

Эксплуатация Arenadata Streaming (Kafka, NiFi)



Подготовка окружения к установке ADS, ADS Control: ZooKeeper, Apache Kafka, Apache NiFi, ADS Control

Agenda

- Планирование кластера:
Типы узлов. Профили нагрузки. Аппаратное обеспечение. Примеры кластеров.
- Настройка параметров окружения:
Подготовка к установке. Настройка сети и DNS. Монтирование и настройка файловой системы. Лабораторная работа.
- Введение в Arenadata Cluster Manager:
Основные понятия, компоненты и возможности. Развертывание системы управления.
- Установка и настройка ADS:
*Планирование и первичная конфигурация кластера. Мониторинг. Установка Arenadata Streaming. Конфигурирование сервисов.
Лабораторная работа.*
- Мониторинг ADS:
Архитектура кластера мониторинга. Мониторинг ADS: Kafka, NiFi. Встроенный мониторинг NiFi.
- Установка и настройка ADS Control (ADSC):
*Установка Arenadata Enterprise Tools. Развертывание ADSC с использованием ET.
Лабораторная работа.*

Планирование кластера

Узлы кластеров и планирование ролей




Типы узлов

- **Cluster Nodes:**
 - Kafka-брокер
 - Kafka Streams, KSQL
 - Kafka REST Proxy, Kafka Manager, Schema Registry, Connect
 - NiFi, NiFi-Registry
- **Edge Node (Utility, Gateway, Clients, ET)** – другие кластерные процессы (кворумность, внешние СУБД, ...), клиентские точки доступа для запуска задач на кластере.
- **Platform Security Node** (Ranger, Knox, Solr, Zookeeper, ...)





Варианты размещения

- Приоритетнее всего использовать физические серверы.
- Arenadata Streaming работает как на bare metal, так и в облаке.
- При обеспечении сетевой доступности между серверами возможны также следующие сценарии:
 - гетерогенная ИТ-инфраструктура;
 - Multi-clouds;
 - распределенная установка на разные инфраструктуры.

Узлы кластеров и планирование ролей

	Cluster Nodes	Edge Nodes	Platform Security Nodes	ET Nodes
ADCM				
Monitoring				
External DB's (PostgreSQL, MySQL,...)				
Arenadata Streaming				
ADS Control				
Arenadata Platform Security				
Enterprise Tools				

Узлы кластеров и планирование ролей (ADS 3.6.2+)

	Cluster Nodes	Edge Nodes	Platform Security Nodes	ET Nodes
ADCM				
External DB's (PostgreSQL, MySQL,...)				
Arenadata Streaming	   			
ADS Control				
Arenadata Platform Security				
Enterprise Tools				

Шаблоны рабочей нагрузки для ADS/ADPS

ADS:

- Minimal
- Proof of concept
- Commodity
- Balanced
- Compute-intensive

ADPS:

- Minimal – минимальная конфигурация (pilot).
- Typical – сбалансированная нагрузка для малых и больших систем.

[Аппаратные требования к серверам](#)

ZooKeeper, Kafka Manager, Schema Registry		Kafka Broker	
Storage: <ul style="list-style-type: none">• OS: RAID 1• Data: SSD RAID 1• Volume: 1 TB + CPU: 8 RAM: 32 GB		Storage: <ul style="list-style-type: none">• OS: RAID 1• Data: SSD RAID 10• Volume: 1 TB +/- CPU: 16 RAM: 64 GB,	
ksqlDB		REST Proxy	
Storage: <ul style="list-style-type: none">• Data: SSD CPU: 4 RAM: 32 GB		Storage: <ul style="list-style-type: none">• Data: HDD CPU: 16 RAM: 4+ GB,	
NiFi Nodes		Platform Security Nodes	
Storage: <ul style="list-style-type: none">• OS: RAID 1• Data: SSD RAID 10• Volume: 1 TB + CPU: 24 RAM: 96 GB		Storage: <ul style="list-style-type: none">• OS: RAID 1• Data: RAID 10 (SSD)• Volume: 1 TB +/- CPU: 16+ RAM: 128+ GB, из расчета 1 CPU ~ 8 GBs	

Настройка параметров окружения

Arenadata Streaming, Arenadata Platform Security

Компонент	Требования
Операционная система	Centos 7.6.1810; RHEL 7.6; AltLinux sp 8.4—доступно только в Enterprise-версии; Astra Linux 1.7 "Опел" SE с Axiom JDK—доступно только в Enterprise-версии; RedOS 7.3.2—доступно только в Enterprise-версии, Ubuntu 22.04.
Браузеры, поддерживающие сервис NiFi	Chrome FireFox Edge Safari Пользовательский интерфейс поддерживается в текущей стабильной версии этих браузеров и в предыдущей
Java	Java 17(ADS 3.6.2+), Java 8 (ADS 3.3.2<)

