Arenadata Streaming

Инструкция по установке Arenadata Streaming

2019 ИБС Экспертиза

Arenadata Streaming

Инструкция по установке Arenadata Streaming

Настоящий документ содержит описание действий для установки платформы Arenadata Streaming, а также информацию необходимую для ее эксплуатации.

Документ может быть полезен администраторам, программистам, разработчикам и сотрудникам подразделений информационных технологий, осуществляющих внедрение платформы.

Контактная информация службы поддержки:

E-mail: info@arenadata.io

Оглавление

1. Загрузка дистрибутива платформы 4
2. Установка ADS с помощью ADCM 5
2.1. Загрузка бандла ADS 6
2.2. Создание кластера ADS 7
2.2.1. Создание экземпляра кластера 7
2.2.2. Конфигурация кластера 9
2.2.3. Добавление сервисов 10
2.2.4. Добавление хостов 15
2.2.5. Размещение компонентов сервисов на хостах 16
2.2.6. Способы установки сервисов 19
2.3. Инструменты управления 22
2.3.1. Операции на уровне кластера 23
2.3.2. Операции на уровне сервиса 23
2.3.3. Запуск и остановка сервисов 25
3. Установка ADS с помощью Ambari 27
3.1. Настройка удаленного репозитория YUM 27
3.2. Установка сервера Ambari 27
3.3. Настройка сервера Ambari 27
3.4. Запуск сервера Ambari 28
4. Подготовка к установке Arenadata Streaming MPack 30
4.1. Загрузка Arenadata Streaming MPack 30
4.2. Импортирование MPack в Ambari стек 30

1. ЗАГРУЗКА ДИСТРИБУТИВА ПЛАТФОРМЫ

Для загрузки дистрибутива платформы Arenadata Streaming необходимо на странице https://arenadata.tech/products/ads/ выбрать вкладку “Скачать”. При этом происходит загрузка дистрибутива платформы в виде архива.

2. УСТАНОВКА ADS С ПОМОЩЬЮ ADCM

Самым простым вариантом установки кластера ADS является использование Arenadata Cluster Manager – ADCM.

Преимуществами установки кластера ADS через ADCM являются:

- Вся логика по установке и конфигурированию находится внутри бандла ADS:
  - Пользователю нет необходимости вручную вводить множество команд в консоли;
  - Сконфигурированы все важные настройки ОС и сервисов;
- Единый интерфейс доступа и управления конфигурациями:
  - Возможность управления несколькими сервисами через единый интерфейс;
  - Возможность управления несколькими кластерами через единый интерфейс;
- Существует возможность использования как облачной, так и физической инфраструктуры;
- Все rpm-пакеты для ADS берутся из репозитория Arenadata:
  - Все rpm протестированы;
  - Используется фиксированный набор пакетов, что упрощает помощь пользователям и составление баг-репортов.

Для установки кластера ADS посредством ADCM необходимо выполнить следующие предварительные шаги:

- Установить ADCM;
- Создать хосты для кластера ADS:
  - Загрузить выбранный бандл хоста. В текущем примере используется бандл SSH;
  - Инициализировать необходимое количество хостов (в текущем примере данный шаг пропускается, так как используются готовые хосты, а не облачный провайдер);
  - Добавить хосты в ADCM (в текущем примере используется 4 хоста): для Zookeeper (zk), для Kafka брокеров (kafka1 и kafka2), для Nifi (nifi1);
- (Опционально) Создать кластер мониторинга:
  - Загрузить бандл мониторинга;
  - Создать экземпляр кластера мониторинга и установить его.

2.1. Загрузка бандла ADS

Загрузка бандла ADS необходима для создания в ADCM прототипа кластера, из которого в дальнейшем возможно создание его экземпляров.

Для загрузки бандла следует выполнить следующие действия:

1. Открыть в ADCM вкладку “BUNDLES” (Рис.1.).

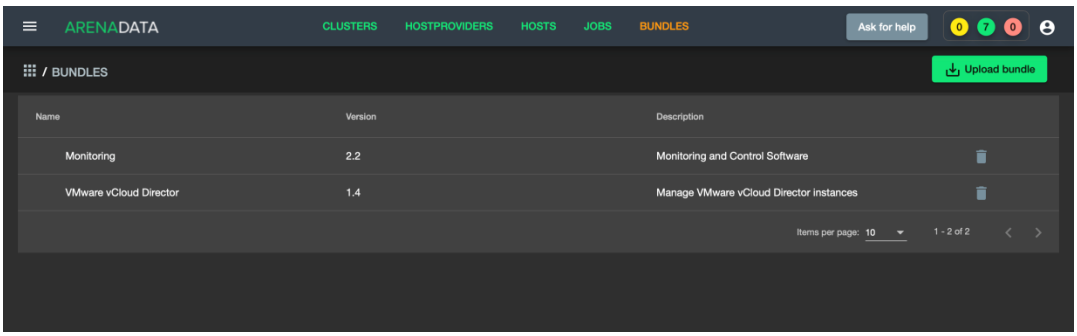


Рис. 1. Вкладка “BUNDLES”

2. Нажать “Upload bundle” и в открывшейся форме выбрать файл бандла ADS (Рис.2.).

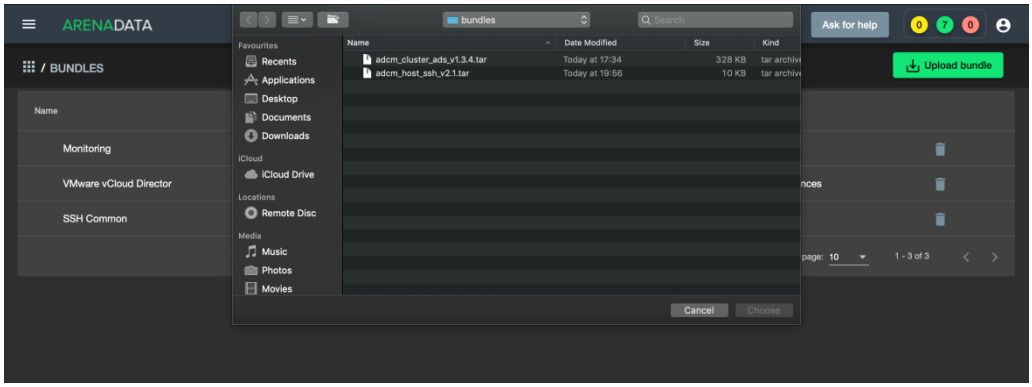


Рис. 2. Выбор бандла

3. В результате выполненных действий факт успешной загрузки отображается в общем списке бандлов на вкладке “BUNDLES” (Рис.3.).

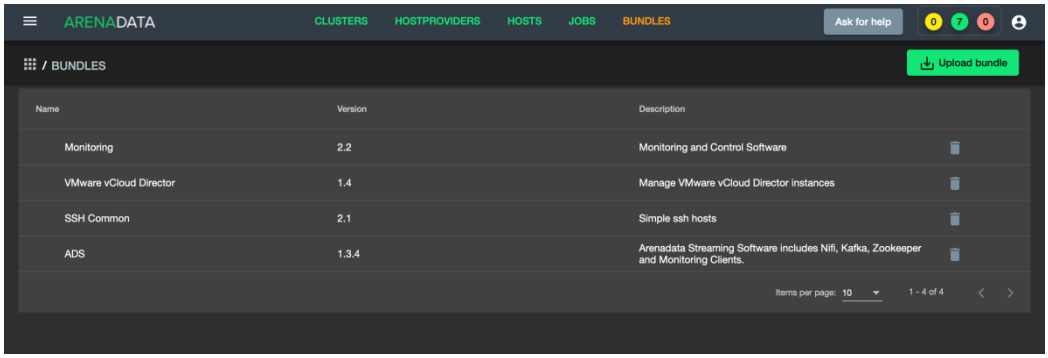


Рис. 3. Результат успешной загрузки бандла

2.2. Создание кластера ADS

После выполнения предварительных действий и загрузки бандла в кластер-менеджере ADCM содержится следующий список объектов (Рис.4):

- Прототип кластера мониторинга и созданный на его основе экземпляр (программа мониторинга развернута на хосте, ADCM содержит записи о его результатах и настройках);
- Прототип кластера ads для создания экземпляров;
- Предварительно подготовленные хосты, в базе данных ADCM хранятся записи о них и их учетных данных – ssh-ключах или паролях.

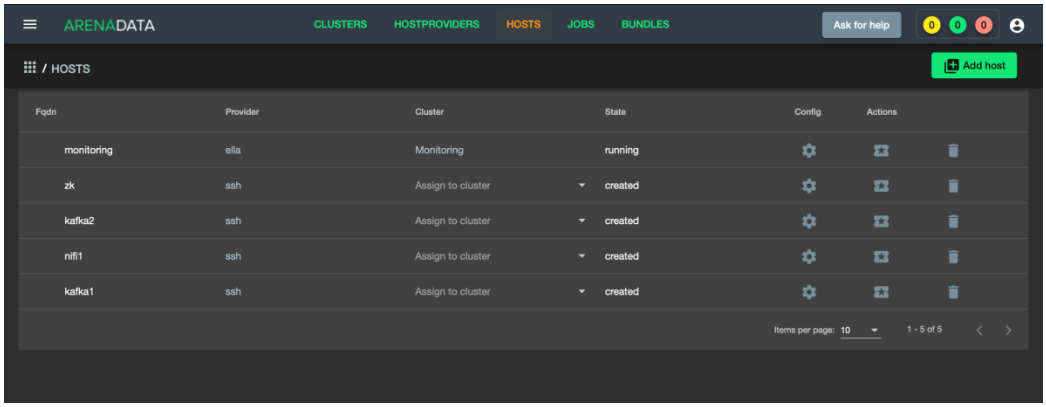


Рис. 4. Список объектов в ADCM

Данным объектам доступен следующий функционал:

- Создание экземпляра кластера;
- Конфигурация кластера;
- Добавление сервисов;
- Добавление хостов;
- Размещение компонентов сервисов на хостах;
- Установка сервиса Zookeeper;
- Установка сервиса Kafka;
- Установка сервиса Nifi;
- Установка сервиса Monitoring Clients.

2.2.1. Создание экземпляра кластера

При создании кластера в веб-интерфейсе ADCM генерируется новый экземпляр кластера ads, что означает только добавление данных о нем в базу данных ADCM – на этом этапе не производится установка ads на хосты.

1. Открыть в ADCM вкладку “CLUSTERS” (Рис.5.).

