли производится вызов функции, которая присвоена полю `action`.

В примере используется анонимная лямбда-функция, задаваемая синтаксисом:

```
action: () => { ... }
```

В качестве задаваемой функции также может быть использована анонимная функция с синтаксисом:

```
action: function() { ... }
```

или заранее объявленная именованная функция:

```
function foo() { ... }

provide({ category: category, context: 'computer' }).items =
function (ctx) {
  return {
    label: 'Запустить PowerShell',
    action: foo
  }
}
```

или сокращенный синтаксис функционального поля:

```
provide({ category: category, context: 'computer' }).items =
function (ctx) {
  return {
    label: 'Запустить PowerShell',
    action() { ... }
  }
}
```

## Совет

Обратите внимание, что в функцию, задаваемую в поле `action`, передаются контекст и параметры, которые могут быть использованы при ее реализации:

```
action: (ctx, params) => { ... }
```

```
action: function(ctx, params) { ... }
```

```
function foo(ctx, params) { ... }

provide({ category: category, context: 'computer' }).items =
function (ctx) {
  return {
    label: 'Запустить PowerShell',
    action: foo
  }
}
```

#### 4.13.5 Расширенные сценарии реализации плагинов

Инфраструктура плагинов приложения Помощника и среда исполнения NodeJS предоставляют широкий спектр встроенных функций и интерфейсов (API), которые могут быть использованы при реализации плагинов:

##### Чтение конфигурации

При необходимости изменения логики работы плагина в зависимости от конфигурационных параметров занесите поддерживаемые параметры и их значения по-умолчанию в файл **config.template.ini**, например:

```
[Search]
enable=true
```

Для чтения данного файла воспользуйтесь встроенной функцией `configurator.load`:

```
import { provide, configurator } from 'zodiac'

const config = configurator.load();

provide({ category: 'hello', context: 'computer' }).items =
function (ctx) {

  if (config.search.enable)
    return {
      label: 'Искать в Яндекс',
      icon: 'ssh.svg',
      url: 'https://yandex.ru/search/?text=' + ctx.hostname
    }
}
```

---

#### Совет

Обратите внимание, что для доступа к параметрам, задаваемым в конфигурационном файле вне секции (в корне файла) предназначен синтаксис:

```
const config = configurator.load();
config.myparam1
```

а для доступа к параметрам, задаваемым в рамках секции:

```
const config = configurator.load();
config.mysection1.myparam1
```

---

### Логирование

При необходимости записи в лог ошибок или других сообщений, формируемых при работе плагина, воспользуйтесь объектом `log`:

```
import { provide, run, log } from 'zodiac'

provide({ category: 'hello', context: 'computer' }).items =
function (ctx) {
  return {
    label: 'Записать в лог',
    action: (ctx) => { log.info(ctx.hostname) }
  }
}
```

Объект `log` содержит функции для всех поддерживаемых типов сообщений:

```
log.error(...)
log.warn(...)
log.info(...)
log.verbose(...)
log.debug(...)
```



### Совет

Обратите внимание, что первый параметр, передаваемый в функции объекта `log`, должен иметь строковый тип:

```
log.info(ctx.hostname)
log.error('Произошла ошибка')
```

Функции объекта `log` поддерживают передачу второго (диагностического, отладочного) параметра произвольного типа:

```
log.error('Произошла ошибка', err)
log.info('Информация о компьютере', ctx)
```

Такой второй параметр будет записан в формате JSON в лог вместе с текстом, задаваемым первым параметром.

---

## Выдача уведомлений

При необходимости формирования и отображения уведомления пользователю плагина воспользуйтесь функцией `notify`:

```
import { provide, run, notify } from 'zodiac'

provide({ category: 'hello', context: 'computer' }).items =
function (ctx) {
  return {
    label: 'Выдать уведомление',
    action: (ctx) => { notify(ctx.hostname) }
  }
}
```

---

### Совет

Обратите внимание, что функция `notify` поддерживает как строковую, так и объектную форму задания параметра:

```
notify('Привет, мир');
notify({
  type: 'info',
  text: 'Привет, мир'
});
```

Использование объектной формы позволяет помимо текста уведомления задать и тип уведомления. Поддерживаемые типы уведомлений:

```
'none'
'info'
'error'
'warn'
```

---

### Примечание

Конфигурация приложения Помощника позволяет отключить уведомления, формируемые плагинами, при помощи параметра `plugins` секции `[Notifications]`:

```
[Notifications]
plugins = false
```

При отключении уведомлений плагинов вызовы функции `notify` не будут приводить к отображению уведомлений, при этом функция будет завершаться корректно, без формирования ошибки.

---

### Пункты контекстного меню с опциями

По мере развития и усложнения создаваемого плагина типичной ситуацией является рост количества его пунктов меню. Большое количество пунктов меню усложняет использование функционала Помощника, поскольку заставляет пользователя плагина выполнять поиск требуемого пункта меню в обширном списке.

Одним из способов улучшения опыта использования таких плагинов является разбиение пунктов меню по разным категориям (папкам) с тем, чтобы количество пунктов меню в каждой категории не превышало приемлемое значение: максимум 10-15 пунктов.

Другим способом является **параметризация пунктов меню**. Данный способ позволяет организовать пункт меню, при вызове которого из веб-консоли пользователю будет отображено окно **опций**, в котором он сможет выбрать требуемую вариацию действия. Пример окна опций для плагина connections приведен на рисунке 4.29:

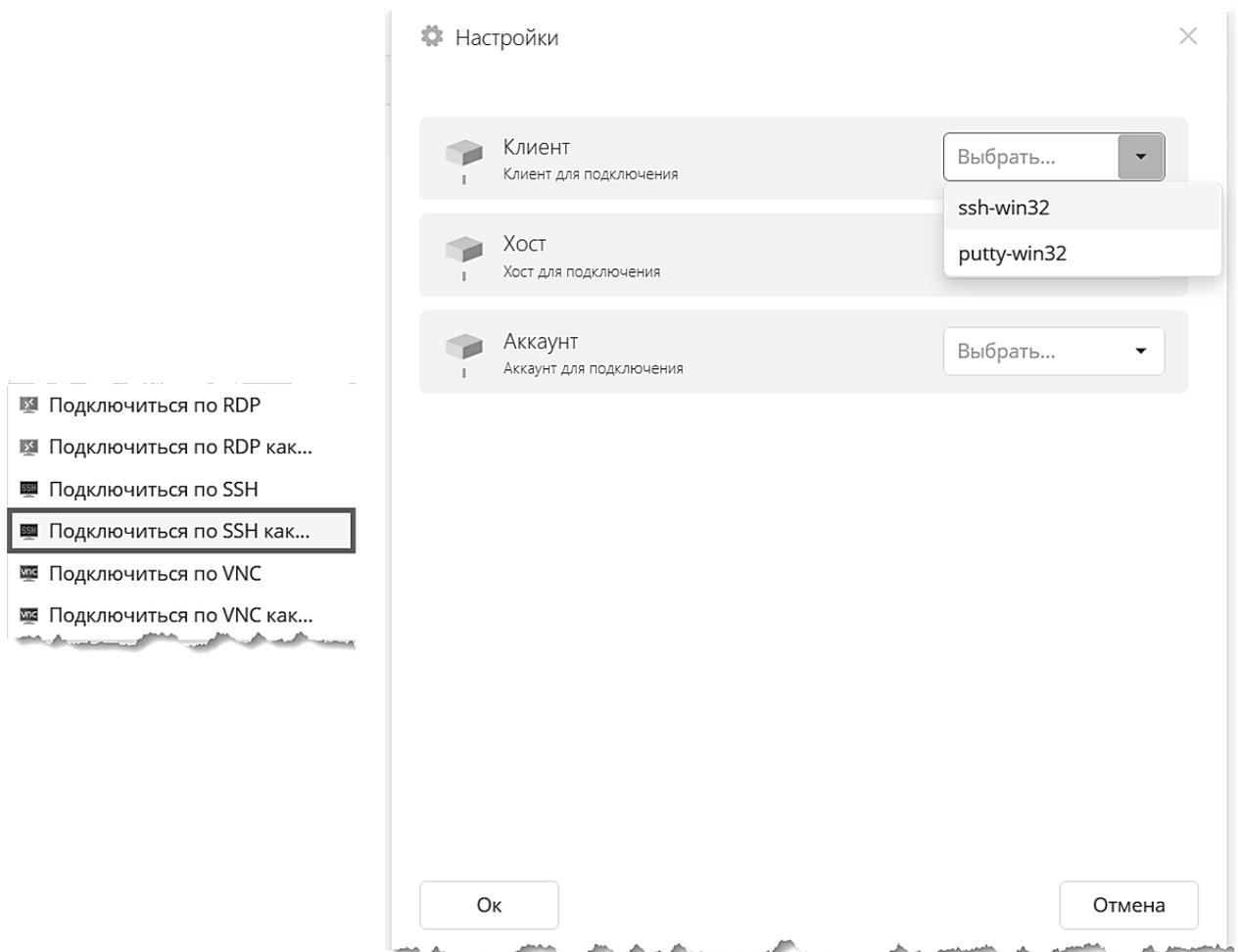


Рисунок 4.29 Окно опций пункта меню «Подключиться по SSH как...» плагина connections

## Примечание

Использование параметризованных пунктов меню позволяет существенно сократить общее количество пунктов меню. Например, если поведение пункта меню регулируется  $N$  параметрами, каждый из которых может иметь  $M$  значений, то общее количество вариантов (непараметризованных пунктов меню) равно  $M^N$ .

Уже для значений  $N=3$  и  $M=3$  получившееся количество равняется 27, что превышает приемлемое количество пунктов меню в одной категории.

Для указания опций пункта меню предназначено поле `opts` объекта пункта меню:

```
provide({ category: 'hello', context: 'computer' }).items =
function (ctx) {
  return {
    label: 'Пункт с опциями',
    opts: {
      'Моя опция 1': ['Значение 1', 'Значение 2'],
      'Моя опция 2': ['Значение 1', 'Значение 2']
    },
    action: (ctx, params) => { log.info('Выбраны опции', params.opts) }
  }
}
```

В приведенном примере в качестве значения поля `opts` передан объект-словарь, в котором ключами являются имена опций, отображаемых пользователю, а значениями – массивы из возможных значений каждой опции.