Доступность портов для ADS



Сервис	Порт	Назначение
Общие	22	Порт для подключения к SSH-провайдеру
	81	TCP-порт Arenadata Enterprise Tools (доступ к репозиториям в случае offline-инсталляции)
	8000	TCP-порт для отправки статусов компонентов кластера в ADCM
	2015	TCP-порт для отправки метрик на сервер мониторинга
	2016	UDP-порт для отправки метрик на сервер мониторинга
ZooKeeper	2181	Порт доступа к сервису
	2888 3888	Порты межсерверного взаимодействия для компонентов кворума
Kafka	9092	HTTP-порт доступа к сервису
	9093	HTTPS-порт доступа к сервису
	9999	JMX-порт для доступа к метрикам сервиса
Kafka-Manager	9898	Порт доступа к сервису
Kafka REST Proxy	8082	Порт доступа к сервису
	9998	JMX-порт для доступа к метрикам сервиса
Schema-Registry	8081	Порт доступа к сервису
	9997	JMX-порт для доступа к метрикам сервиса
ksqlDB	8088	Порт доступа к компоненту ksqlDB Server
NiFi	9090	HTTP-порт доступа к UI сервиса
	9091	HTTPS-порт доступа к UI сервиса
	11443	Порт взаимодействия между серверами NiFi, объединенными в кластер
	18080	Порт доступа к UI компонента NiFi-Registry

Системные требования ADPS



Компонент	Требования
Операционная система	Ubuntu 22.04 RedHat 7.x. CentOS 7.x. ALT Linux 8.4 SP. AstraLinux 1.7 "Опел" SE с Axiom JDK. Параметру LC_TIME должно быть присвоено значение en_US.utf8. JDK 8+ (compatible with OpenJDK and Oracle JDK).
СУБД, внешняя	MariaDB 5.5.68, PostgreSQL (9.4+)
Ranger Admin	RAM 1,5 Gb+
Solr	Хранение индексов Ranger Admin выделено достаточно ресурсов (зависит количества узлов)

Доступность портов для ADPS



Сервис	Компонент/Сервер	Порт	Протокол	Описание
ADCM	ADCM	22	TCP	Порт подключения ADCM
Arenadata PostgreSQL	Arenadata PostgreSQL	5432	TCP	Порт для подключения к PostgreSQL
Knox	Knox Gateway	8443	HTTPS	Порт шлюза
MariaDB	MariaDB Master Server	3306	TCP	Порт для подключения к MariaDB
Ranger	Ranger Admin	6080/6182	HTTP/HTTPS	Порт для доступа к API и веб-интерфейсу Ranger Admin
Ranger	Ranger Admin	6085	TCP (локальный)	Порт Ranger Shutdown
Ranger	Ranger Usersync	5151	TCP/SSL	Порт для сервиса UNIX Auth
Ranger	Ranger KMS	9292/9393	HTTP/HTTPS	Порт для Ranger KMS
Ranger	Ranger KMS	7085	TCP (локальный)	Порт Ranger KMS Shutdown
Solr	Solr Server	8983	HTTP	Порт для доступа к API и веб-интерфейсу Solr
Solr	Jetty	7983	TCP (локальный)	Порт Jetty Shutdown для Solr
Zookeeper	Zookeeper	2888	TCP	Порт для координирования кластера (только для внутреннего использования в кластере)
Zookeeper	Zookeeper	3888	TCP	Порт для координирования кластера (только для внутреннего использования в кластере)
Zookeeper	Zookeeper	2181	TCP	Порт для клиентских соединений
Zookeeper	Zookeeper (Admin Server)	5181	HTTP/HTTPS	HTTP-интерфейс для доступа к Zookeeper REST API

Подготовка к установке ADS/ADPS

- Настройка беспарольного доступа к узлам по SSH
- Проверка аппаратной конфигурации всех узлов и версии ОС:

Команды для проверки:

`cat /etc/*-release, lsblk, lscpu, free -m` и т.д.

- Проверка списка всех РЕПО (по умолчанию):

Команда для проверки:

`yum repolist all` (CentOS), `cat /etc/apt/sources.list` (Ubuntu)

- Синхронизации времени на узлах друг с другом. Для этого следует включить службу NTP и убедиться, что синхронизация происходит автоматически.

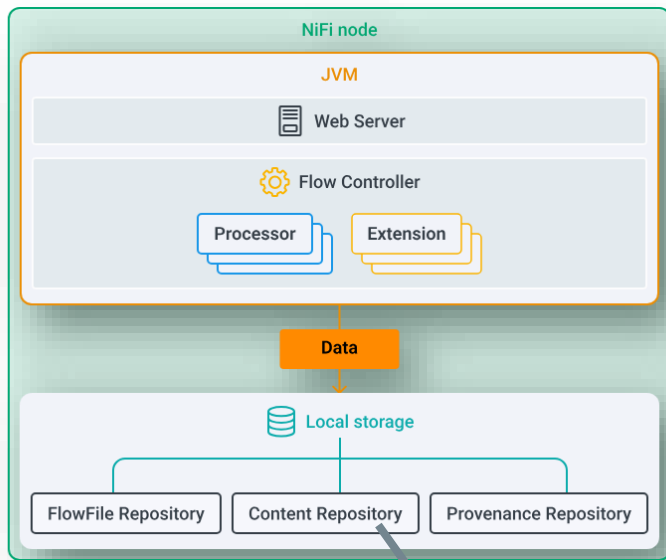
Команды для проверки:

`timedatectl status, ...`

Подготовка файловой системы

- Рекомендованы к применению Ext4, XFS
- Ext4 is supported up to 50TB
- XFS up to 500TB
- Для разделов, обрабатывающих различные репозитории NiFi, отключаем запись информации о последнем времени доступа (atime) при каждом чтении файла
- В файле /etc/fstab (который используется для настройки параметров монтирования различных блочных устройств, разделов на диске и удаленных файловых систем) необходимо добавить для интересующих разделов параметр noatime
- Для хранения данных Kafka:
 - Не рекомендуется использовать одни и те же диски, используемые для данных Kafka, совместно с журналами приложений или другой активностью файловой системы ОС.
 - Если вы настраиваете несколько каталогов данных, брокер помещает новую партицию по пути с наименьшим количеством сохраненных партиций. Каждая партиция будет полностью находиться в одном из каталогов данных. Если данные не сбалансированы между партициями, это может привести к дисбалансу нагрузки между дисками.
 - Массив дисков RAID 10 рекомендуется как лучший вариант для большинства случаев использования. Он обеспечивает улучшенную производительность чтения и записи, защиту данных.
 - Не рекомендуется использовать сетевые хранилища (Network Attached Storage, NAS). NAS часто медленнее, показывает большие задержки с более широким отклонением средней задержки и является единой точкой отказа.

Подготовка файловой системы для NiFi. Масштабируемость



```
base path
  |__Container 0
    |__Section 1
      |__identifier
      |__offset
    |__Section 2
      |__identifier
      |__offset
    |_____ ...
  |__Container 1
    |__Section 1
    |__Section 2
    |__ ...
  |__...
```

Nodes	Data rate/sec	Events/sec	Data rate/day	Events/day
1	192.5 MB	946,000	16.6 TB	81.7 Billion
5	881 MB	4.97 Million	76 TB	429.4 Billion
25	5.8 GB	26 Million	501 TB	2.25 Trillion
100	22 GB	90 Million	1.9 PB	7.8 Trillion
150	32.6 GB	141.3 Million	2.75 PB	12.2 Trillion

<https://blog.cloudera.com/benchmarking-nifi-performance-and-scalability/>

Throughput Target	Number of NiFi nodes	CPU Cores/node	Number of disks/node, size of each disk (RAID 5/10)	RAM/node	Ideal Networking Setup
50 MB/s, 1000 events/s	3	16+	6+, 1TB	8+ GB	1 Gigabit bonded NICs
100 MB/s, 10,000 events/s	5	16+	6+, 2TB	8+ GB	1 Gigabit bonded NICs
200 MB/s, 100,000 events/s	7	24+	12+, 4TB	16+ GB	10 Gigabit bonded NICs
400 MB/s, 100,000+ events/s	9	24+	12+, 8TB	16+ GB	10 Gigabit bonded NICs

https://docs.cloudera.com/HDPDocuments/HDF3/HDF-3.1.1/bk_planning-your-deployment/content/ch_hardware-sizing.html

Подготовка к установке ADS/ADPS (Centos 7)

- Проверка настройки Firewall (IPTables). Необходимо либо настроить сетевые фильтры/файрволлы для беспрепятственного пропуска трафика служб, либо попросту отключить сетевые фильтры внутри контура кластера ([карта портов ADS](#), [карта портов ADPS](#)):

Команды для проверки:

```
systemctl status firewalld, systemctl disable firewalld, systemctl stop firewalld
```

- Отключение протокола IPv6, параметры в файле /etc/sysctl.conf:

```
net.ipv6.conf.all.disable_ipv6 = 1
```

```
net.ipv6.conf.default.disable_ipv6 = 1
```

- CentOS:

- Работа с SELinux не поддерживается, поэтому необходимо его отключить:

Команда:

```
setenforce 0
```

- Далее необходимо установить значение SELINUX в файле /etc/selinux/config:

```
SELINUX = disabled
```

```
SELINUXTYPE=targeted
```

Настройка DNS и NSCD

- Все узлы в кластере должны быть настроены как для прямого, так и для обратного DNS и разрешаться по FQDN (fully qualified domain name):

Команды для проверки:

```
ping localhost (resolve loopback ipv4-address)
ping <hostname> (resolve non-loopback ipv4-address)
hostname, hostname -f
dig -x <ip> (CentOS)
nslookup <ip> (Ubuntu)
```

- В случае если не получается настроить DNS, необходимо отредактировать файл /etc/hosts на каждом узле кластера так, чтобы он содержал IP-адреса и FQDN's каждого узла:

```
10.122.111.11 ads-d-XX-node-1.ru-central1.internal node-1
```

(шаблон: <ip> <fqdn> <alias hostname>)

- На всех узлах кластера должны быть настроено прямое/обратное разрешение DNS для MIT/LDAP-сервера (AuthN+AuthZ).