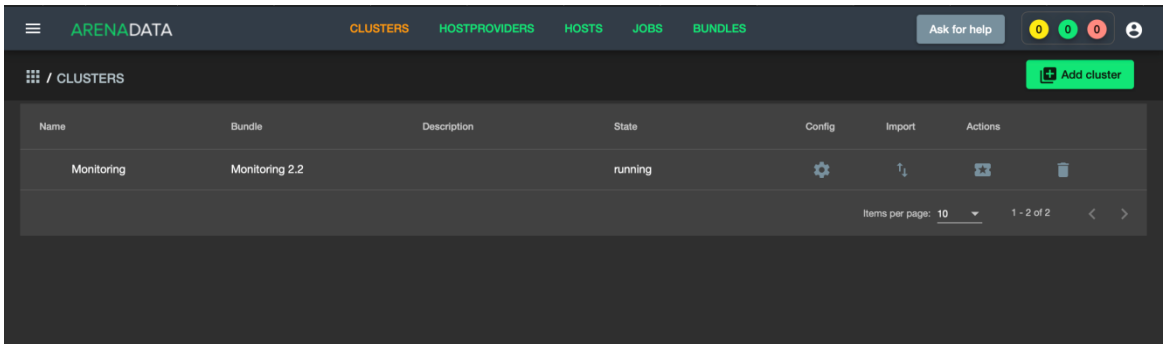


Рис. 5. Вкладка “CLUSTERS”

2. Нажать “Add cluster” и в открывшейся форме создать экземпляр кластера из прототипа ads, полученного из бандла (Рис.6.).

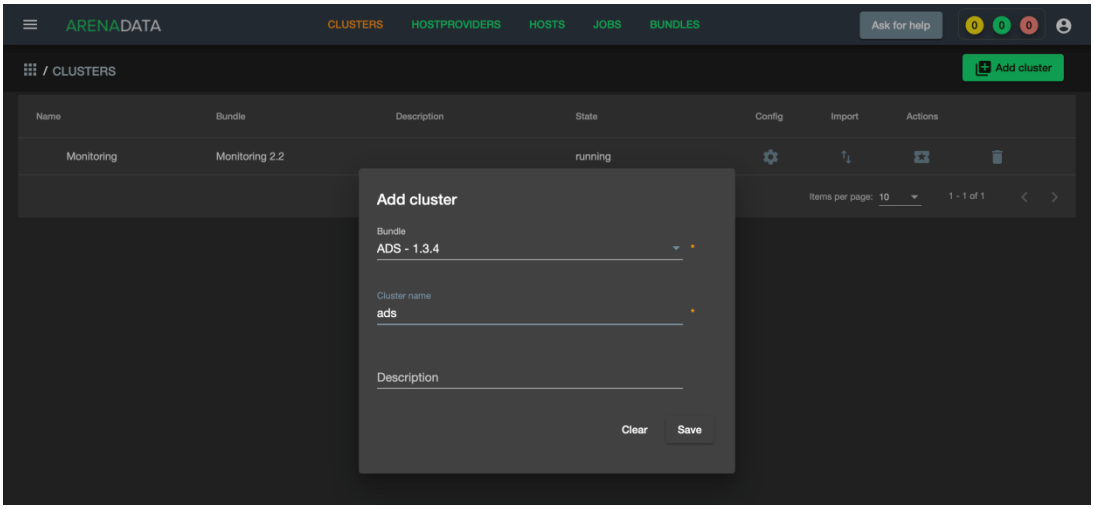


Рис. 6. Создание экземпляра кластера

3. В результате выполненных действий факт создания экземпляра кластера отображается в базе данных ADCM на вкладке “CLUSTERS” (Рис.7.).

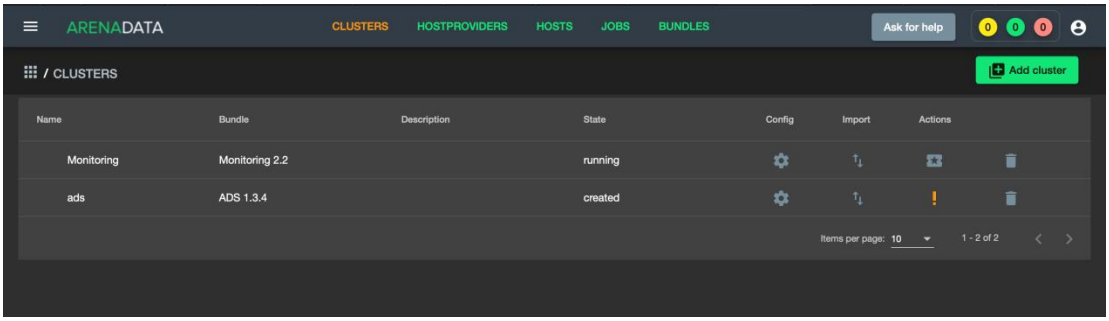


Рис. 7. Результат успешного создания экземпляра кластера

2.2.2. Конфигурация кластера

Для перехода к настройкам экземпляра кластера ads необходимо нажать кнопку с пиктограммой шестеренки в соответствующей строке вкладки “CLUSTERS” (Рис.11.) и перейти в раздел меню “Configuration”. При этом открывается окно конфигурации выбранного экземпляра (Рис.8.).

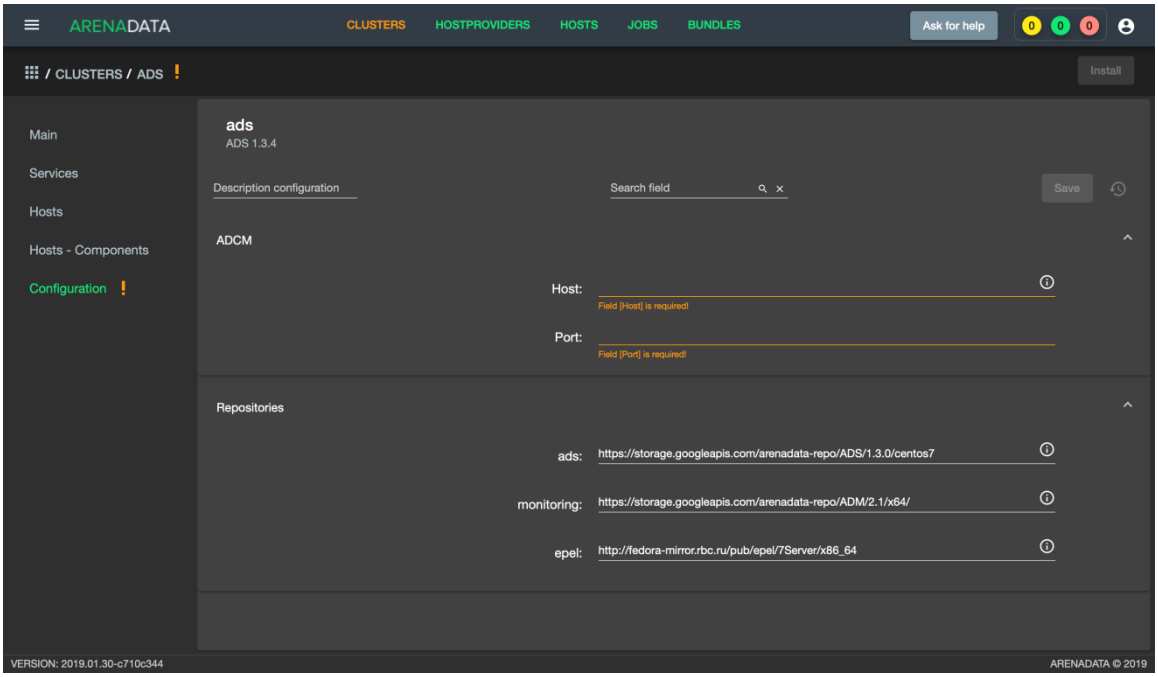


Рис. 8. Окно конфигурации кластера

В блоке “ADCM” есть возможность настройки адресов:

- Host (можно указать fqdn и IP-адрес);
- Port.

В блоке настроек “Repositories” указываются требуемые для установки ads пакеты из различных yum-репозиториев, при этом в каждом из параметров можно изменить заданный по умолчанию url на необходимый:

- ads;
- monitoring;
- epel.

Каждый компонент сервиса кластера имеет возможность отсылать статусную информацию о своем состоянии (keep alive) процессу ADCM в докер-контейнере. В ряде случаев ADCM может оказаться за NAT, и тогда исключается очевидный вариант автоматического определения его адреса, видимого со стороны сервисного компонента на хосте кластера. Поэтому данный адрес указывается вручную. Во время установки ADS адрес ADCM используется для заполнения конфигурации компонентов, отвечающих за передачу статусной информации.

2.2.3. Добавление сервисов

Кластер ADS содержит следующие сервисы:

- Zookeeper – сервис, предназначенный для хранения конфигураций, выполнения распределенной синхронизации процессов;
- Kafka – распределенная пплатформа для потоковых операций и данных;
- Nifi – распрделенная платформа, предназначенная для построения и автоматизации потоков данных между различными системами;
- Monitoring Clients – агенты, отсылающие информацию о хосте и ADS в мониторинг.

Не все сервисы являются обязательными для установки. Например, если вы не планируете использовать Nifi, то нет необходимости добавлять этот сервис. Или в случае, когда применяется сервис мониторинга (не на базе Graphite), незачем ставить агенты из Monitoring Clients. Однако, если планируется использование Kafka, одноименный сервис и Zookeeper обязательны, тоже самое можно сказать и про сервис Nifi. В тоже время сервис может состоять из обязательных и необязательных компонентов. Например, сервис Kafka состоит из обязательного компонента broker и необязательных: manager и schema-registry.

В настоящем примере в кластер добавлены все сервисы:

- Настройка сервиса Zookeeper;
- Настройка сервиса Kafka;
- Настройка сервиса Nifi;
- Настройка сервиса Monitoring Clients.



НА ТЕКУЩИЙ МОМЕНТ НЕВОЗМОЖНО УДАЛИТЬ ИЗ КЛАСТЕРА УЖЕ ДОБАВЛЕННЫЙ СЕРВИС

2.2.3.1. Настройка сервиса Zookeeper

Для перехода к настройкам сервиса Zookeeper необходимо нажать кнопку с пиктограммой шестеренки в соответствующей строке вкладки “SERVICES” в интерфейсе ADCM и перейти в раздел меню “Configuration”. При этом открывается окно конфигурации сервиса Zookeeper (Рис.9.).