Для сопровождения опций описаниями воспользуйтесь объектной формой их задания:

```
provide({ category: 'hello', context: 'computer' }).items =
function (ctx) {
  return {
    label: 'Пункт с опциями',
    opts: {
      option1: {
        label: 'Моя опция 1',
        description: 'Описание 1',
        values: ['Значение 1', 'Значение 2']
      },
      option2: {
        label: 'Моя опция 2',
        description: 'Описание 2',
        values: ['Значение 1', 'Значение 2']
      }
    },
    action: (ctx, params) => { log.info('Выбраны опции', params.opts) }
  }
}
```

Выбранные пользователем значения опций будут переданы в функцию поля `action`. Для доступа опциям, заданным в короткой форме, используйте синтаксис:

```
params.opts['Моя опция 1']
```

а к опциям, заданным в объектной форме синтаксис:

```
params.opts.option1
```

### Асинхронная логика

Инфраструктура плагинов приложения Помощника и среда исполнения языка ECMAScript NodeJS ориентированы на исполнение **асинхронной логики**. Оформление логики плагина в асинхронной манере позволяет обеспечить эффективное одновременное исполнение программной логики всех плагинов, загруженных в приложение Помощника.

Большинство встроенных функций как Помощника, так и среды NodeJS организованы в асинхронной манере и предполагают их использование из асинхронного кода.

---

### Примечание

Простейшие сценарии, приведенные ранее, были организованы в виде синхронного кода. В том числе, в приведенных примерах осуществлялся вызов асинхронных функций (например, функции `notify`) из синхронной функции поля `action`.

Подобное использование асинхронных функций из синхронного кода не является синтаксической ошибкой и не запрещено, но во многих случаях приводит не к ожидаемому результату.

Необходимо понимать, что любой вызов асинхронной функции без ее ожидания (например, из синхронной функции) приводит к исполнению вызванной функции в фоновом режиме, а следующая инструкция начнет исполнение до полного завершения предыдущей.

Более того, порядок исполнения нескольких последовательно вызванных асинхронных функций может быть перемешан, например в следующем коде

```
import { promises as fs } from 'fs'

fs.mkdir('/tmp/mydir/')
fs.writeFile('/tmp/mydir/myfile', 'Hello')
```

используется последовательный вызов функции создания папки (`fs.mkdir`) и записи в файл в созданной папке `fs.writeFile`. Обе функции являются асинхронными, но вызываются в синхронной манере, поэтому несмотря на правильный порядок вызовов (сначала создание папки, потом – запись в файл), реальный порядок исполнения файловых операций в фоне может быть обратным, что приведет к ошибке.

---

Основными конструкциями языка ECMAScript для организации асинхронной логики являются ключевые слова `async` и `await`.

Ключевое слово `async` используется при объявлении функции для ее пометки как асинхронной. При отсутствии ключевого слова `async` объявленная функция считается синхронной:

```
// Синхронная функция в конструкции обеспечения
provide({ category: category, context: 'computer' }).items =
function (ctx) { ... }

// Асинхронная функция в конструкции обеспечения
provide({ category: category, context: 'computer' }).items =
async function (ctx) { ... }
```

Объявление асинхронной функции может встречаться в произвольных местах кода, в том числе в теле синхронной функции:

```
// Функция в конструкции обеспечения - синхронная.
provide({ category: 'hello', context: 'computer' }).items =
function (ctx) {
  return {
    label: 'Синхронное действие',

    // Функция действия - синхронная
    action(ctx) { log.info('Выбраны опции', params.opts) }

  }
}

// Функция в конструкции обеспечения - синхронная.
provide({ category: 'hello', context: 'computer' }).items =
function (ctx) {
  return {
    label: 'Асинхронное действие',

    // Функция действия - асинхронная.
    async action(ctx) { log.info('Выбраны опции', params.opts) }

  }
}
```

---

### Совет

В подавляющем большинстве ситуаций асинхронное объявление функции используется для функций в поле `action`, а функции в конструкции обеспечения не содержат асинхронной логики и не требуют асинхронного объявления.

---

Ключевое слово `await` используется в теле асинхронных функций и предназначено для ожидания полного завершения вызываемых асинхронных функций:

```
import { provide, log } from 'zodiac'
import { promises as fs } from 'fs'

provide({ category: 'hello', context: 'computer' }).items =
function (ctx) {
  return {
    label: 'Запись в файл',
    async action(ctx) {
      await fs.mkdir('/tmp/mydir/')
      await fs.writeFile('/tmp/mydir/myfile', 'Hello')
      log.info('Файл записан')
    }
  }
}
```

В приведенном примере ключевое слово `await` используется для ожидания завершения операции создания папки (`fs.mkdir`) перед записью в файл в этой папке (`fs.writeFile`).

### Примечание 1

Обратите внимание, что вызов синхронных функций (например, функции `log.info`) из тела асинхронной функции не требует использования ключевого слова `await` для ожидания завершения этих вызовов.

---

### Примечание 2

Разделение функций на синхронные и асинхронные и использование ключевого слова `await` для ожидания асинхронных функций призвано не усложнить написание кода на языке ECMAScript, а открыть возможность реализации сложных сценариев с параллелизмом, например одновременной записи в два разных файла с ожиданием завершения обеих операций при помощи функции `Promise.all`:

```
async action(ctx) {
  await fs.mkdir('/tmp/mydir/')
  await Promise.all([
    fs.writeFile('/tmp/mydir/myfile1', 'Hello'),
    fs.writeFile('/tmp/mydir/myfile2', 'Hello')
  ]);
}
```

В приведенном коде выполняется ожидание создания общей родительской папки для файлов и единое ожидание двух параллельных фоновых операций записи в файлы **myfile1** и **myfile2**. При записи большого объема данных подобный подход позволяет завершить все тело функции быстрее, чем отдельное ожидание каждой операции:

```
async action(ctx) {
  await fs.mkdir('/tmp/mydir/')
  await fs.writeFile('/tmp/mydir/myfile1', 'Hello')
  await fs.writeFile('/tmp/mydir/myfile2', 'Hello')
}
```

---

## Совет

Для определения вида вызываемой функции (синхронная или асинхронная) воспользуйтесь интерактивными подсказками редактора Visual Studio Code. При наведении курсора на вызов функции редактор отображает всплывающее окно с описанием и сигнатурой функции, приведенное на рисунке 4.30.

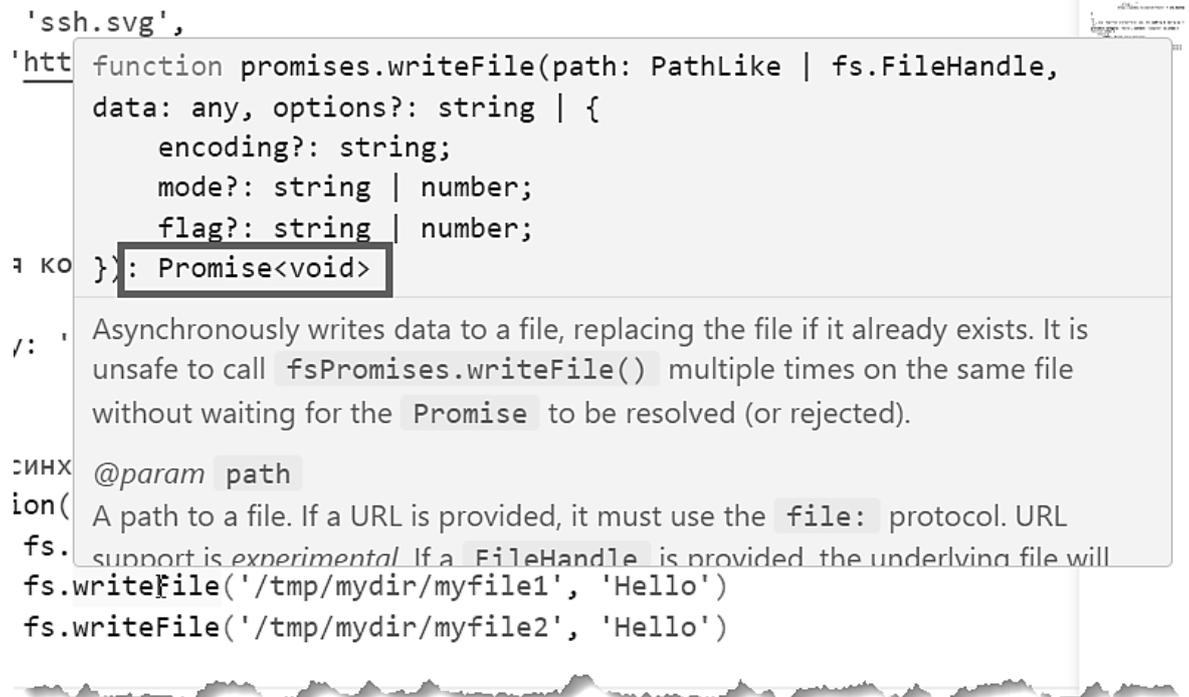


Рисунок 4.30 Всплывающее окно с документацией и сигнатурой функции writeFile

Если возвращаемое значение функции значится как `Promise<...>` - используемая функция асинхронная и, в большинстве случаев требует использования ключевого слова `await` при ее вызове.

### Возврат сообщений из действий пунктов контекстного меню

Функции, задаваемые как значение поля `action` поддерживают возврат сообщения о результате своего исполнения для отображения в веб консоли. Такие сообщения позволяют сообщить пользователю, вызвавшему этот пункт контекстного меню об успешном или неуспешном результате выполнения действия.