Настройка MIT Kerberos

- MIT KDC хранит базу данных пользователей для синхронизации с ADPS через principals.
- Kerberos MIT: аутентификация принципалов, ADPS: авторизация
- База данных участников Kerberos MIT должна быть синхронизирована с базой данных пользователей ADPS.
- При работе с кластером ADS все участники получают TGT от KDC Kerberos MIT.
- **Установка сервера KDC**
 - Установить Kerberos server package
 - Настроить /etc/krb5.conf, /var/kerberos/krb5kdc/kdc.conf, /var/kerberos/krb5kdc/kadm5.acl
 - Создать БД Kerberos (*kdb5_util*)
 - Настроить службы и запустить *Kerberos*
 - Создать администратора Kerberos (Kadmin.local -q "addprinc admin / admin")

<https://docs.arenadata.io/ru/ADH/current/tutorials/security/kerberos/MIT/setup-kerberos-mit.html>

Настройка LDAP Active Directory

Кластеры ADS/ADPS могут быть керберизованы через KDC Active Directory (LDAP)

- Установить доменные службы
- Добавить роль **AD Certification Service**
- Создать Root CA
- Экспортировать сертификат в формат X.509
- Создать логическую структуру объектов AD(OU,Users, Groups,...)

<https://docs.arenadata.io/ru/ADH/current/tutorials/security/kerberos/active-directory/setup-ad.html>

Настройка внешней СУБД (PostgreSQL). Опционально

Для Metastore Hive:

```
CREATE USER hive WITH password 'hive';  
CREATE DATABASE "hive" WITH OWNER = hive ENCODING = 'UTF8' TEMPLATE = template0 TABLESPACE = pg_default LC_COLLATE = 'ru_RU.UTF-8'  
LC_CTYPE = 'ru_RU.UTF-8' CONNECTION LIMIT = -1;  
GRANT ALL ON DATABASE hive TO hive;  
GRANT USAGE, CREATE ON SCHEMA public TO hive;
```

Для Ranger Admin:

```
CREATE USER rangeradmin WITH password 'rangeradmin';  
CREATE DATABASE "ranger" WITH OWNER = rangeradmin ENCODING = 'UTF8' TEMPLATE = template0 TABLESPACE = pg_default LC_COLLATE =  
'ru_RU.UTF-8' LC_CTYPE = 'ru_RU.UTF-8' CONNECTION LIMIT = -1;  
GRANT ALL ON DATABASE ranger TO rangeradmin;  
GRANT USAGE, CREATE ON SCHEMA public TO rangeradmin;
```

Для Ranger KMS:

```
CREATE USER rangerkms WITH password 'rangerkms';  
CREATE DATABASE "rangerkms" WITH OWNER = rangerkms ENCODING = 'UTF8' TEMPLATE = template0 TABLESPACE = pg_default LC_COLLATE =  
'ru_RU.UTF-8' LC_CTYPE = 'ru_RU.UTF-8' CONNECTION LIMIT = -1;  
GRANT ALL ON DATABASE rangerkms TO rangerkms;  
GRANT USAGE, CREATE ON SCHEMA public TO rangerkms;
```

Настройка внешней СУБД (PostgreSQL). Опционально

Файл postgresql.conf:

```
max_connections = 1000
```

Настройки для подключения psql:

- rolconnlimit: -1, либо 60+
- datconnlimit: -1, либо 60+

Значение шифрования пароля в postgresql.conf, должно соответствовать выставленному в pg_hba.conf, файл postgresql.conf:

```
password_encryption = md5          # md5 or scram-sha-256
```

Файл pg_hba.conf:

```
host    all         all         0.0.0.0/0         md5
```

Pre-install Checklist

- Доступ SSH по ключу
- Проверка аппаратной конфигурации (OS, Hardware)
- На всех серверах проверить монтирование в **/etc/fstab** и корректность согласно ТЗ
- Доступность репозиторийев
- Синхронизации времени на узлах
- Проверка настройки Firewall (IPTables)
- Отключение IPv6
- Отключение SELinux (CentOS)
- Настроить прямое и обратное разрешение имен
- Проверка конфигурации и доступности KDC (MIT, AD, ...)
- Настройка внешней СУБД. Опционально.

Лабораторная работа

Выполнить подготовку узлов по следующим пунктам настроек:

1. Проверить настройки SELINUX.
2. Отключить IPv6 (требуется перезапуск сетевой службы).
3. Опционально. Внести изменения в файл /etc/hosts на всех узлах.
4. Проверить прямое и обратное разрешение имен на всех узлах (результат для всех узлов должен совпадать).
5. Проверить прямое и обратное разрешение LDAP-сервера (10.129.0.148 winda-test-1.adh-sec.com winda adh-sec.com).

Улучшения совместимости NiFi, Kafka и ОС

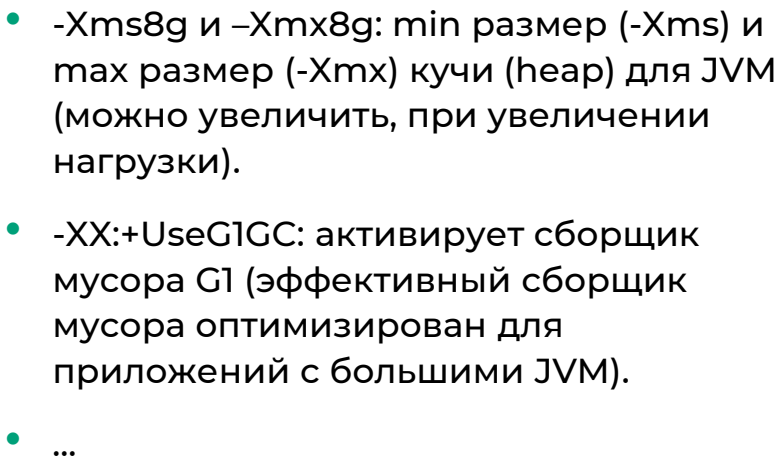
Рекомендации по настройке NiFi

- Увеличение количества файловых дескрипторов
- Увеличение количества разветвленных процессов
- Увеличение количества доступных портов сокетов TCP
- Установка длительности состояния TIMED_WAIT для сокетов
- Отключение параметра подкачки
- ...