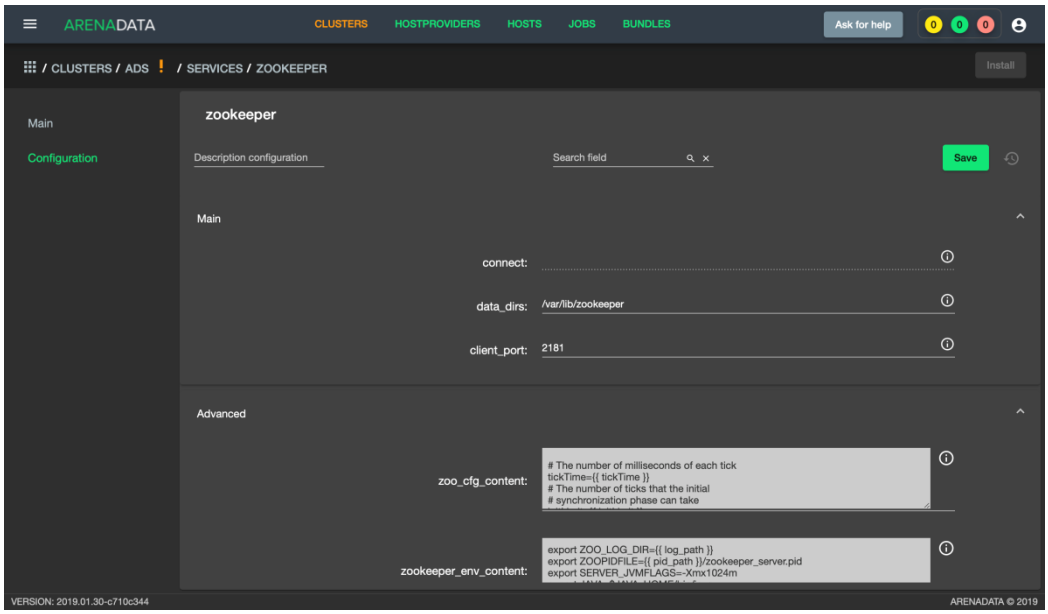


Рис. 9. Окно конфигурации сервиса Zookeeper

В блоке настроек “Main” задаются основные параметры:

- connect – строка подключения к Znode, в которой Zookeeper хранит конфигурацию текущего кластера, используется сервисом Kafka. В текущей реализации данный параметр недоступен для редактирования и генерируется на стороне ADCM автоматически;
- data_dirs – каталог для хранения транзакционных логов Zookeeper. Данный параметр указывается как dataDir в конфигурационном файле zoo.cfg;
- client_port – порт, на котором Zookeeper слушает клиентские подключения.

В блоке настроек “Advanced” задаются следующие расширенные параметры:

- zoo_cfg_content – содержимое файла zoo.cfg, которое в дальнейшем шаблонизируется. Данный параметр может использоваться для внесения дополнительных настроек;
- zookeeper_env_content – содержимое файла zookeeper-env.sh, которое в дальнейшем шаблонизируется. Данный параметр может использоваться для внесения переменных окружения.

2.2.3.2. Настройка сервиса Kafka

Для перехода к настройкам сервиса Kafka необходимо нажать кнопку с пиктограммой шестеренки в соответствующей строке вкладки “SERVICES” и перейти в раздел меню “Configuration”. При этом открывается окно конфигурации сервиса Kafka (Рис.10.).

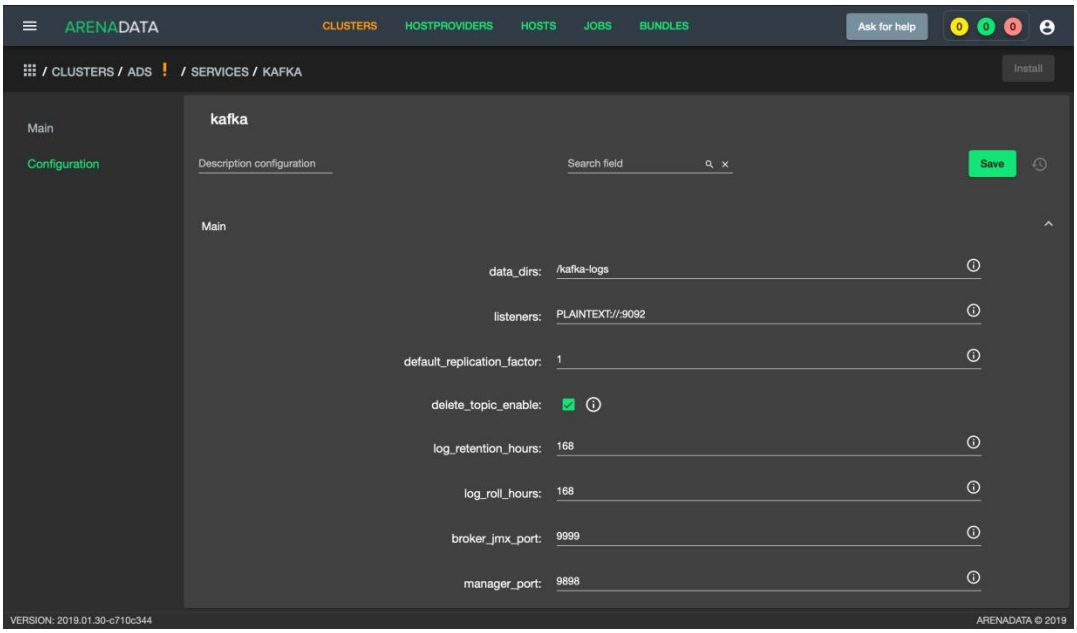


Рис. 10. Окно конфигурации сервиса Kafka

В блоке настроек “Main” задаются основные параметры:

- data_dirs – каталог для хранения данных в Kafka. Указывается к качестве параметра log.dirs в конфигурационном файле server.properties;
- listeners – список URI (протокол, хост и порт, на котором поднят брокер), разделенный запятыми. Если используется не PLAINTEXT протокол, то необходимо также указать listener.security.protocol.map. Для привязки ко всем интерфейсам указать имя хоста как 0.0.0.0. Оставить имя хоста пустым для привязки к интерфейсу по умолчанию. Указывается в качестве параметра listeners в конфигурационном файле server.properties;
- default_replication_factor – фактор репликации, с которым по умолчанию создаются и хранятся топики. Указывается в качестве параметра default.replication.factor в конфигурационном файле server.properties;
- delete_topic_enable – данный параметр позволяет удалять топики. Если параметр выключен, то удаление топика через инструменты администрирования не приводит к фактическому удалению. Указывается в качестве параметра default.replication.factor в конфигурационном файле server.properties;
- log_retention_hours – количество часов, в течение которых топики хранятся в Kafka. Указывается в качестве параметра log.retention.hours в конфигурационном файле server.properties;
- log_roll_hours – максимальное время, после которого появляется новый журнал сегмента, даже если старый журнал не переполнен. Указывается в качестве параметра log.roll.hours в конфигурационном файле server.properties;
- broker_jmx_port – порт, по которому Kafka брокер отдает jmx-метрики. Указывается в качестве параметра JMX_PORT в файле kafka-env.sh;

- manager_port – порт, на котором поднимается Kafka-Manager. Указывается в файле kafka-manager-env.sh;
- schema_registry_heap_opts – размер кучи, выделяемой процессу schema-registry. Указывается в качестве параметра SCHEMA_REGISTRY_HEAP_OPTS в schema-registry-env.sh;
- schema_registry_listener_port – порт, который слушает schema-registry. Указывается в качестве параметра listeners в конфигурационном файле schema-registry.properties;

В блоке настроек “Advanced” задаются следующие расширенные параметры:

- server_properties_content – содержимое файла server.properties, которое в дальнейшем шаблонизируется. Данный параметр может использоваться для внесения дополнительных настроек;
- kafka_env_content – содержимое файла kafka-env.sh, которое в дальнейшем шаблонизируется. Данный параметр может использоваться для внесения переменных окружения.

2.2.3.3. Настройка сервиса Nifi

Для перехода к настройкам сервиса Nifi необходимо нажать кнопку с пиктограммой шестеренки в соответствующей строке вкладки “SERVICES” и перейти в раздел меню “Configuration”. При этом открывается окно конфигурации сервиса Nifi (Рис.11.).

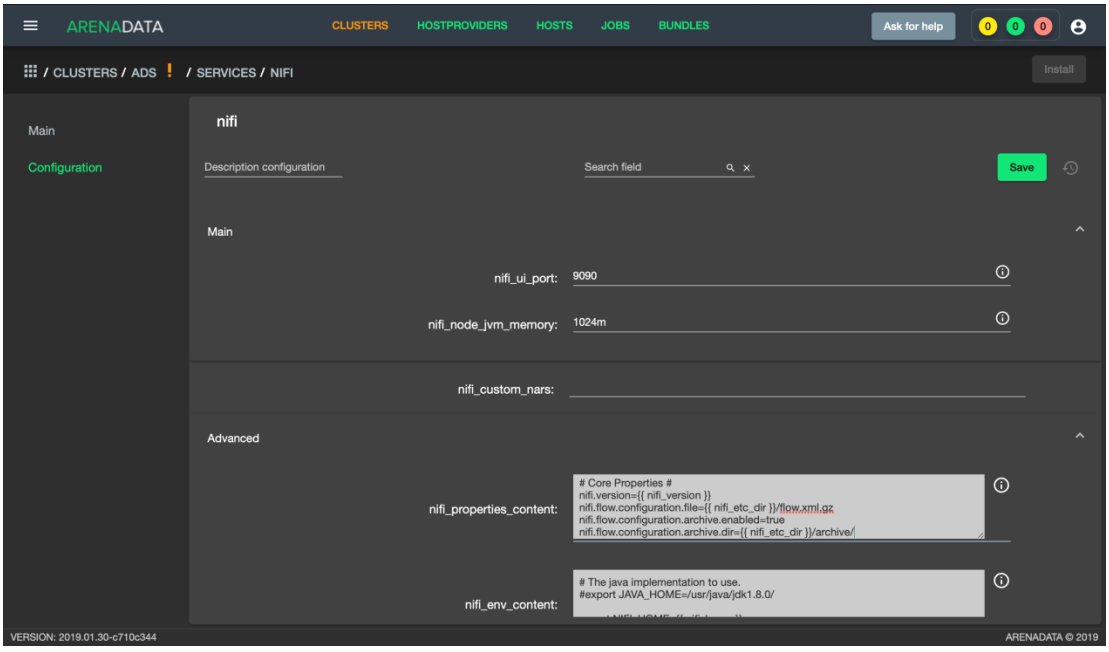


Рис. 11. Окно конфигурации сервиса Nifi

В блоке настроек “Main” задаются основные параметры:

- nifi_ui_port – http-порт, на котором поднимается веб-интерфейс сервиса Nifi. Указывается в качестве параметра nifi.web.http.port в конфигурационном файле nifi.properties;
- nifi_node_jvm_memory – размер кучи, выделяемой процессу сервиса Nifi. Указывается в конфигурационном файле bootstrap.conf.

В блоке настроек “Custom” задаются следующие необязательные параметры:

- nifi_custom_nars – параметр следует использовать в случае добавления custom nars; перечисляются через запятую. Указываются в качестве nifi.nar.library.directory.lib… в конфигурационном файле nifi.properties;

В блоке настроек “Advanced” задаются следующие расширенные параметры:

- nifi_properties_content – содержимое файла nifi.properties, которое в дальнейшем шаблонизируется. Данный параметр может использоваться для внесения дополнительных настроек;
- nifi_env_content – содержимое файла nifi-env.sh, которое в дальнейшем шаблонизируется. Данный параметр может использоваться для внесения переменных окружения;
- bootstrap_content – содержимое файла bootstrap.conf, которое в дальнейшем шаблонизируется. Данное поле может использоваться для внесения настроек, связанных с запуском сервиса;
- logback_content – содержимое файла logback.xml, которое в дальнейшем шаблонизируется. Данное поле может использоваться для внесения настроек, связанных с логированием;
- state_management_content – содержимое файла state_management.xml, которое в дальнейшем шаблонизируется. Данное поле может использоваться для внесения настроек, связанных с хранением состояния сервиса Nifi;
- authorizers_content – содержимое файла authorizers.xml, которое в дальнейшем шаблонизируется. Данное поле может использоваться для внесения настроек авторизации в сервис Nifi в том случае, если настроены политики безопасности;
- login_identity_providers_content – содержимое файла login_identity_providers.xml, которое в дальнейшем шаблонизируется. Данное поле может использоваться для внесения настроек авторизации, используемые state провайдером, в том случае, если настроены политики безопасности.