Пример подобного сообщения для пункта меню «Имя по IP» плагина `net-tools` приведен на рисунке 4.31:

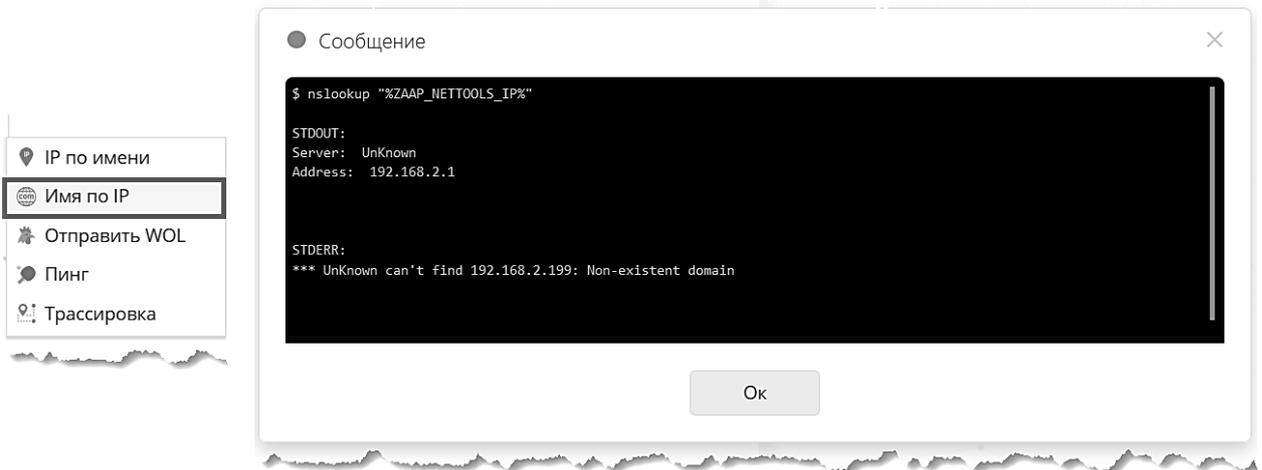


Рисунок 4.31 Всплывающее окно с сообщением о результате выполнения пункта меню «Имя по IP» плагина `net-tools`

Для возврата результата используйте ключевое слово `return` с объектом из двух полей: `type` и `message`.

```
provide({ category: 'hello', context: 'computer' }).items =
function (ctx) {
  return {
    label: 'Действие с сообщением',
    action(ctx) {
      return {
        type: 'error',
        message: 'Произошла ошибка'
      }
    }
  }
}
```

Поле `type` предназначено для задания типа сообщения и поддерживает следующие значения:

```
error
info
warn
```

Поле `message` предназначено для задания текста сообщения. В задаваемом тексте поддерживаются переносы строки при помощи спецсимволов `\n` и `\r\n`.

## 5. РЕЖИМЫ УПРАВЛЯЕМЫХ КОМПЬЮТЕРОВ

### 5.1 Пользовательский интерфейс агента

#### 5.1.1 Компаньон агента

Инсталляция агента системы «Зодиак.АйТиЭм» на управляемом компьютере включает в себя также **компаньон агента** – приложение, работающее в фоновом режиме в сеансе пользователя и обеспечивающее взаимодействие пользователя с агентом посредством системных уведомлений и веб-консоли.

После установки пользователю доступны ярлык компаньона агента на рабочем столе и иконка в области уведомлений, как показано на рисунке 5.1. Ярлык также доступен в списке приложений в меню «Пуск».

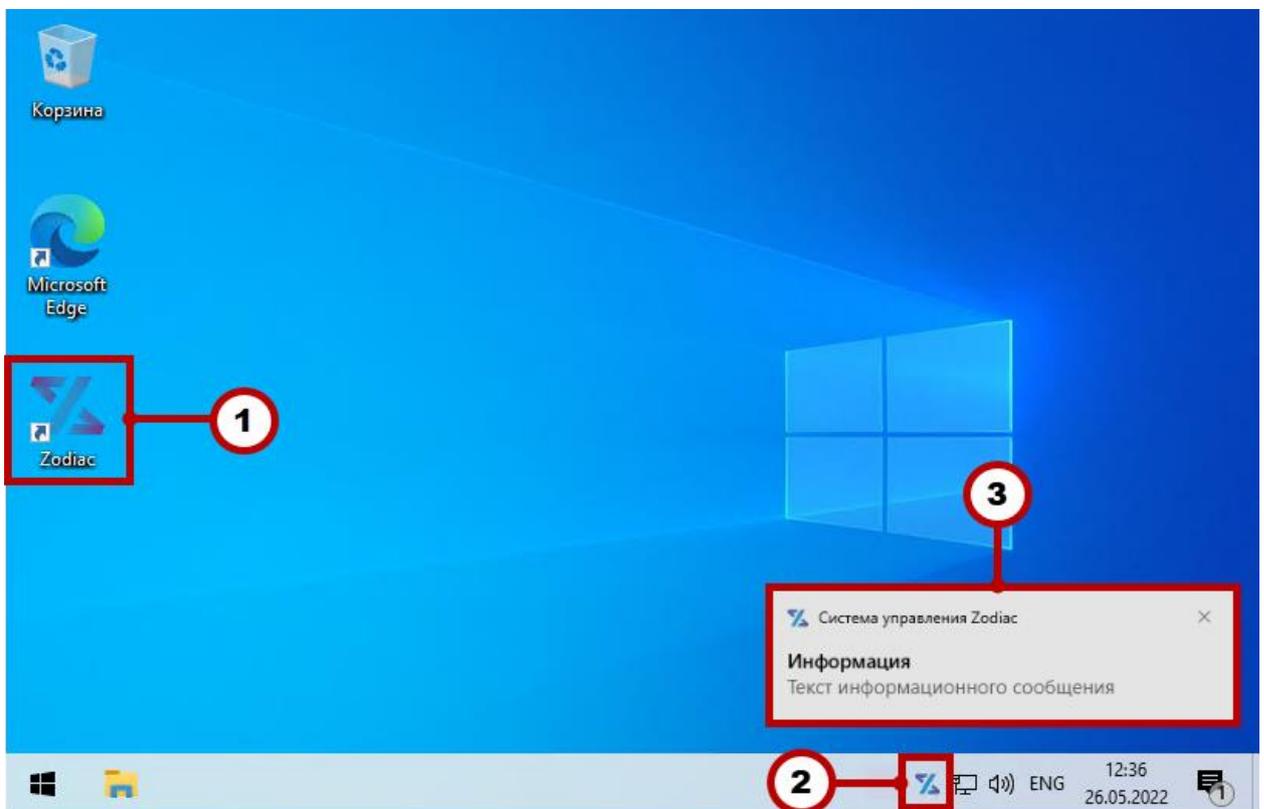


Рисунок 5.1 Компаньон агента – ярлык и иконка в области уведомлений

1. Ярлык компаньона агента на рабочем столе.
2. Иконка компаньона агента в области уведомлений.
3. Всплывающее оповещение – пример всплывающего оповещения от агента.

#### **i** Примечание

Уведомления пользователю могут направляться скриптами, исполняемыми агентом, а также утилитой **zonot** (раздел 5.2.1).

Веб-консоль компаньона агента вызывается двойным кликом по ярлыку на рабочем столе или из контекстного меню в области уведомлений.

### 5.1.2 Общий вид пользовательского интерфейса

Пользовательский интерфейс компаньона, представляет собой одностраничное веб-приложение (рисунок 5.2).

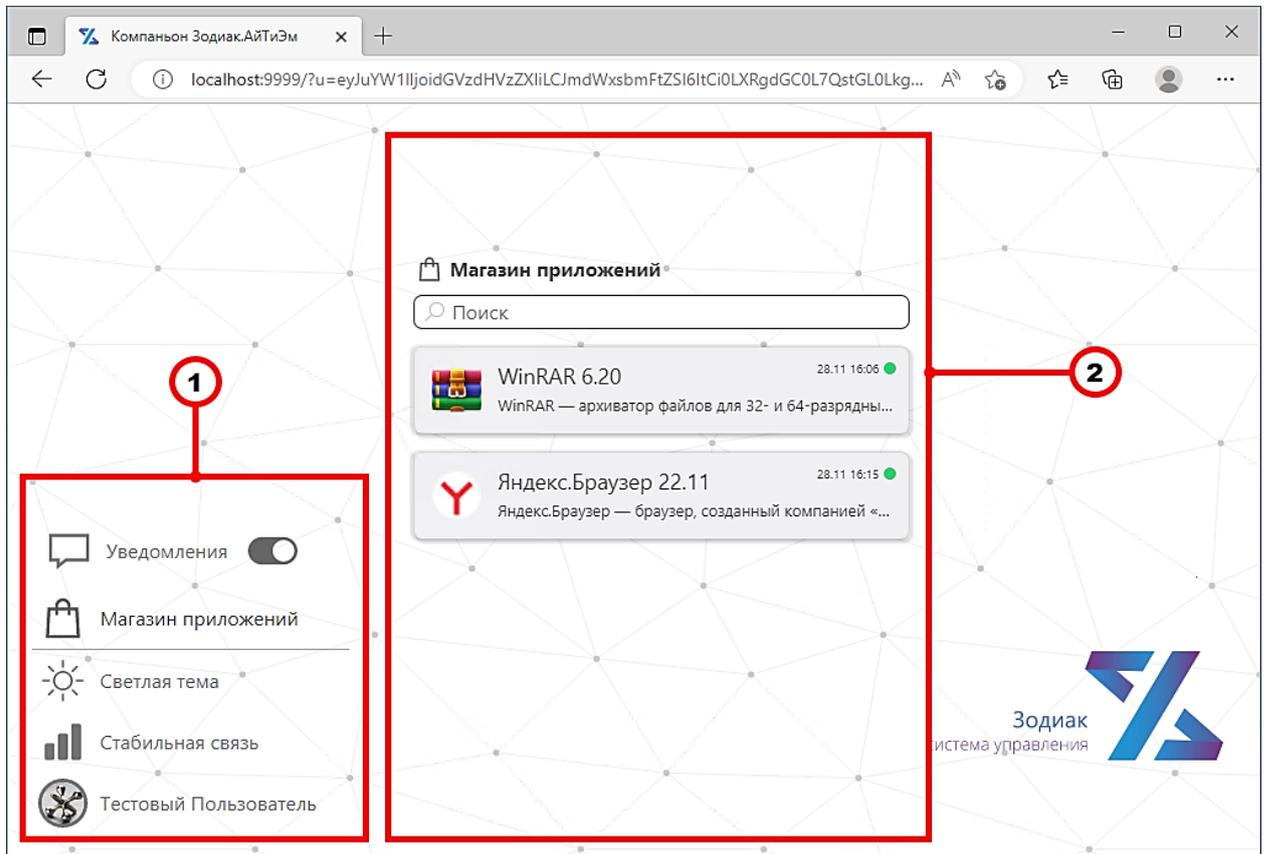


Рисунок 5.2 Пользовательский интерфейс компаньона

1. Панель переключения виджетов и управления настройками.
2. Область виджетов (в нашем примере показан виджет «Магазин приложений»).