Рекомендации по настройке Kafka

- Аппаратное обеспечение
- Программное обеспечение
- Файловые дескрипторы
- Виртуальная память
- ...

- NiFi



② запись/чтение через Cache, но с сохранением на диск

- -Xms6g и -Xmx6g: min размер (-Xms) и max размер (-Xmx) кучи (heap) для JVM (6 - 8 Gb).
- -XX:+UseG1GC: активирует сборщик мусора G1 (эффективный сборщик мусора оптимизирован для приложений с большими JVM).
- -XX:MetaspaceSize=96m: определяет начальный размер мета-пространства (место хранения метаданных классов и методов).
- -XX:MaxGCPauseMillis=20: максимальная пауза сборщика мусора (JVM будет стремиться к сокращению времени сборки мусора до 20 миллисекунд).
- -XX:InitiatingHeapOccupancyPercent=35: процент заполнения кучи для запуска сборки мусора.
- -XX:G1HeapRegionSize=16M: размер региона кучи (heap region) для G1GC.
- -XX:MinMetaspaceFreeRatio=50 и -XX:MaxMetaspaceFreeRatio=80: min и max процент свободного места в мета-пространстве.
- ...

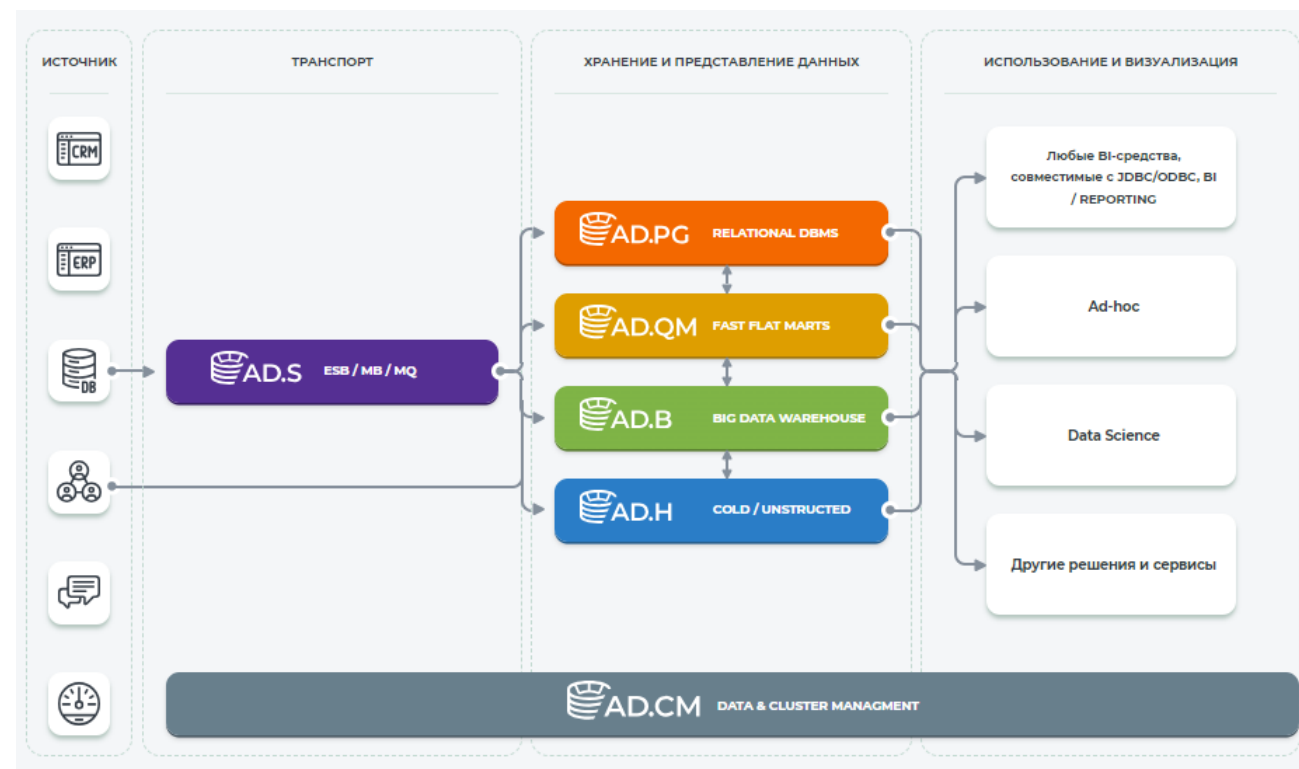
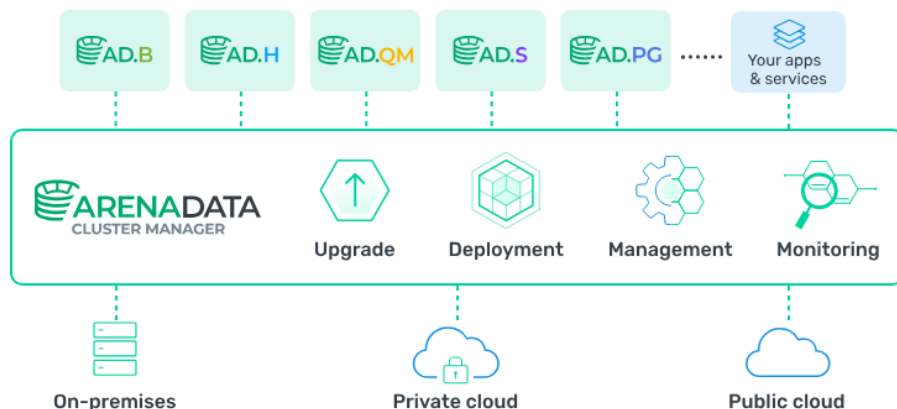
Введение в Arenadata Cluster Manager