2.2.3.4. Настройка сервиса monitoring clients

Для перехода к настройкам сервиса monitoring clients необходимо нажать кнопку с пиктограммой шестеренки в соответствующей строке вкладки “SERVICES” и перейти в раздел меню “Configuration”. При этом открывается окно конфигурации сервиса monitoring clients (Рис.12.).

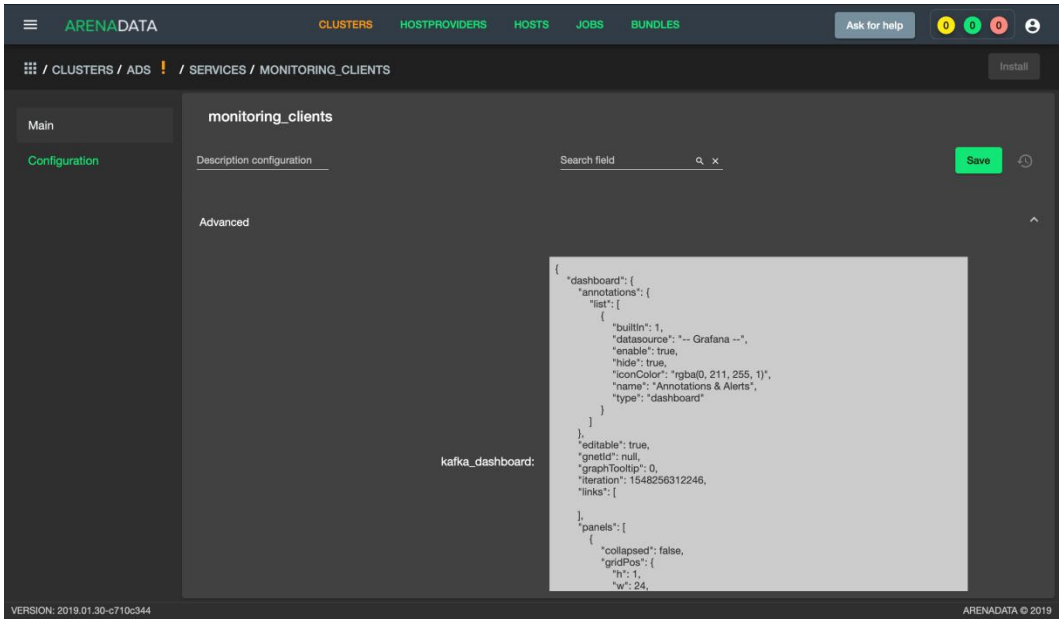


Рис. 12. Окно конфигурации сервиса Monitoring Clients

В блоке настроек “Advanced” задаются следующие расширенные параметры:

- kafka_dashboard – файл в формате json, который в дальнейшем шаблонизируется и отправляется в Grafana;
- kafka_metrics – файл в формате yaml, который в дальнейшем шаблонизируется. Включает в себя jmx метрики брокеров Kafka.

2.2.4. Добавление хостов

Для добавления хостов в кластер ads необходимо:

1. В меню кластера ads открыть вкладку “Hosts” (Рис.13.).

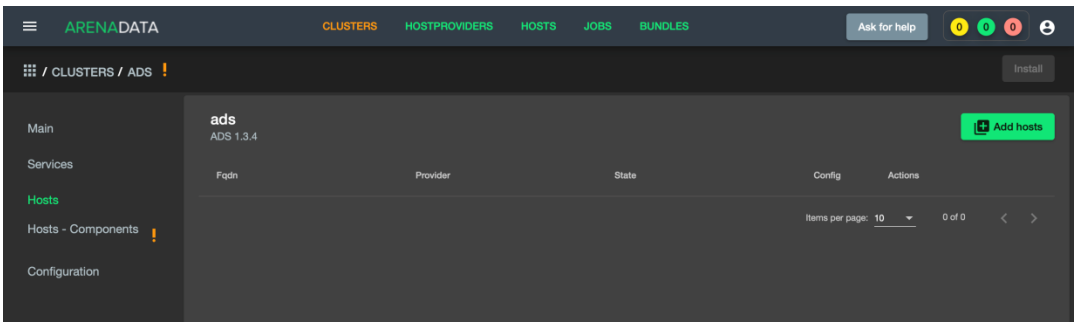


Рис. 13. Вкладка “Hosts” кластера ads

2. Нажать “Add hosts” и в открывшейся форме выбрать необходимые хосты (Рис.14).

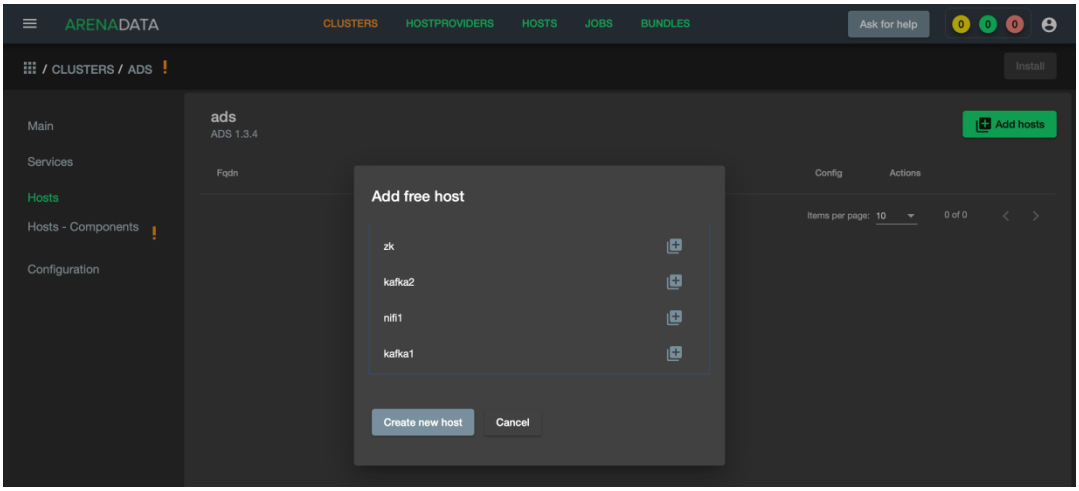


Рис. 14. Выбор хостов

3. В результате выполненных действий факт добавления хостов отображается в кластере ads в списке вкладки “Hosts” (Рис.15.).

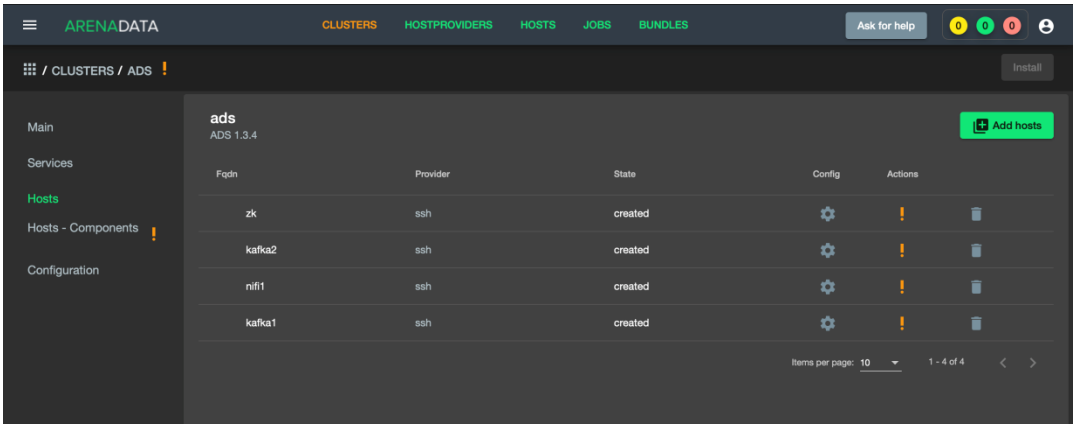


Рис. 15. Результат успешного добавления хостов

2.2.5. Размещение компонентов сервисов на хостах

Каждый сервис состоит из обязательных компонентов, которые должны быть размещены, и необязательных, которые могут быть не разщены на хостах кластера. Для этого необходимо на вкладке кластера “Hosts - Components” выбрать компонент посредством нажатия на него мышкой в колонке “Components” и определить для него необходимый хост в колонке “Hosts” (Рис.16.).

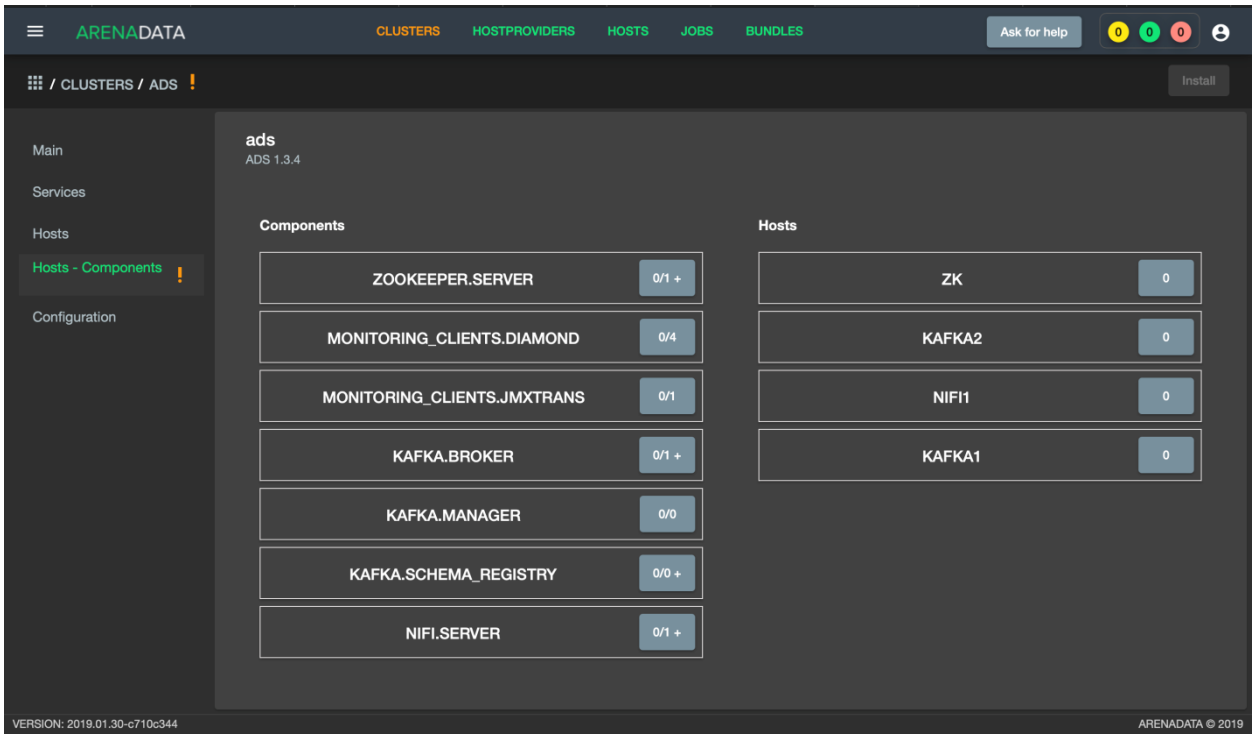


Рис. 16. Размещение компонентов сервисов на хостах

Поскольку сервисы Zookeeper, Kafka, Nifi и Monitoring Clients добавлены в кластер ADS, но еще не размещены на хостах, то изначально ни на одном из хостов нет компонентов:

1. Компоненты сервиса Zookeeper (Рис. 17.):

- Zookeeper.SERVER – необходимо добавить на один или нечетное количество хостов (zk1); для больших кластеров рекмоендуется не более 5.