### 5.1.3 Изменение цветовой палитры интерфейса

Пользовательский интерфейс компаньона поддерживает переключение темной и светлой темы, как показано на рисунке 5.3.

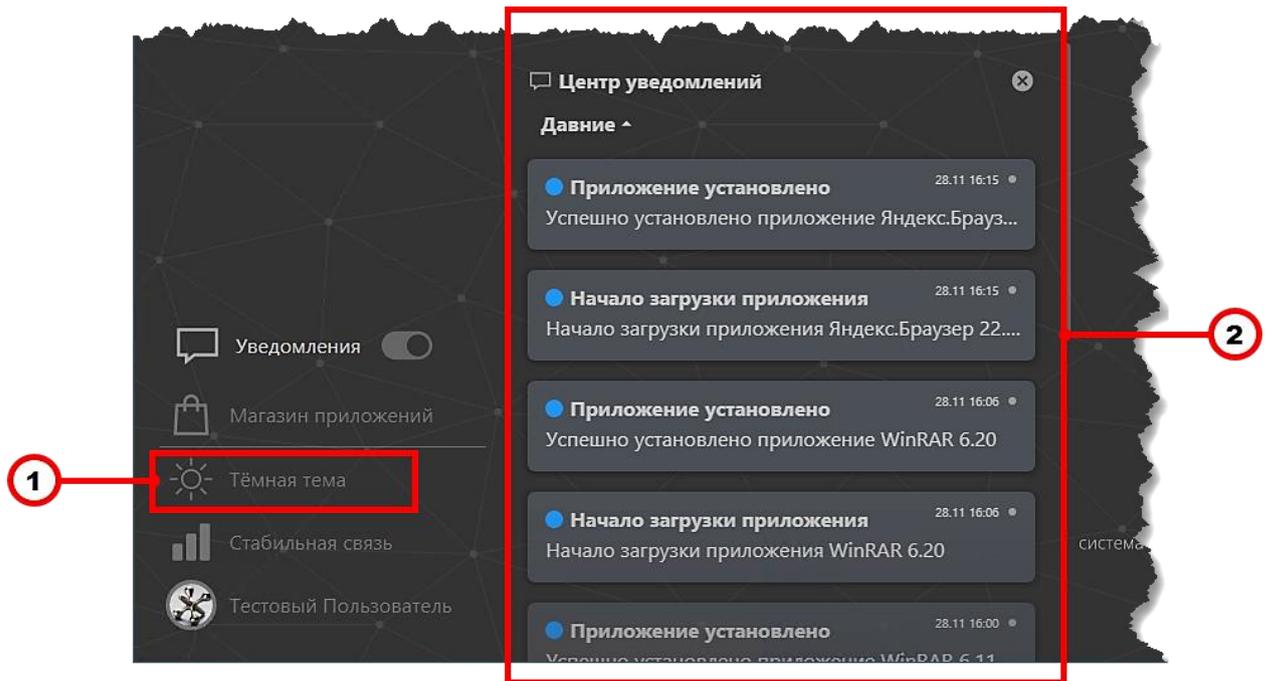


Рисунок 5.3 Переключение тем пользовательского интерфейса

1. Кнопка переключения светлой/темной темы.
2. Виджет «Центр уведомлений».

#### 5.1.4 Центр уведомлений

Виджет «Центр уведомлений» содержит список уведомлений (рисунок 5.4), полученных пользователем от агента «Зодиак.АйТиЭм». Уведомления сгруппированы по времени и снабжены цветовым индикатором категории важности. Пользователь может удалять отдельные сообщения или полностью очистить список.

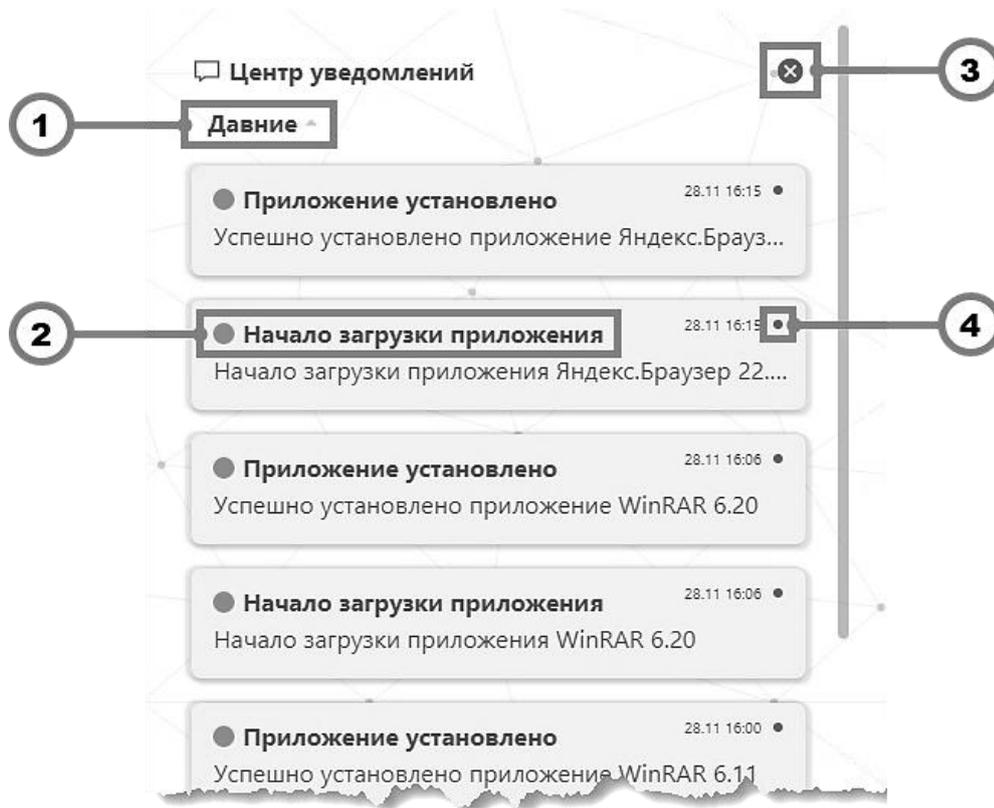


Рисунок 5.4 Центр уведомлений

1. Кнопка свертывания / развертывания сгруппированных по времени уведомлений.
2. Заголовок сообщения с цветовым индикатором.
3. Кнопка удаления всех сообщений.
4. Кнопка удаления сообщения.

#### 5.1.5 Приостановка уведомлений

При необходимости пользователь может приостанавливать отображение всплывающих оповещений в области уведомлений, как показано на рисунке 5.5. Поступившие за время приостановки уведомления будут доступны для просмотра в центре уведомлений.

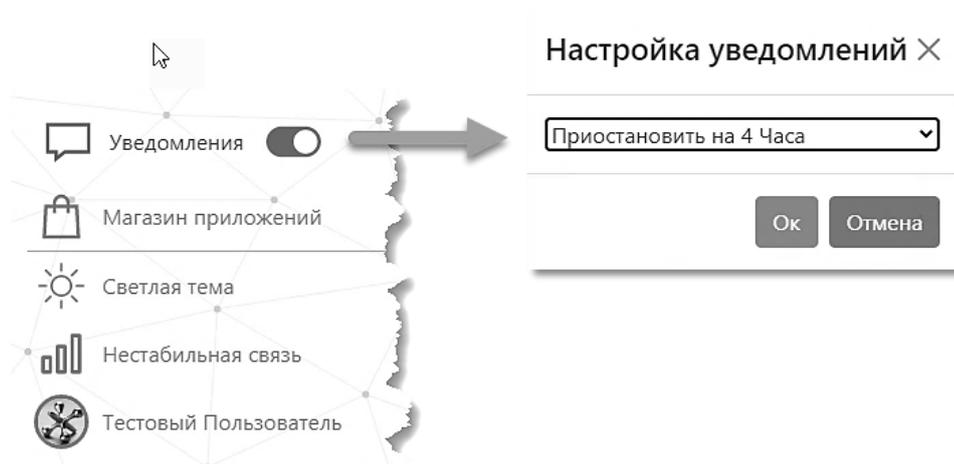


Рисунок 5.5 Приостановка уведомлений

### 5.1.6 Магазин приложений

Виджет «Магазин приложений» позволяет пользователю загружать и устанавливать приложения из списка предоставляемых сервером «Зодиак.АйТиЭм» пакетов развертывания. Имеется также возможность переустановки и автоматического обновления приложений.

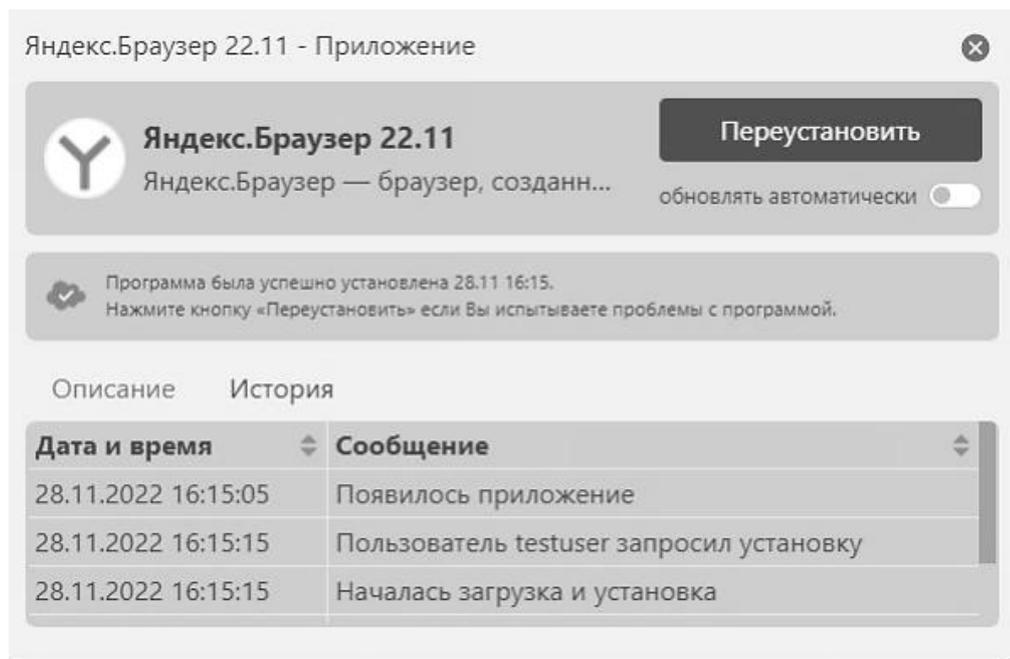


Рисунок 5.6 Панель установленного приложения

## 5.2 Утилиты командной строки

### 5.2.1 Утилита нотификации zonot

Инсталляция агента системы «Зодиак.АйТиЭм» на управляемом компьютере включает в себя также утилиту командной строки **zonot**, предназначенную для отправки уведомлений пользователю.