Arenadata Cluster Manager



Arenadata Cluster Manager (ADCM) – универсальный оркестратор гибридного ландшафта. Позволяет быстро устанавливать, настраивать все data-сервисы и управлять ими независимо от инфраструктуры.

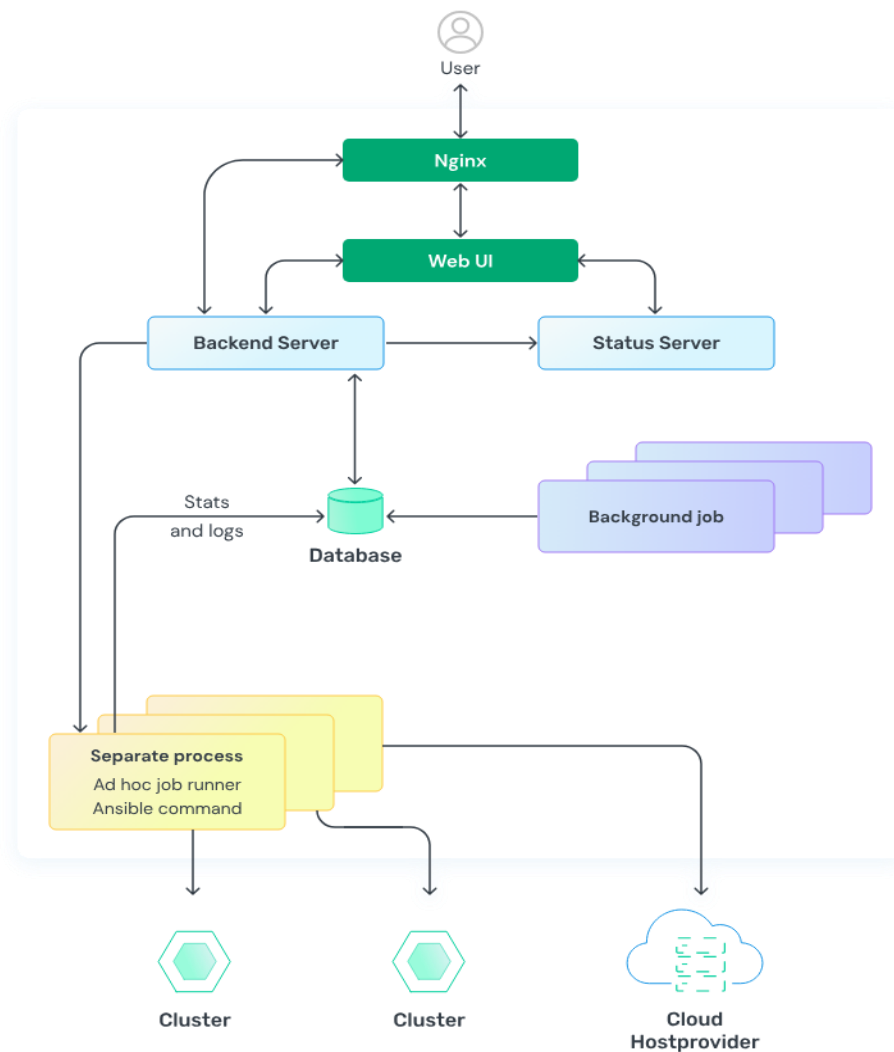
- В Arenadata Cluster Manager установка, настройка и обновление кластеров в оркестраторе производятся по «нажатию кнопки» в веб UI или по запросу в API.
- Все настройки ОС, сервисов, сети, монтирование дисков происходят автоматически.
- ADCM имеет возможность размещать сервисы с гетерогенной инфраструктурой (cloud, on-premises, PaaS).