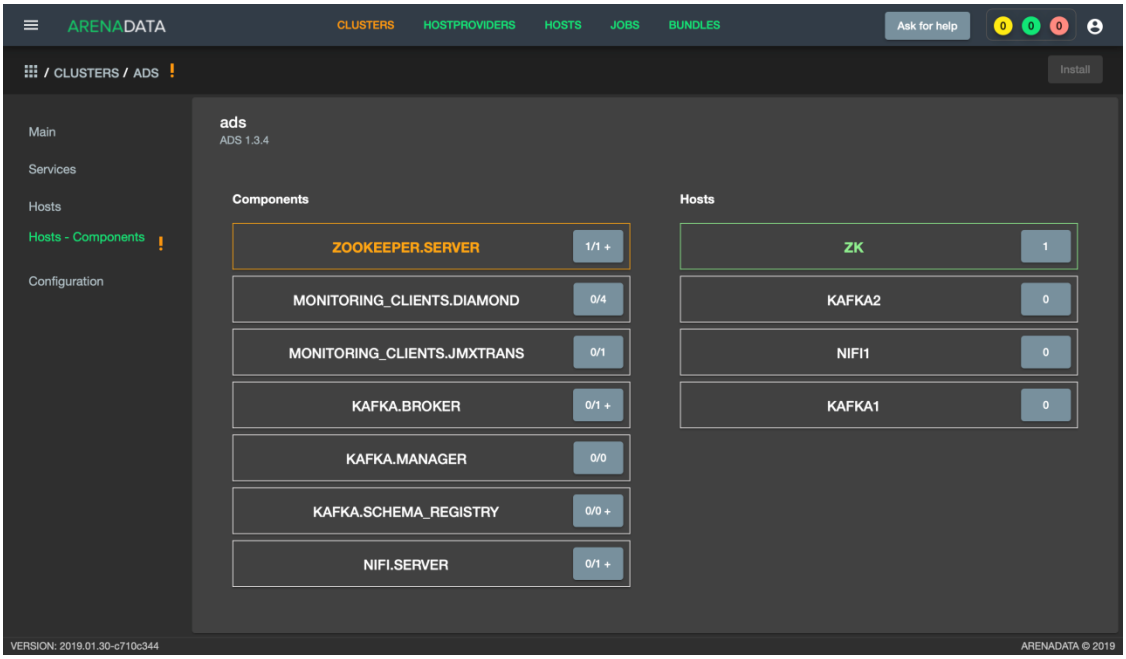


Рис. 17. Компоненты сервиса Zookeeper

2. Компоненты сервиса Kafka (Рис.18.):

- kafka.BROKER – необходимо добавить на один и более хостов брокеров (kafka1, kafka2);
- kafka.MANAGER – опционально может быть добавлен на один любой хост (kafka1);
- kafka.SCHEMA_REGISTRY – опционально может быть добавлен на один любой хост (kafka2).

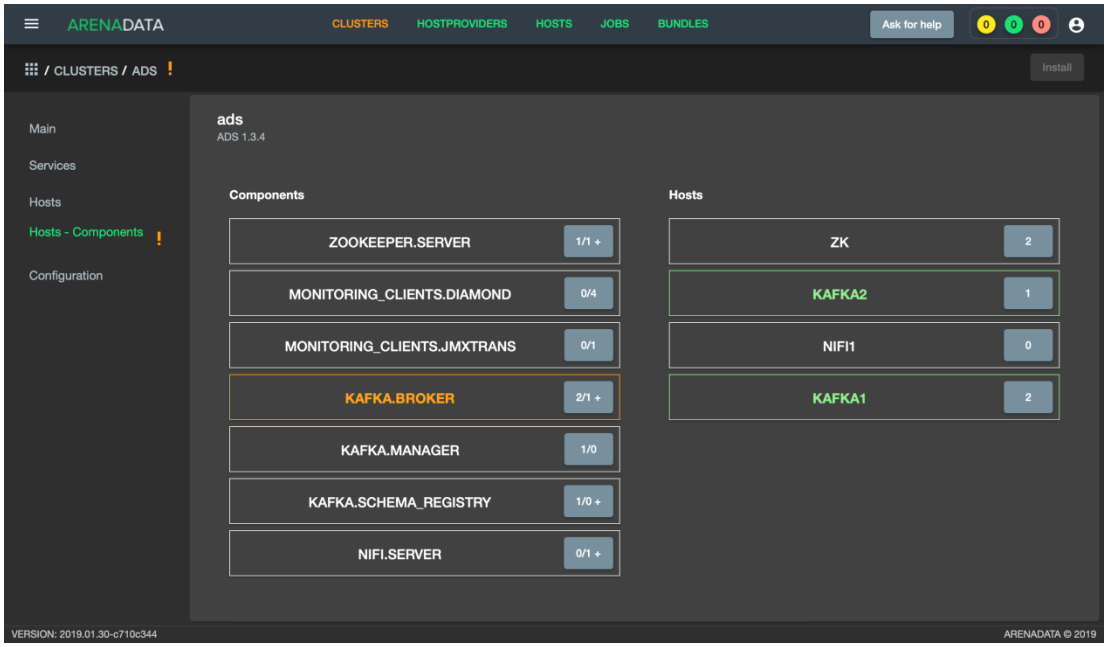


Рис. 18. Компоненты сервиса Kafka

3. Компоненты сервиса Nifi (Рис.19.):

- nifi.SERVER – необходимо добавить на один или более хостов (df-mdw);

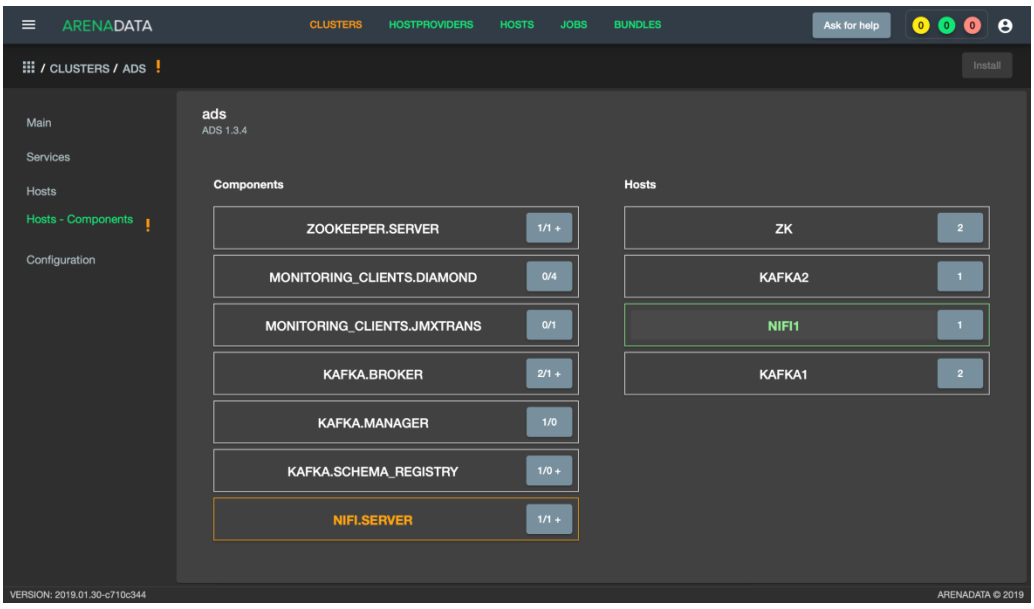


Рис. 19. Компоненты сервиса Nifi

4. Компоненты сервиса monitoring clients (Рис.20.):

- monitoring_clients.diamond – должен быть добавлен на все хосты (zk1, kafka1, kafka2, nifi2). Собирает системные метрики и отправляет их в ADCM;
- monitoring_clients.jmxtrans – необходимо добавить на один хост (zk1). Собирает специфичные для сервисов метрики и отправляет их в ADCM;

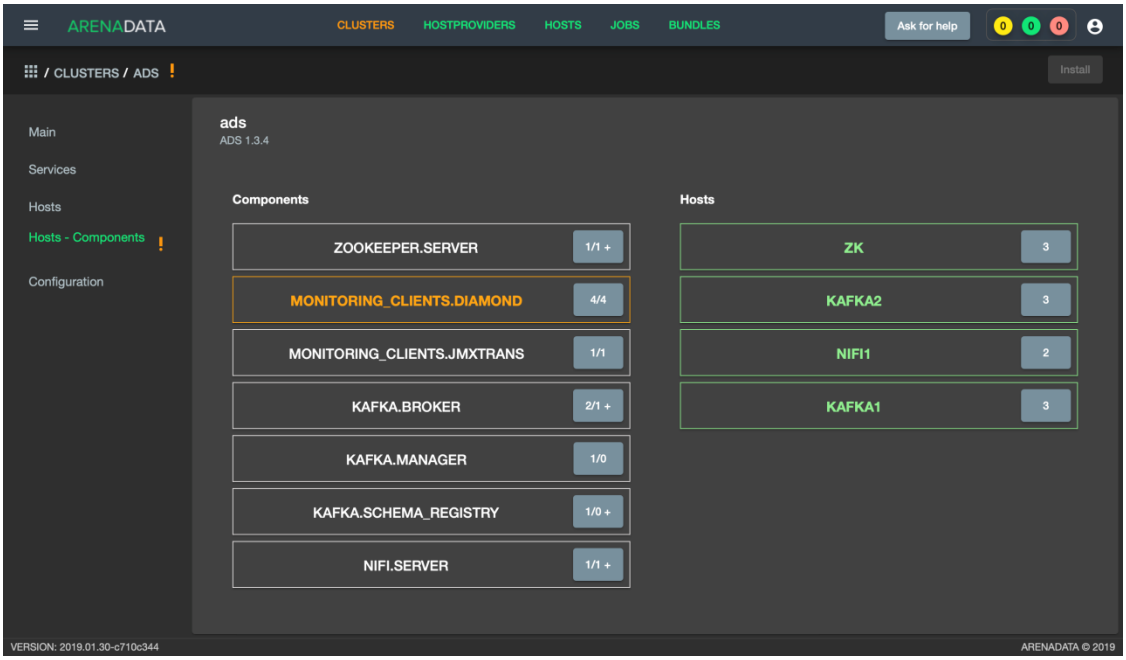


Рис. 20. Компоненты сервиса Monitoring Clients

2.2.6. Способы установки сервисов

Существует два способа установки сервисов:

- Установка всех сервисов кластера. Операция на уровне кластера, представляет собой последовательный вызов процессов установки и конфигурирования всех добавленных сервисов;
- Выборочная установка сервиса. Операция на уровне сервиса, представляет собой установку отдельно взятого сервиса.