Для отправки уведомления пользователю в командной строке выполните следующую команду:

```
zonot "Заголовок сообщения" "Текст сообщения"
```

При отправке уведомления можно указать тип уведомления: **info**, **warning**, **error**. Для отправки предупреждения выполните следующую команду:

```
zonot --type warning "Заголовок предупреждения" "Текст предупреждения"
```

Для отправки предупреждения определенному пользователю компьютера используйте параметр **--user**:

```
zonot --type warning --user Иванов "Заголовок предупреждения" "Текст предупреждения"
```

### 5.2.2 Утилита инвентаризации zocol

Инсталляция агента системы «Зодиак.АйТиЭм» на управляемом компьютере включает в себя также утилиту командной строки **zocol**, представляющую собой коллектор данных инвентаризации управляемого компьютера.

Следующая команда выводит данные **базовой инвентаризации** в формате **JSON**.

```
zocol --basic
```

Пример вывода данных базовой инвентаризации представлен ниже (рисунок 5.7).

```
{
  "hostname": "ZDC-TEST-W10",
  "fqdn": "ZDC-TEST-W10",
  "zover": "1.0.0",
  "cpu": {
    "brand": "Ryzen 7 2700X Eight-Core P
rocessor",
    "cores": 8,
    "manufacturer": "AMD",
    "physicalCores": 4,
    "processors": 1,
    "speed": 3.7
  },
  "ram": {
    "total": 42938
B2704,
    "modules": []
  },
  "os": {
    "arch": "x64",
    "platform": "win32",
    "distro": "Майкрософт Windows 10 Pro",
    "release": "1
0.0.19043",
    "isServer": false,
    "sku": 48,
    "time": {
      "lastBootTime": "2022-05-12T06:24:43.584Z",
      "timezone": "GMT+030
0",
      "inventoryTime": "2022-05-26T23:02:43.600Z"
    },
    "disks": {
      "physicalDisks": [
        {
          "type": "HD",
          "name": "Msft Virtual
Disk",
          "vendor": "",
          "size": 136365211648,
          "interfacetype": "SAS",
          "serialNum": null
        },
        {
          "type": "Virtual Disk",
          "name": "C:
",
          "size": 135694102528,
          "fsType": "NTFS"
        }
      ],
      "logicalDisks": [
        {
          "name": "C:
",
          "size": 135694102528,
          "fsType": "NTFS"
        }
      ]
    },
    "network": {
      "networkInterfaces": [
        {
          "iFace": "Ethernet",
          "iFaceName": "Mi
crosoft Hyper-V Network Adapter",
          "ip4": "192.168.1.56",
          "mac": "00:15:5D:01:2D:31",
          "speed": 1000,
          "default": true
        }
      ]
    }
  }
}
Main function completed
Collector exited with 0 after 3 s 208.096 ms
```

Рисунок 5.7 Пример вывода данных, собранных утилитой zocol

Для форматирования выходных данных используйте параметр **--pretty**.

## 6. ПРИМЕРЫ ИСПОЛЬЗОВАНИЯ СИСТЕМЫ «ЗОДИАК.АЙТИЭМ»

---

### **Осторожно**

Функциональные возможности по решению бизнес-задач со стороны системы не ограничиваются приведенными сценариями. В данном разделе представлены наиболее показательные – для понимания подходов к организации решений практических задач.

---

### 6.1 Инвентаризация

#### 6.1.1 Базовая инвентаризация

Базовая инвентаризация осуществляется **автоматически** агентами, установленными на клиентских компьютерах, через заданные интервалы времени.

---

### **Примечание**

Периодичность базовой инвентаризации задается в конфигурации агентов. По умолчанию, агенты проводят инвентаризацию один раз в час.

---

- Данные базовой инвентаризации представляют собой фиксированный набор наиболее важных параметров компьютеров и всегда доступны в окне «Все компьютеры» (разд. 3.1.1).
- Для поиска компьютеров по основным параметрам инвентаризации (имя компьютера, ОС, IP-адрес и т.п.) используйте табличное представление данных с возможностями сортировки, группировки и фильтрации (рис. 3.1).
- По каждому отдельному компьютеру данные базовой инвентаризации можно просмотреть в карточке компьютера в режиме «Сводка» (рис. 3.3). Детальный обзор данных доступен в карточке компьютера в режиме «Подробности» (рис. 3.4).

#### 6.1.2 Настраиваемая инвентаризация

В качестве простого примера настраиваемой инвентаризации можно привести инвентаризацию всех служб, запущенных на компьютерах под управлением ОС Windows.

1. Создайте группу, например **«Компьютеры Windows»**, и включите в нее компьютеры под управлением ОС семейства Windows, подлежащие инвентаризации, как описано в разделе 3.3.2.
2. Создайте директиву, например, **«Инвентаризация компьютеров Windows»**, как описано в разделе 3.4.1, и включите в него элемент типа скрипт **PowerShell** (разд. 3.4.2) с именем **«Win32\_Service»** и следующим текстом:

```
Get-WmiObject -Class Win32_Service -filter "Started=True" `
| Select-Object -Property Name, Caption, StartMode, StartName
```

а также со следующими настройками:

| Настройка           | Значение                                   |
|---------------------|--------------------------------------------|
| По расписанию       | Включено<br>Каждую минуту                  |
| Собирать статистику | Включено                                   |
| Собирать результат  | Включено                                   |
| Свойства            | Name,<br>Caption<br>StartMode<br>StartName |

3. Создайте ассоциацию, например «**Назначения на компьютеры Windows**» как описано в разделе 3.2.2.
4. Назначьте директиву «**Инвентаризация компьютеров Windows**» на группу «**Компьютеры Windows**» с помощью ассоциации «**Назначения на компьютеры Windows**» как описано в разделе 3.2.3.
5. Правильность назначения можно проверить способом, описанным в разделе 3.1.2. При правильной последовательности действий назначение для какой-либо из выбранных компьютеров будет выглядеть, как на рисунке 6.1.

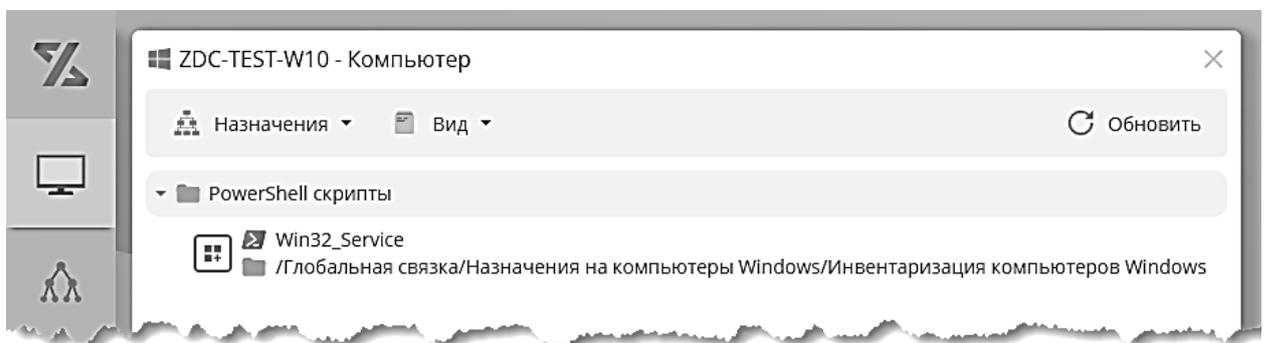


Рисунок 6.1 Проверка назначения директивы на компьютер

6. Проверьте корректность работы скрипта как описано в разделе 3.4.11. Для появления данных должно пройти как минимум одно **успешное выполнение** скрипта (рис. 3.49).
7. Предварительно, результаты работы скрипта можно просмотреть способом, описанным в разделе 3.4.12.
8. Создайте выборку «**Службы компьютеров Windows**» как описано в разделе 3.5, задав отображаемые имена для свойств «Name», «Caption», «StartMode» и «StartName» как «Имя службы», «Описание службы», «Тип Запуска» и «Вход от имени», соответственно. При правильной последовательности действий выборка будет выглядеть приблизительно как показано на рисунке 6.2.

| Компьютер    | Описание службы                                  | Имя службы               | Тип запуска | Вход от имени             |
|--------------|--------------------------------------------------|--------------------------|-------------|---------------------------|
| ZDC-AG-WIN11 | Служба шлюза уровня приложения                   | ALG                      | Manual      | NT AUTHORITY\LocalService |
| ZDC-AG-WIN11 | Удостоверение приложения                         | AppIDSvc                 | Manual      | NT Authority\LocalService |
| ZDC-AG-WIN11 | Сведения о приложении                            | Appinfo                  | Manual      | LocalSystem               |
| ZDC-AG-WIN11 | Управление приложениями                          | AppMgmt                  | Manual      | LocalSystem               |
| ZDC-AG-WIN11 | Готовность приложений                            | AppReadiness             | Manual      | LocalSystem               |
| ZDC-AG-WIN11 | Microsoft App-V Client                           | AppVClient               | Disabled    | LocalSystem               |
| ZDC-AG-WIN11 | Служба развертывания AppX (AppXSVC)              | AppXSvc                  | Manual      | LocalSystem               |
| ZDC-AG-WIN11 | Служба AssignedAccessManager                     | AssignedAccessManagerSvc | Manual      | LocalSystem               |
| ZDC-AG-WIN11 | Средство построения конечных точек Windows Audio | AudioEndpointBuilder     | Auto        | LocalSystem               |
| ZDC-AG-WIN11 | Windows Audio                                    | Audiosrv                 | Auto        | NT AUTHORITY\LocalService |
| ZDC-AG-WIN11 | Время в сети мобильной связи                     | autotimesvc              | Manual      | NT AUTHORITY\LocalService |

Рисунок 6.2 Примерный вид выборки для настраиваемой инвентаризации

### 6.1.3 Файловая инвентаризация под Windows

Файловая инвентаризация является вариантом настраиваемой инвентаризации, подробно описанной в разделе 6.1.2.