Возможности ADCM

Возможности работы с сервисами:	Возможности работы с инфраструктурой:	Дополнительные возможности:
<ul style="list-style-type: none">• Установка и настройка• Обновление• Управление• Мониторинг• Настройка прав доступа• Интеграция с другими сервисами• Контейнеризация	<ul style="list-style-type: none">• Создание и удаление виртуальных машин• Конфигурирование ОС• Мониторинг• Управление пользователями• Настройка прав доступа	<ul style="list-style-type: none">• Настройка прав безопасности в ADCM• Открытый ADCM API• Открытый формат для создания новых бандлов - добавьте ваши собственные сервисы/инфраструктуру

Архитектура ADCM



Кластеры ADCM:

- ADS
- ADB
- ADH
- ADPG
- ADQM
- ADPS
- ADSC
- ...

СУБД:

- PostgreSQL
- SQLite

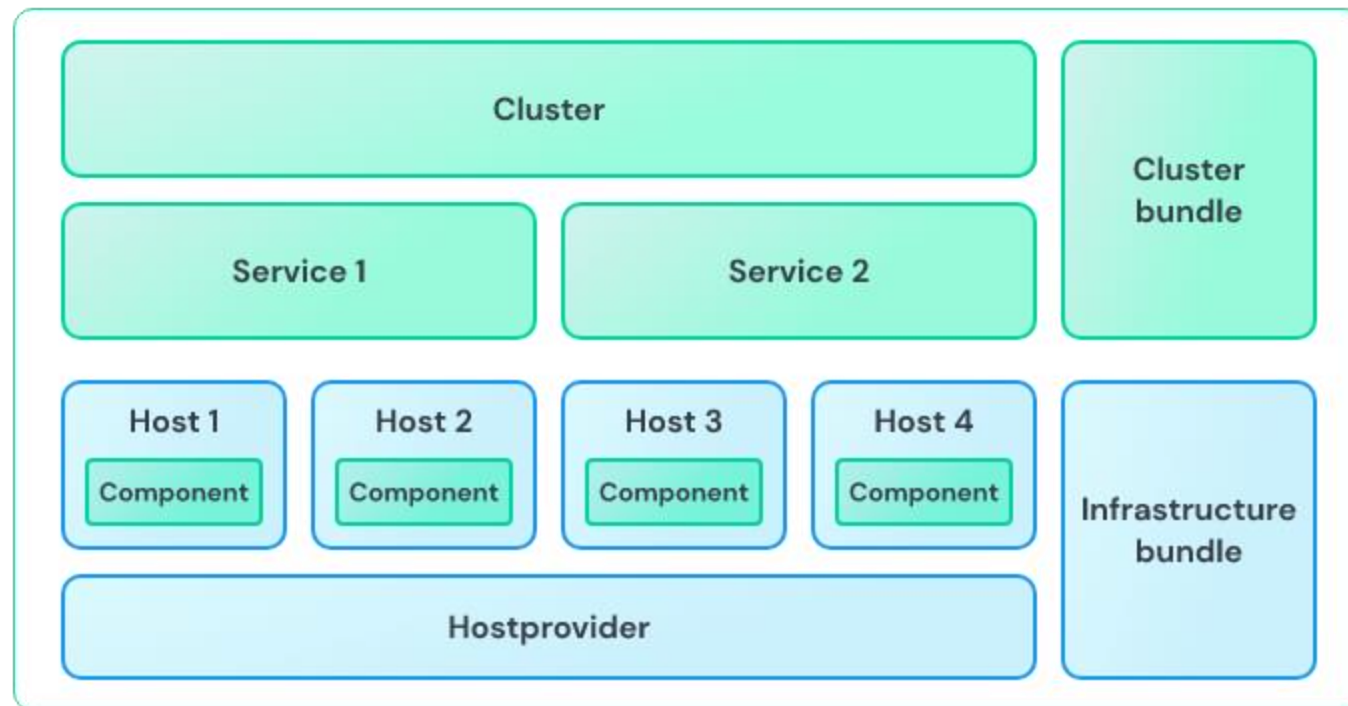
Облачные провайдеры:

- Google Cloud Platform
- Yandex Cloud
- DataFort Cloud
- DigitalEnergy
- ...

Компоненты объекты

Объекты ADCM:

- Bundle
 - Cluster Bundle
 - Infrastructure Bundle
- Host
- Provider
- Service
- Component
- Cluster
- ADCM разворачивается на основе Docker-контейнера
- Поддерживает установку кластера Online, Offline



Развертывание ADCM

- Требования к software и hardware

Компонент	Требования
Операционная система	Centos 7; RHEL 7; AltLinux sp 8; Astra Linux 1.7 "Орел" SE. Ubuntu 22.04
Браузеры, поддерживающие сервис	Chrome FireFox Edge Safari пользовательский интерфейс поддерживается в текущей стабильной версии этих браузеров и в предыдущей
Hardware (минимальные)	CPU: 4 ядра CPU; RAM: 16 ГБ; HDD: 50 ГБ.
SELinux (CentOS)	Отключен
Firewall/IPTables	Отключен
Репозитории	CentOS Extras и CentOS Base
Docker	CentOS/RedHat – docker (docker-ce не подходит) AltLinux – docker-cli, docker-engine, docker-proxy