НЕЗАВИСИМО ОТ ВЫБРАННОГО ТИПА УСТАНОВКИ ИЗНАЧАЛЬНО НЕОБХОДИМО ИМПОРТИРОВАТЬ НАСТРОЙКИ ИЗ КЛАСТЕРА MONITORING, В СЛУЧАЕ ЕСЛИ ДОБАВЛЕН СЕРВИС MONITORING CLIENTS

2.2.6.1. Установка всех сервисов кластера

Для установки всех добаленных сервисов в кластере ads необходимо выбрать соответствующий кластер в ADCM и нажать кнопку Install в правом вехнем углу экранной формы (Рис.21.).

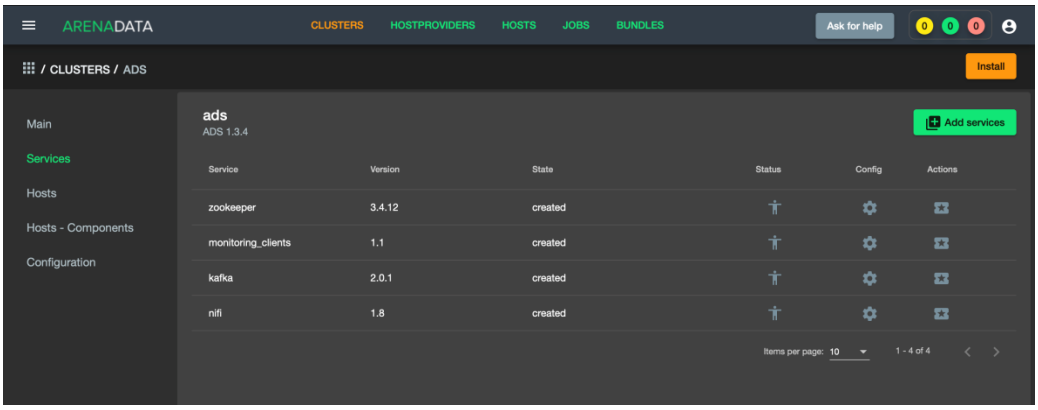


Рис. 21. Установка всех сервисов кластера

По результатам установки все добавленные сервисы меняют состояние с created – создан, на installed – установлен (Рис.22.).

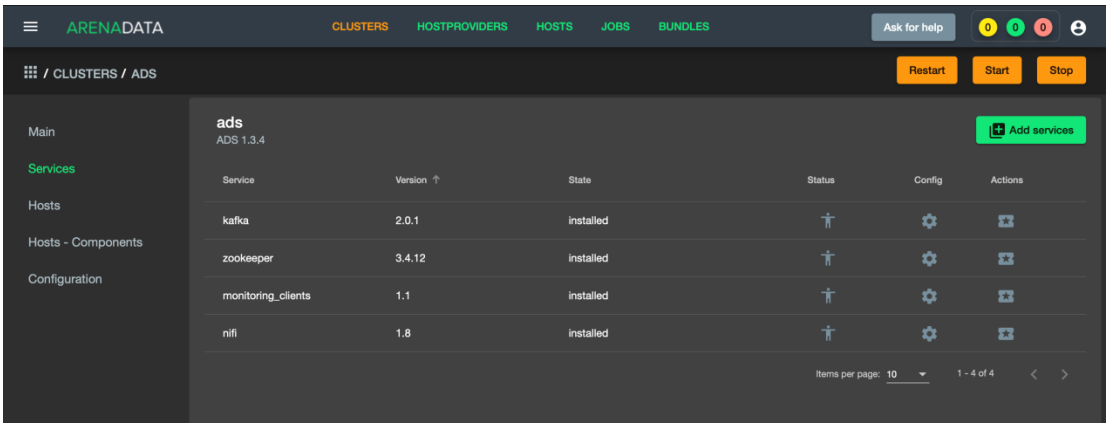


Рис. 22. Установка всех сервисов кластера

2.2.6.2. Установка сервиса Zookeeper

Для установки сервиса Zookeeper на вкладке кластера “Services” необходимо выполнить:

- В поле “Actions” нажать на пиктограмму в строке сервиса Zookeeper и выбрать действие Install (Рис.23.). Производится настройка хостов, установка необходимых пакетов.

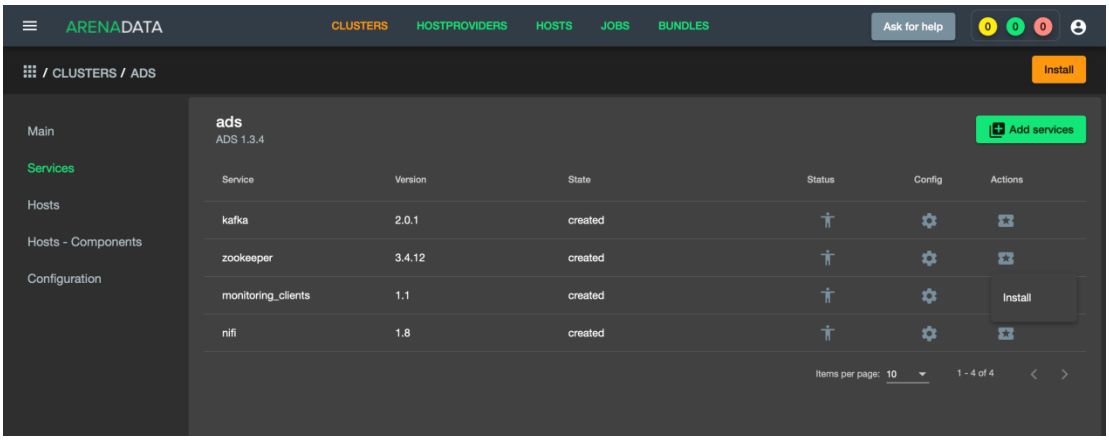


Рис. 23. Install Zookeeper

- По результатам инсталляции сервис Zookeeper меняет состояние с created – создан, на installed – установлен.

2.2.6.3. Установка сервиса Kafka

Для установки сервиса Kafka на вкладке кластера “Services” необходимо выполнить:

- В строке сервиса Kafka в поле “Actions” нажать на пиктограмму и выбрать действие Install (Рис.24.).

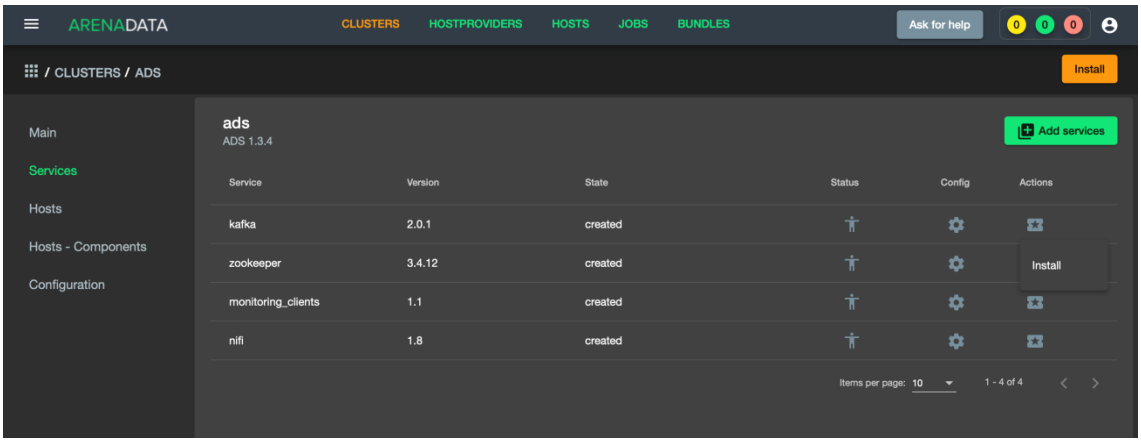


Рис. 24. Install Kafka

- По результатам инсталляции сервис Kafka меняет состояние с created – создан, на installed – установлен.

2.2.6.4. Установка сервиса Nifi

Для установки сервиса Nifi на вкладке кластера “Services” необходимо выполнить:

- В строке сервиса Nifi в поле “Actions” нажать на пиктограмму и выбрать действие Install (Рис.25.).

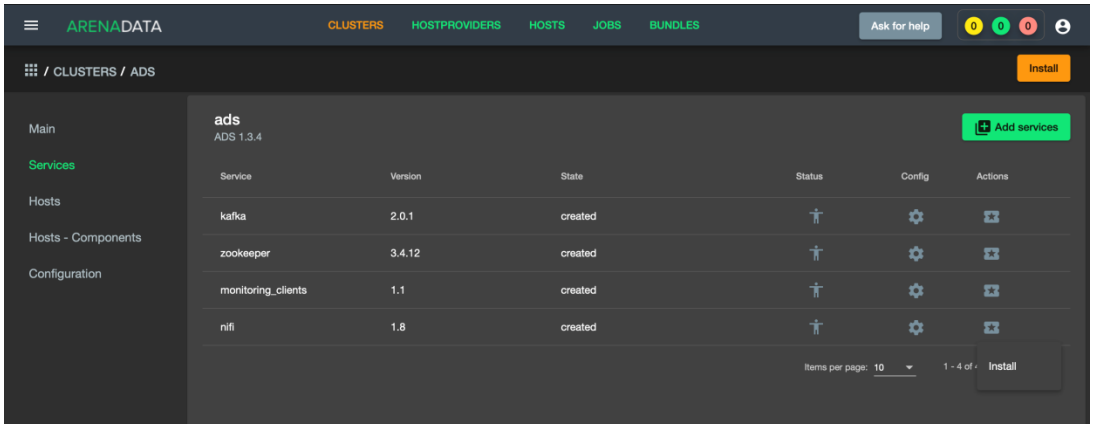


Рис. 25. Install Nifi

- По результатам инсталляции сервис Nifi меняет состояние с created – создан, на installed – установлен.

2.2.6.5. Установка сервиса Monitoring Clients

Сервис Monitoring Clients требует импорта конфигурационных параметров кластера мониторинга (адреса, логин/пароль) в кластер ads:

1. Для импорта конфигурации мониторинга в кластер ads необходимо открыть в ADCM вкладку “CLUSTERS”, выбрать опцию Import и отметить импортируемые настройки сервисов с помощью простановки флажков в открывшейся форме (Рис.26.).