Пример скрипта **PowerShell** для получения списка файлов под ОС Windows:

```
# Скрипт проводит поиск файлов в заданной папке с учетом настроек.
# Возвращает массив объектов (объект - на каждый файл) вида:
# @{
#   'Name' = 'имя файла'
#   'Path' = 'путь до файла'
#   'OriginalName' = '(опционально из FileVersionInfo)'
#   'Description' = '(опционально из FileVersionInfo)'
#   'FileVersion' = '(опционально из FileVersionInfo)'
#   'ProductVersion' = '(опционально из FileVersionInfo)'
#   'ProductName' = '(опционально из FileVersionInfo)'
# }

# Настройки поиска

# Папка
$folder = 'C:\Program Files\';
# Признак поиска во вложенных папках
$recursive = $true;
# Фильтр
$filter = '*.exe';
# Максимальное число возвращаемых файлов
$amount = 50;
# Признак включения полей из FileVersionInfo
$includeFileVersionInfo = $true;

if ($recursive) {
    $items = Get-ChildItem -Path $folder -Filter $filter -Recurse | Select-Object -First $amount
}
else {
    $items = Get-ChildItem -Path $fullPath | Select-Object -First $amount;
}

$files = @();

foreach ($item in $items) {
```

```
$file = @{
    Name = $item.Name;
    Path = $item.FullName;
}

if ($includeFileVersionInfo) {

    $versionInfo = [System.Diagnostics.FileVersionInfo]::GetVersionInfo($item.FullName);

    $file.OriginalName = $versionInfo.OriginalFilename;
    $file.Description = $versionInfo.FileDescription;
    $file.FileVersion = $versionInfo.FileVersion;
    $file.ProductVersion = $versionInfo.ProductVersion;
    $file.ProductName = $versionInfo.ProductName;
}

$files += $file;
}

$files
```

Для возвращения результата задайте в настройках элемента директивы следующие свойства:

| Настройка          | Значение                                                                                     |
|--------------------|----------------------------------------------------------------------------------------------|
| Собирать результат | Включено                                                                                     |
| Свойства           | Name,<br>Path<br>OriginalName<br>Description<br>FileVersion<br>ProductVersion<br>ProductName |

Данный скрипт вернет exe-файлы из папки 'C:\Program Files\' (с поиском в подпапках) с ограничением количества – максимум 50 файлов. Для изменений параметров поиска можно использовать настройки в тексте скрипта. Примерный вид результатов работы приведен на рисунке 6.3.

| hostname     | le_time           | Name                         | ProductName                             | Description                                               |
|--------------|-------------------|------------------------------|-----------------------------------------|-----------------------------------------------------------|
| ZDC-AG-WIN11 | 14.12.2023, 17:46 | 7z.exe                       | 7-Zip                                   | 7-Zip Console                                             |
| ZDC-AG-WIN11 | 14.12.2023, 17:46 | 7zFM.exe                     | 7-Zip                                   | 7-Zip File Manager                                        |
| ZDC-AG-WIN11 | 14.12.2023, 17:46 | 7zG.exe                      | 7-Zip                                   | 7-Zip GUI                                                 |
| ZDC-AG-WIN11 | 14.12.2023, 17:46 | InputPersonalization.exe     | Операционная система Microsoft WindowsR | Сервер персонализации ввода                               |
| ZDC-AG-WIN11 | 14.12.2023, 17:46 | ShapeCollector.exe           | Операционная система Microsoft WindowsR | Интерфейс персонализации распознавания рукописного текста |
| ZDC-AG-WIN11 | 14.12.2023, 17:46 | TabTip.exe                   | Операционная система Microsoft WindowsR | Сенсорная клавиатура и панель рукописного ввода           |
| ZDC-AG-WIN11 | 14.12.2023, 17:46 | msinfo32.exe                 | Операционная система Microsoft WindowsR | Сведения о системе                                        |
| ZDC-AG-WIN11 | 14.12.2023, 17:46 | DB Browser for SQLCipher.exe | DB Browser for SQLite                   | DB Browser for SQLite                                     |
| ZDC-AG-WIN11 | 14.12.2023, 17:46 | DB Browser for SQLite.exe    | DB Browser for SQLite                   | DB Browser for SQLite                                     |
| ZDC-AG-WIN11 | 14.12.2023, 17:46 | ExtExport.exe                | Internet Explorer                       | Internet Explorer ImpExp                                  |
| ZDC-AG-WIN11 | 14.12.2023, 17:46 | ieddiagcmd.exe               | Internet Explorer                       | Diagnostics utility for Inte                              |
| ZDC-AG-WIN11 | 14.12.2023, 17:46 | ieinstal.exe                 | Internet Explorer                       | Установщик надстроек                                      |
| ZDC-AG-WIN11 | 14.12.2023, 17:46 | ielowutil.exe                | Internet Explorer                       | Internet Low-Mic Utility T                                |
| ZDC-AG-WIN11 | 14.12.2023, 17:46 | iexplore.exe                 | Internet Explorer                       | Internet Explorer                                         |
| ZDC-AG-WIN11 | 14.12.2023, 17:46 | expediteupdater.exe          | Microsoft WindowsR Operating System     | expediteupdater                                           |
| ZDC-AG-WIN11 | 14.12.2023, 17:46 | uhssvc.exe                   | Microsoft WindowsR Operating System     | Microsoft Update Health                                   |

Рисунок 6.3 Результат работы скрипта файловой инвентаризации под ОС Windows

### 6.1.4 Файловая инвентаризация под Linux

Файловая инвентаризация является вариантом настраиваемой инвентаризации, подробно описанной в разделе 6.1.2.

Пример скрипта **shell** для получения списка файлов под ОС Linux:

```
# Скрипт проводит поиск файлов в заданной папке с учетом настроек.
# Для каждого найденного файла:
# - свойства выводятся каждое отдельной строкой в формате env
# - список свойств отделяется пустой строкой

# Настройки поиска

# Папка
path="/usr/lib/libreoffice/"
# Разрешения
perm="-u=x,g=x,o=x"

find -L $path -type f -perm $perm 2>/dev/null |
while read -r line
do
  fname=$line
  xpath=${fname%/*}
  xbase=${fname##*/}
  echo "path: $xpath"
  echo "name: $xbase"
  echo ""
done
```

Для возвращения результата задайте в настройках элемента директивы следующие свойства:

| Настройка          | Значение      |
|--------------------|---------------|
| Собирать результат | Включено      |
| Свойства           | Name,<br>Path |

Данный скрипт вернет исполняемые файлы из папки '/usr/lib/libreoffice/' (с поиском в подпапках). Для изменений параметров поиска можно использовать настройки в тексте скрипта. Примерный вид результатов работы приведен на рисунке 6.4.

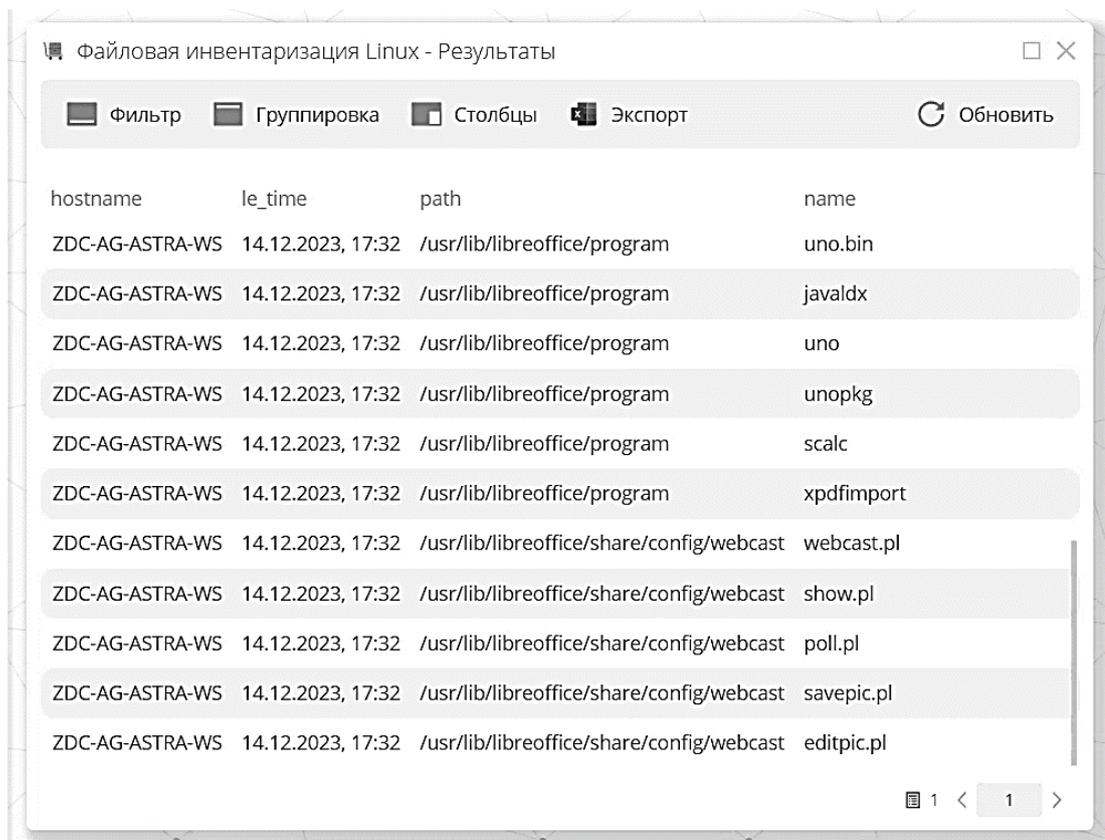


Рисунок 6.4 Результат работы скрипта файловой инвентаризации под ОС Linux

## 6.2 Управление конфигурацией

### 6.2.1 Выполнение скриптов PowerShell под Windows

В качестве простого примера организации изменений на управляемых компьютерах, приведем настройку выполнения скрипта для удаления временных файлов. Используем также концепцию авто-групп для автоматического распределения компьютеров по группам.

1. Настройте авто-группу «**Компьютеры Windows**», как описано в разделе 3.3.3, используя критерий **Платформа = win32**.

2. Создайте директиву, например, «**Обслуживание компьютеров Windows**», как описано в разделе 3.4.1, и включите в него элемент типа скрипт **PowerShell** (разд. 3.4.2) с именем, например, «**Remove temp files**» и следующим текстом:

```
rm c:\Windows\Temp\* -Recurse -Force -ErrorAction  
SilentlyContinue
```

Задайте также для скрипта исполнение по расписанию и сбор статистики для проверки работоспособности скрипта.

3. Создайте ассоциацию, например «**Назначения на компьютеры Windows**», как описано в разделе 3.2.2.
4. Назначьте директиву «**Обслуживание компьютеров Windows**» на группу «**Компьютеры Windows**» с помощью ассоциации «**Назначения на компьютеры Windows**», как описано в разделе 3.2.3.
5. Правильность назначения можно проверить способом, описанным в разделе 3.1.2.
6. Проверьте корректность работы скрипта, как описано в разделе 3.4.11. Для удаления временных файлов на компьютере должно пройти как минимум одно **успешное выполнение** скрипта на данном компьютере (рис. 3.49).

## 6.2.2 Выполнение скриптов Shell под Linux

1. Настройте авто-группу «**Компьютеры Linux**», как описано в разделе 3.3.3, используя критерий **Платформа = linux**.
2. Создайте директиву, например, «**Обслуживание компьютеров Linux**», как описано в разделе 3.4.1, и включите в него элемент типа скрипт **Shell** (разд. 3.4.2) с именем, например, «**Remove temp files**» и следующим текстом:

```
find /tmp -ctime +10 -exec rm -rf {} +
```

---

### **Примечание**

Приведенный для примера скрипт удаляет из каталога /tmp все файлы и папки старше, чем 10 дней.

В некоторых ситуациях предпочтительней может оказаться скрипт, который производит удаление всех файлов (но не папок), к которым не обращались более чем 10 дней:

```
find /tmp -type f -atime +10 -delete
```

Кроме того, как правило, /tmp очищается при перезагрузке, но для серверов, работающих **без перезагрузки**, регулярное очищений временных файлов может быть очень актуально.

---

Задайте также для скрипта исполнение по расписанию и сбор статистики для проверки работоспособности скрипта.

3. Создайте ассоциацию, например «**Назначения на компьютеры Linux**», как описано в разделе 3.2.2.

4. Назначьте директиву **«Обслуживание компьютеров Linux»** на группу **«Компьютеры Linux»** с помощью ассоциации **«Назначения на компьютеры Linux»**, как описано в разделе 3.2.3.
5. Правильность назначения можно проверить способом, описанным в разделе 3.1.2.
6. Проверьте корректность работы скрипта, как описано в разделе 3.4.11. Для удаления временных файлов на компьютере должно пройти как минимум одно **успешное выполнение** скрипта на данном компьютере (рис. 3.49).

### 6.3 Обнаружение соседних устройств

Рассмотрим пример обнаружения соседних устройств, т. е. устройств, находящихся в одной сети с определенными нами компьютерами.

1. Создайте директиву, например, **«Директива сканирования»**, как описано в разделе 3.4.1, и включите в него элемент директивы типа **«Сканирование сети»** как описано в разделе 3.4.7.
2. Создайте группу, например **«Сканеры»**, и включите в нее компьютеры, соседей которых вы хотите обнаружить, как описано в разделе 3.3.2
3. Создайте ассоциацию, например **«Назначения сканеров»**, как описано в разделе 3.2.2.
4. Назначьте директиву **«Директива сканирования»**, на группу **«Сканеры»** с помощью ассоциации **«Назначения сканеров»**, как описано в разделе 3.2.3.
5. Через некоторое время результаты сканирования будут доступны в режиме пользовательского интерфейса **«Соседние устройства»**. Работа с результатами сканирования подробно описана в разделе 3.15.

### 6.4 Установка ПО

Рассмотрим пример установки пакета CURL на компьютеры под управлением ОС Linux.

1. Создайте директиву, например, **«Развертывание ПО Linux»**, как описано в разделе 3.4.1, и включите в него элемент директивы типа **«Apt пакеты»** с именем, например, **«Установка curl»**, как описано в разделе 3.4.3.
2. Создайте группу, например **«Компьютеры Linux»**, и включите в нее компьютеры с ОС Linux, как описано в разделе 3.3.2.
3. Создайте ассоциацию, например **«Назначения развертывания ПО»**, как описано в разделе 3.2.2.
4. Назначьте директиву **«Развертывание ПО Linux»**, на группу **«Компьютеры Linux»** с помощью ассоциации **«Компьютеры Linux»**, как описано в разделе 3.2.3.
5. Результаты работы директивы можно проверить, как описано в разделе 3.4.11. Примерный результат работы по установке пакета curl приведен на рисунке 6.5. В статистике отражен **статус установки и текущие версии** пакета для каждого из компьютеров в задействованной группе.

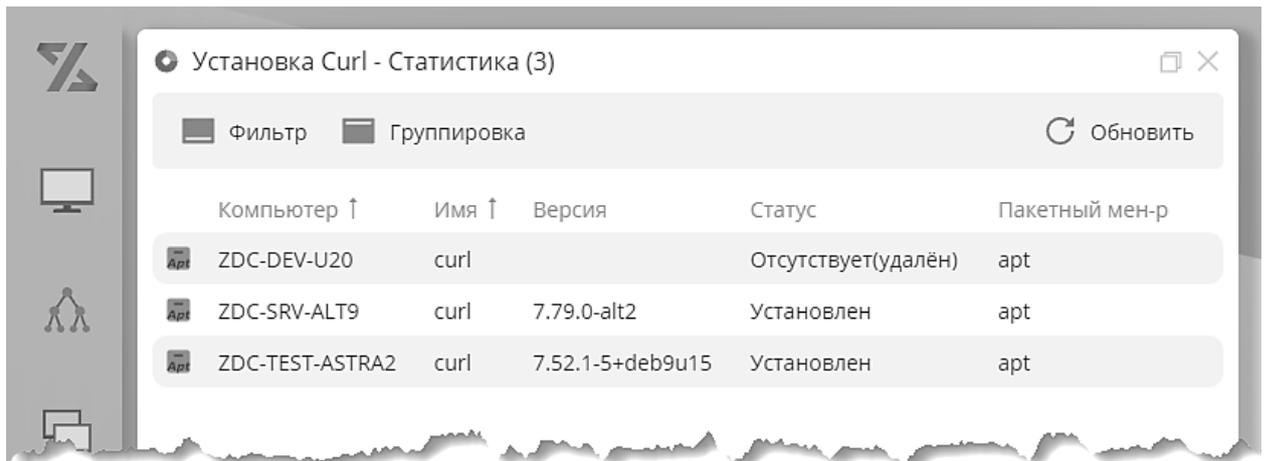


Рисунок 6.5 Примерный результат работы директивы «Apt пакеты»

## 6.5 Применение точек обслуживания

Точки обслуживания применяются для оптимизации передачи контента большого объема, что может иметь место, в частности, при передаче пакетов развертывания на управляемые компьютеры.

В разделе 3.4.6 было показано как элементы директивы «Развертывание» могут быть настроены для использования с ТО.

### ! Предупреждение

Если в настройках элемента директивы «Развертывание» переключатель «Использовать точки обслуживания» **выключен**, то скачивание контента производится **только с центрального сервера**.

Если переключатель «Использовать точки обслуживания» **включен**, то скачивание контента производится **только с ТО**. При этом, если **нет настроенных ТО** (раздел 3.4.13), или не настроено сопоставление УК и ТО с помощью **границ** (3.12), то скачивание контента **не производится**.

Рассмотрим простейший пример, позволяющий задействовать ТО в контуре, состоящем из одного сегмента локальной сети, как это часто бывает в случае стендового тестирования.

Для начала, перейдем в режим «Границы», выберем представление «Сценарии» и создадим **сценарий границ** PowerShell (раздел 3.12.1). Назовем его, для простоты, «hello», и зададим текст скрипта, который будет возвращать одноименную скрипту границу «hello» (рисунок 6.6):

```
@{ 'boundaries' = 'hello' }
```

## Примечание

**Сценарии границ** автоматически ассоциируются со всеми УК и ТО. Результаты исполнения сценариев регулярно **доставляются на сервер**.

Для того, чтобы **сервер сопоставил** некоторые УК и ТО, достаточно, чтобы эти УК и ТО в результате исполнения сценария передали на сервер **одинаковое имя границы**.



Рисунок 6.6 Сценарий границ PowerShell

Остальные настройки можно оставить со значениями по умолчанию. При сохранении выберите **«Включить режим "тест"»**, или включите тестовый режим из контекстного меню, как показано на рисунке 6.7.