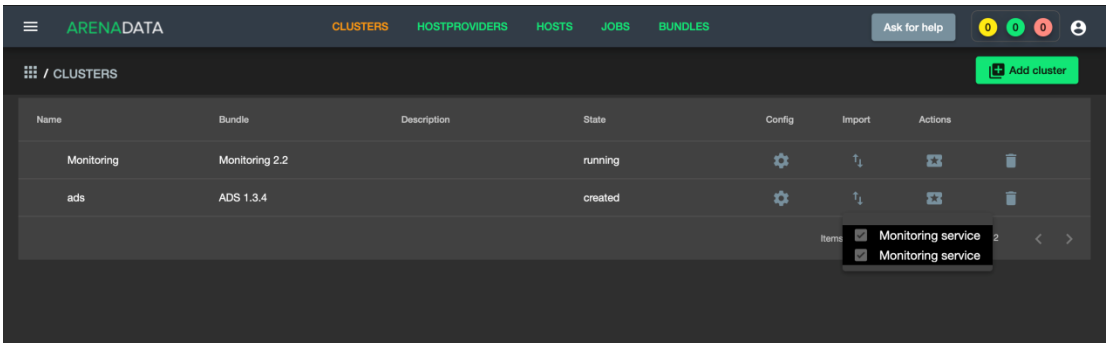


Рис. 26. Импорт конфигурации мониторинга

2. Установка клиентов мониторинга в кластер ads:

- В кластере ads на вкладке “Services” в поле “Actions” нажать на пиктограмму и выбрать действие Install для службы monitoring clients (Рис.27.).

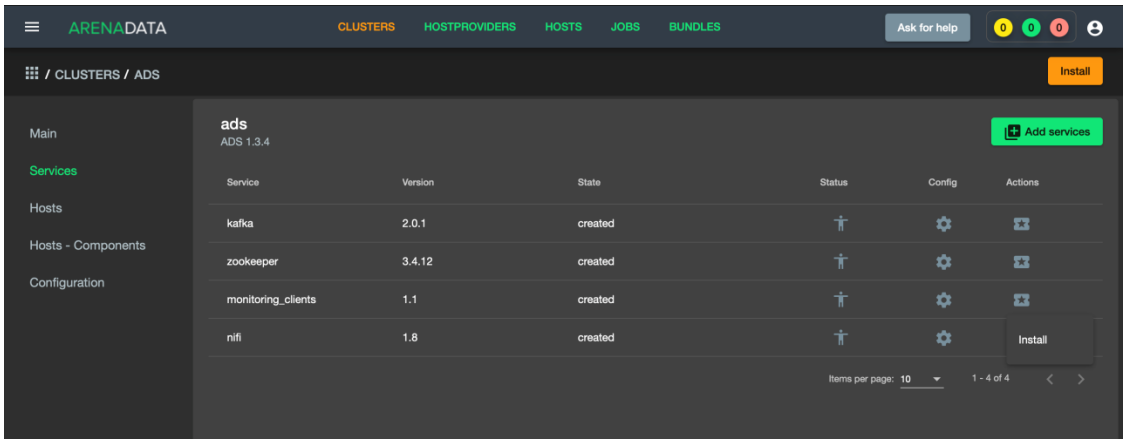


Рис. 27. Установка клиентов мониторинга

- По результатам инсталляции служба monitoring clients меняет состояние с created – создана, на monitored – мониторится.

2.3. Инструменты управления

В ADS предусмотрено 2 вида операций:

- Операции на уровне кластера – предполагают выполнение операций последовательно над всеми сервисами;
- Операции на уровне сервиса – предполагают выполнение операций над отдельным сервисом.

2.3.1. Операции на уровне кластера

Запуск и остановка ADS – существует возможность последовательного запуска и остановки всех сервисов кластера ads через ADCM. Для этого необходимо открыть в ADCM кластер ads и нажать кнопку Start, Stop или Restart в зависимости от требуемой работы с кластером (Рис.28).

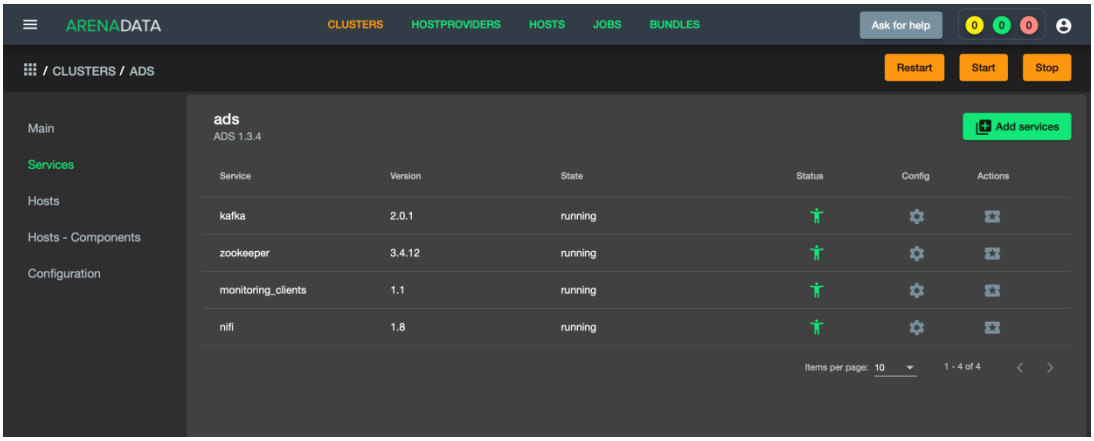


Рис. 28. Операции на уровне кластера

В результате того или иного действия кластер меняет свое состояние на соответствующее – running или stopped, в том случае если кластер уже проинсталлирован и не находится в состоянии created.

2.3.2. Операции на уровне сервиса

Для каждого из сервисов доступна возможность проверки его работоспособности, а также управления им независимо от остальных.

2.3.2.1. Проверка состояния сервиса Zookeeper

Проверка работоспособности сервиса Zookeeper представляет собой подключение к кворуму Zookeeper, создание в нем тестовой znode и проверку доступности созданной znode каждому из хостов кворума.

Для проверки результатов действий над сервисом Zookeeper необходимо:

1. Перейти в кластер ads в ADCM, на вкладке “Services” для сервиса Zookeeper в поле “Actions” нажать на пиктограмму и выбрать действие Check (Рис.29).

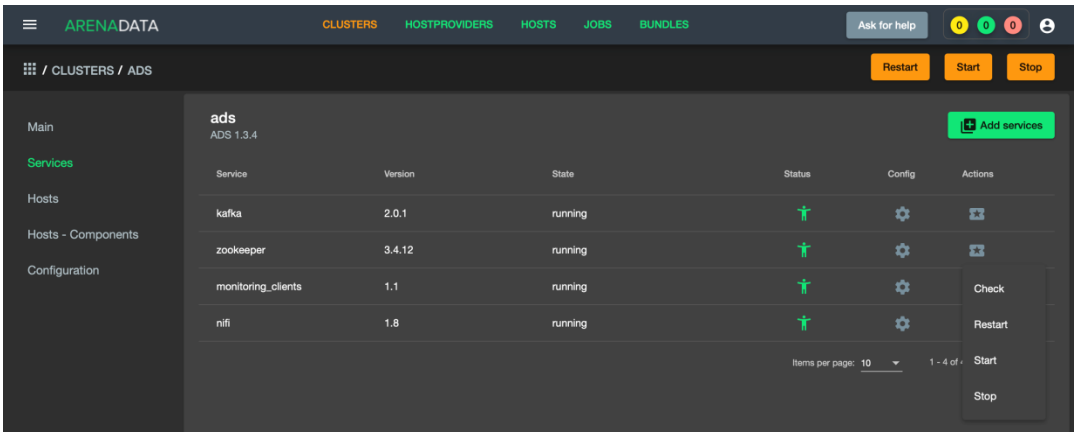


Рис. 29. Запуск проверки состояния сервиса Zookeeper

2. Открыть вкладку “JOBS” (Рис.30).

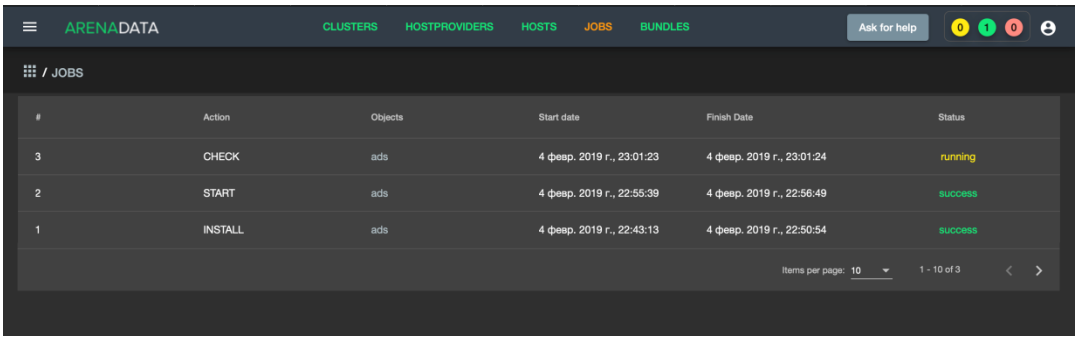


Рис. 30. Вкладка “JOBS”

3. Выбрать последнее действие над кластером ads и в открывшемся окне проверить результаты (Рис.31).

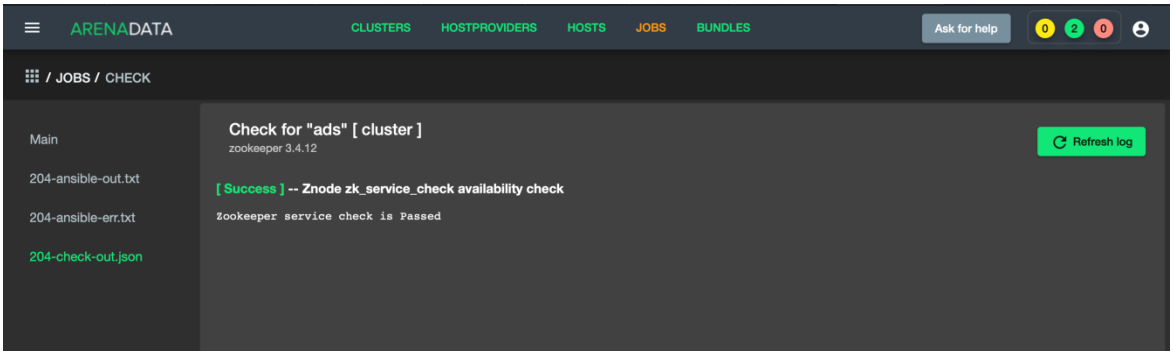


Рис. 31. Проверка состояния сервиса Zookeeper

2.3.2.2. Проверка состояния сервиса Kafka

Проверка работоспособности сервиса Kafka представляет собой создание тестовых топиков на каждом из хостов BROKER и проверку доступности созданных топиков.

Для проверки результатов действий над сервисом Kafka необходимо:

1. Перейти в кластер ads в ADCM, на вкладке “Services” для сервиса Kafka в поле “Actions” нажать на пиктограмму и выбрать действие Check (Рис.32).