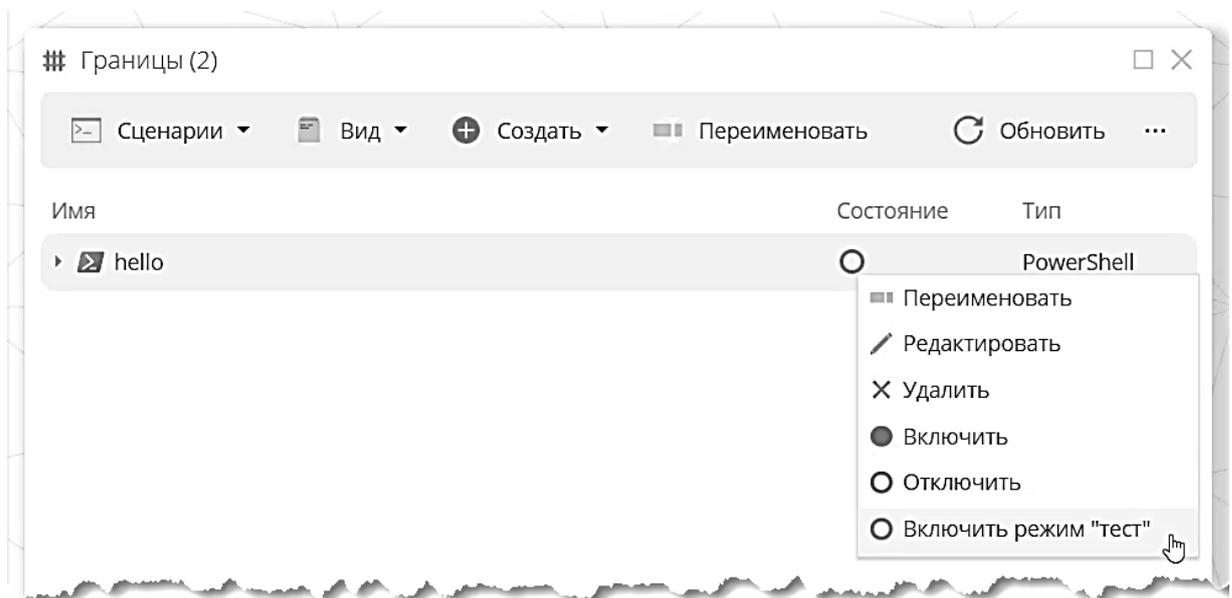


Рисунок 6.7 Переключение режима работы сценария границ

Через некоторое время после выполнения сценария на УК и ТО, в представлении «Границы» в списке границ появится граница **«hello»** (рисунок 6.8).

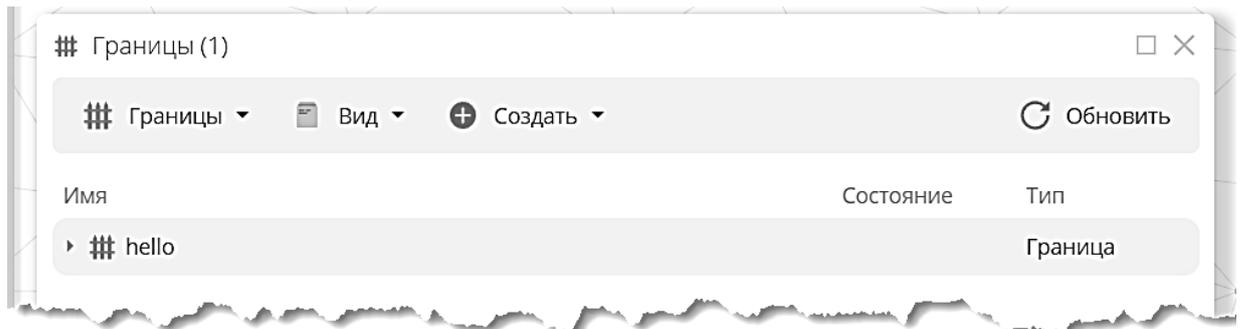


Рисунок 6.8 Список границ с одним элементом

По двойному клику на границу «hello» откроется панель со списком компьютеров, относящихся к данной границе. На рисунке 6.9 в списке мы видим компьютеры с ОС **Windows** – один УК и одна ТО (см. признак в колонке «ПТО»).

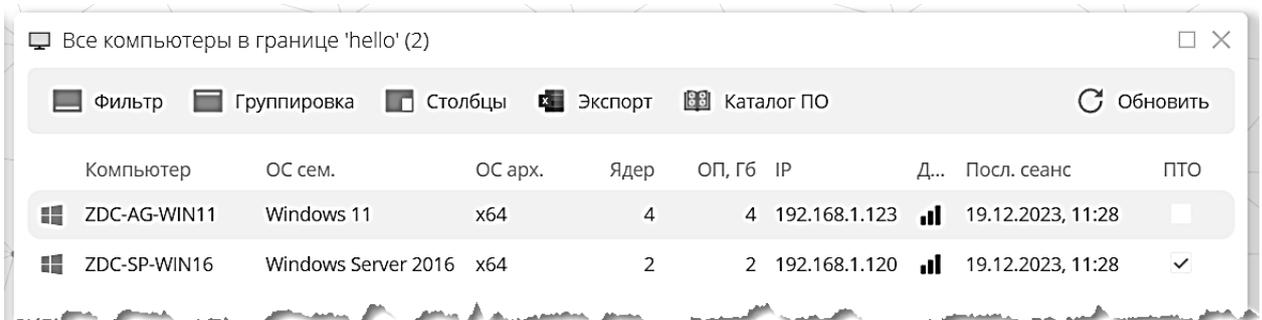


Рисунок 6.9 Список компьютеров, относящихся к границе

Для того, чтобы в границу «hello» вошли также УК под управлением ОС **Linux**, создадим еще один сценарий границ. На этот раз выберем скрипт **Shell**, имя можно использовать тоже самое – «hello». Зададим следующий текст скрипта:

```
echo "boundaries : hello"
```

Через некоторое время после выполнения сценария на УК увидим, что с границей «hello» сопоставлены оба наших скрипта «hello» (рисунок 6.10).

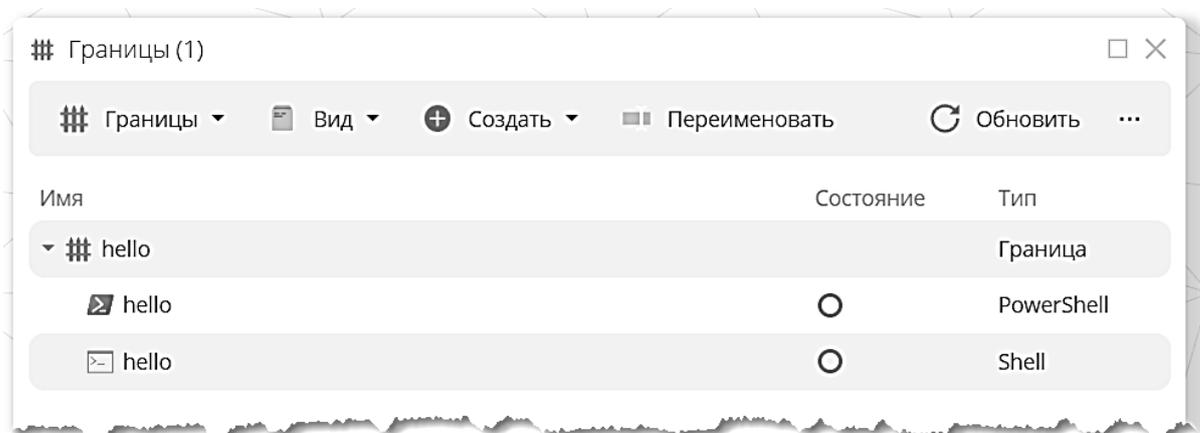
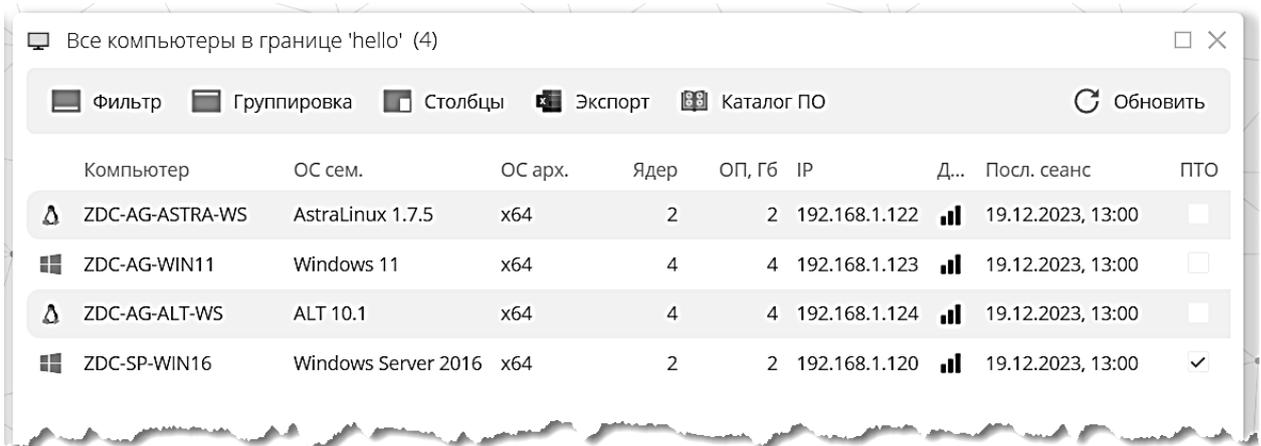


Рисунок 6.10 Граница с двумя определяющими сценариями

А также увидим, что в списке компьютеров, относящихся к границе «hello», появились УК под управлением **ОС Linux** (рисунок 6.11).



| Компьютер       | ОС сем.             | ОС арх. | Ядер | ОП, Гб | IP            | Д... | Посл. сеанс       | ПТО                                 |
|-----------------|---------------------|---------|------|--------|---------------|------|-------------------|-------------------------------------|
| ZDC-AG-ASTRA-WS | AstraLinux 1.7.5    | x64     | 2    | 2      | 192.168.1.122 | 📶    | 19.12.2023, 13:00 | <input type="checkbox"/>            |
| ZDC-AG-WIN11    | Windows 11          | x64     | 4    | 4      | 192.168.1.123 | 📶    | 19.12.2023, 13:00 | <input type="checkbox"/>            |
| ZDC-AG-ALT-WS   | ALT 10.1            | x64     | 4    | 4      | 192.168.1.124 | 📶    | 19.12.2023, 13:00 | <input type="checkbox"/>            |
| ZDC-SP-WIN16    | Windows Server 2016 | x64     | 2    | 2      | 192.168.1.120 | 📶    | 19.12.2023, 13:00 | <input checked="" type="checkbox"/> |

Рисунок 6.11 Список компьютеров с различными ОС в одной границе

При промышленной эксплуатации для определения границ используются различные алгоритмы, зависящие от **топологии сети**. Пример определения границ на основе маски подсети приведен в разделе 3.12.