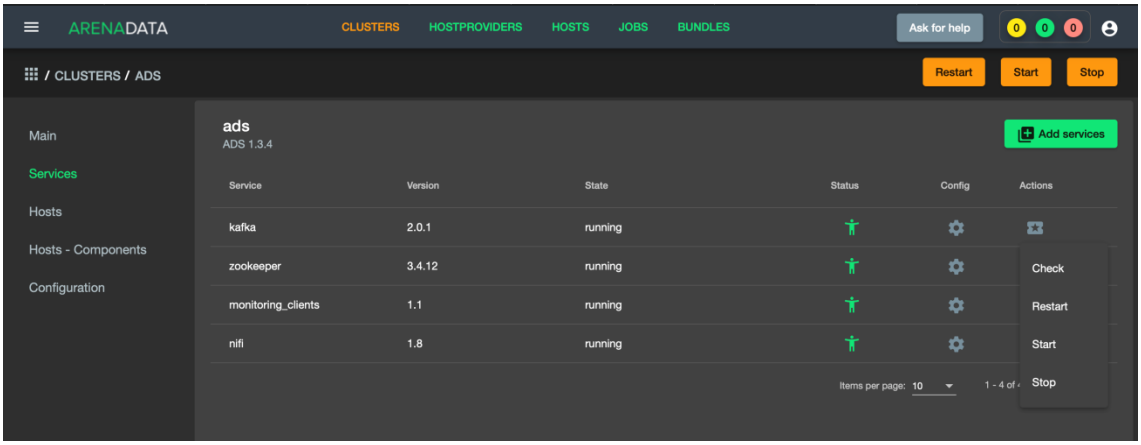


Рис. 32. Запуск проверки состояния сервиса Kafka

2. Открыть вкладку “JOBS” (Рис.33).

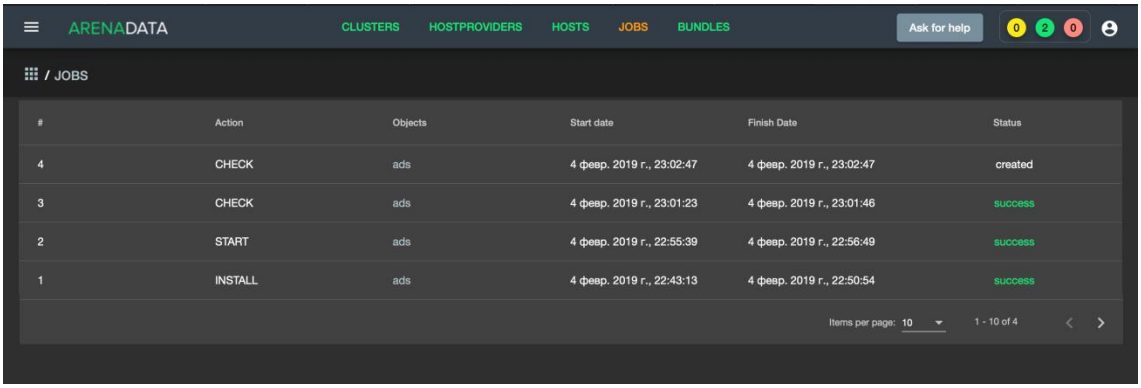


Рис. 33. Вкладка “JOBS”

3. Выбрать последнее действие над кластером ads и в открывшемся окне проверить результаты (Рис.34).

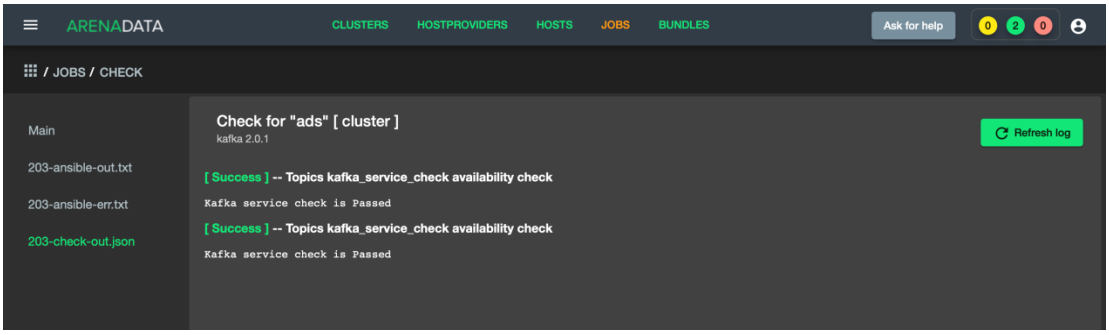


Рис. 34. Проверка состояния сервиса Kafka

2.3.3. Запуск и остановка сервисов

Для каждого из сервисов есть возможность управления им независимо от остальных, выполнять такие операции как Stop, Start, Restart.

Например, для того чтобы перезапустить все компоненты сервиса Kafka необходимо перейти в кластер ads в ADCM, на вкладке “Services” для сервиса Kafka в поле “Actions” нажать на пиктограмму и выбрать действие Restart (см. Рис.32).

В результате того или иного действия статус сервиса меняется на running или stopped, в том случае если сервис уже был проинсталлирован и не находится в состоянии created.

3. УСТАНОВКА ADS С ПОМОЩЬЮ AMBARI

Установка сервера Ambari 2.7.0 проходит в несколько этапов:

- Настройка удаленного репозитория YUM;
- Установка сервера Ambari;
- Настройка сервера Ambari;
- Запуск сервера Ambari.

3.1. Настройка удаленного репозитория YUM

Настройка удаленного репозитория не отличается от настройки любого дополнительного репозитория YUM. Для добавления репозитория необходимо выполнить от имени root команду:

```
yum-config-manager --add-repo http://public-repo-
1.hortonworks.com/ambari/centos7/2.x/updates/2.7.0.0/ambari.repo
```

3.2. Установка сервера Ambari

Сервер Ambari устанавливается из RPM-пакета по команде YUM:

```
yum install ambari-server
```

Команда устанавливает сервер Ambari, являющийся сервером веб-приложений, на порт 8080. Также устанавливается инстанс сервера PostgreSQL на порт 5432.

3.3. Настройка сервера Ambari

Сервер Ambari необходимо настроить для корректной работы. В случае если инстанс PostgreSQL настроен на порт по умолчанию, следует выполнить следующую команду:

```
ambari-server setup
```

В процессе настройки необходимо указать или принять по умолчанию параметры:

- Учетная запись пользователя – для запуска Ambari-сервера можно выбрать любую учетную запись (необязательно выполнять вход от root). В случае если пользователя не существует, он создается автоматически;
- Java JDK – для загрузки Oracle JDK 1.8 необходимо ввести значение 1 и принять лицензию Oracle JDK для загрузки файлов из Oracle. При этом установка JDK выполняется автоматически;
- База данных – выбор базы данных:

```
Enter advanced database configuration
```

В командной строке необходимо ответить n или y:

- n – для использования с Ambari стандартной встроенной базы данных PostgreSQL. По умолчанию для базы данных PostgreSQL устанавливается имя ambari и логин/пароль принимают значения ambari/bigdata.
- y – при необходимости использования с Ambari уже существующей базы данных PostgreSQL, MySQL или Oracle вместо предлагаемой по умолчанию. Далее для выбранной базы данных необходимо указать параметры подключения.

3.4. Запуск сервера Ambari

После установки сервера Ambari запуск осуществляется по команде:

```
ambari-server start
```

Для проверки статуса сервера необходимо использовать команду:

```
ambari-server status
```

Для остановки сервера необходимо использовать команду:

```
ambari-server stop
```

Сервер Ambari доступен на порту 8080. По умолчанию для него установлена следующая учетная запись:

- User: admin
- Password: admin

Для входа в веб-интерфейс Ambari необходимо в адресной строке браузера указать адрес сервера:

```
http://<адрес сервера>:8080
```

При этом запрашивается логин и пароль. После авторизации открывается веб-интерфейс Ambari (Рис.35.).

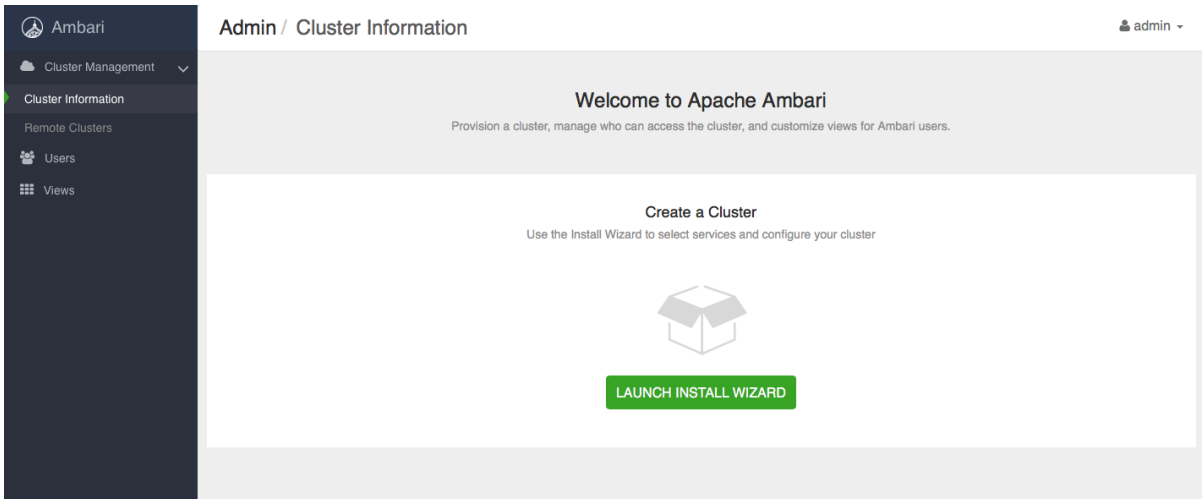


Рис. 35. Веб-интерфейс Ambari до настройки кластера

4. ПОДГОТОВКА К УСТАНОВКЕ ARENADATA STREAMING MPACK

Основные компоненты ADS устанавливаются с помощью Ambari Management Pack (MPack):

- Загрузка Arenadata Streaming MPack;
- Импортирование MPack в Ambari стек.

4.1. Загрузка Arenadata Streaming MPack

Для инсталляции Arenadata Streaming MPack необходимо загрузить и распаковать следующие архивы в выделенном для них месте (при этом следует избегать использования каталога /tmp):

- ads-ambari-mpack-{version}.tar.gz – набор конфигураций сервисов и скриптов для развертывания и настройки платформы через Ambari.

В случае если архивы загружены в каталог /tmp, то для их распаковки в каталоге, например, /staging, необходимо выполнить следующую команду:

```
tar –xzvf /tmp/ads-ambari-mpack-{version}.tar.gz -C /staging/
```

4.2. Импортирование MPack в Ambari стек

Для импортирования стека ADS в Ambari необходимо выполнить следующие команды:

1. Сохранить текущую папку Ambari Resources для возможного восстановления системы:

```
cp -r /var/lib/ambari-server/resources /var/lib/ambari-
server/resources.backup
```

2. Запустить команду импорта стека ADS:

```
ambari-server install-mpack --mpack=/staging/ads-ambari-mpack-
{version}.tar.gz --purge --verbose
```

3. Перезапустить Ambari:

```
ambari-server restart
```