Развертывание ADCM

- Команды (Ubuntu). Пользователь должен иметь привилегии root

```
sudo apt-get update
sudo apt-get install ca-certificates curl
sudo install -m 0755 -d /etc/apt/keyrings
sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc
sudo chmod a+r /etc/apt/keyrings/docker.asc
sudo apt-get update
sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin
sudo systemctl disable firewalld
sudo systemctl stop firewalld
sudo systemctl start docker
sudo systemctl enable docker
sudo docker pull hub.arenadata.io/adcm/adcm:<version>
sudo docker create --name adcm -p 8000:8000 -v /opt/adcm:/adcm/data hub.arenadata.io/adcm/adcm:<version>
sudo docker update --restart=on-failure adcm
sudo docker start adcm
```

Развертывание ADCM

- Команды (Centos 7)

```
sudo yum update -y
sudo yum install -y yum-utils docker device-mapper-persistent-data lvm2
sudo systemctl disable firewalld
sudo systemctl stop firewalld
sudo sed -i 's/SELINUX=enforcing/SELINUX=disabled/g' /etc/selinux/config
sudo setenforce 0
sudo systemctl start docker
sudo systemctl enable docker
sudo docker pull hub.arenadata.io/adcm/adcm:<version>
sudo docker create --name adcm -p 8000:8000 -v /opt/adcm:/adcm/data hub.arenadata.io/adcm/adcm:<version>
sudo docker update --restart=on-failure adcm
sudo docker start adcm
```

- все данные будут храниться на хост-машине в каталоге /opt/adcm/
- <version>—это версия Docker-образа в одном из следующих форматов:
 - <major>.<minor>.<patch>—если необходим конкретный патч ADCM. Пример: 2.0.0.
 - <major>.<minor>—если необходим последний патч в рамках выбранной версии ADCM. Пример: 2.0.
- Обратите внимание, что до выпуска ADCM 2.0.0 для версионирования использовался следующий формат: YYYY.MM.DD.HH.
- [Настройка ADCM для использования внешней базы данных](#)

Установка сервиса ADCM

- Скачиваем docker-образ ADCM из репозитория (используем последнюю версию ADCM):

```
sudo docker pull hub.arenadata.io/adcm/adcm:<latest or version>
```

Можно использовать конкретную версию ADCM: `hub.arenadata.io/adcm/adcm: 2021.11.22.15`

2021.11.22.15 версия ADCM (<https://docs.arenadata.io/adcm/notes.html>)

- Из образа создаем контейнер:

```
sudo docker create --name adcm -p 8000:8000 -v /opt/adcm:/adcm/data hub.arenadata.io/adcm/adcm:<version>
```

Контейнер создается на порту 8000 с ADCM, и все данные будут храниться на хост-машине в каталоге `/opt/adcm/`

Проверка работоспособности

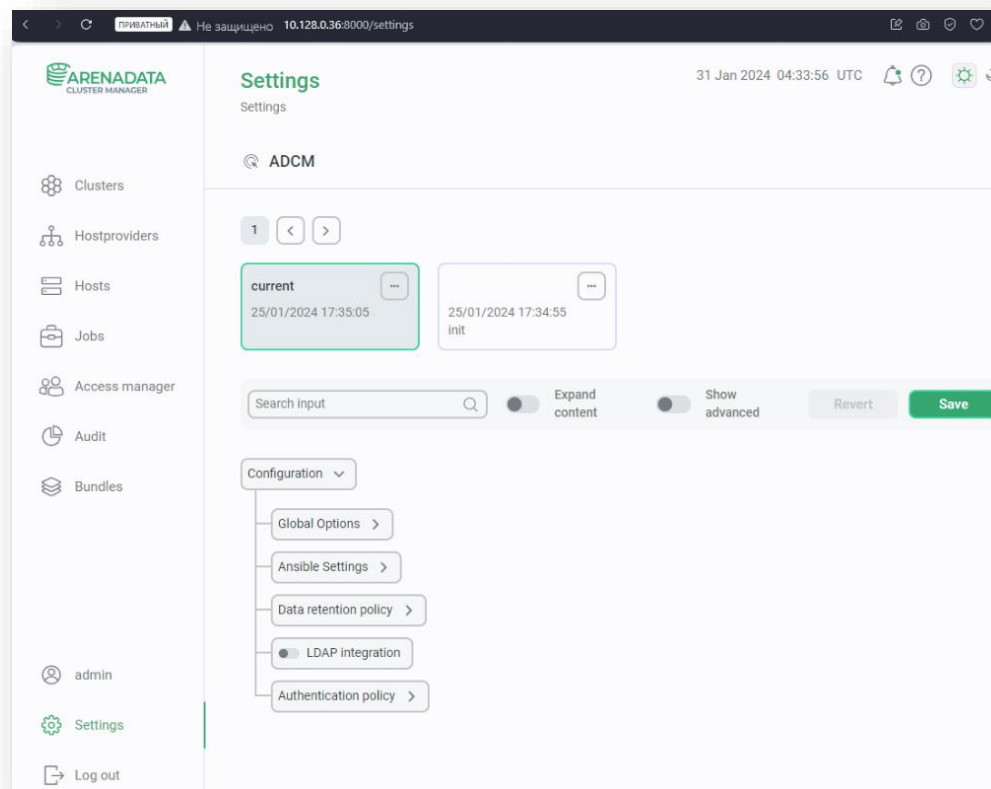
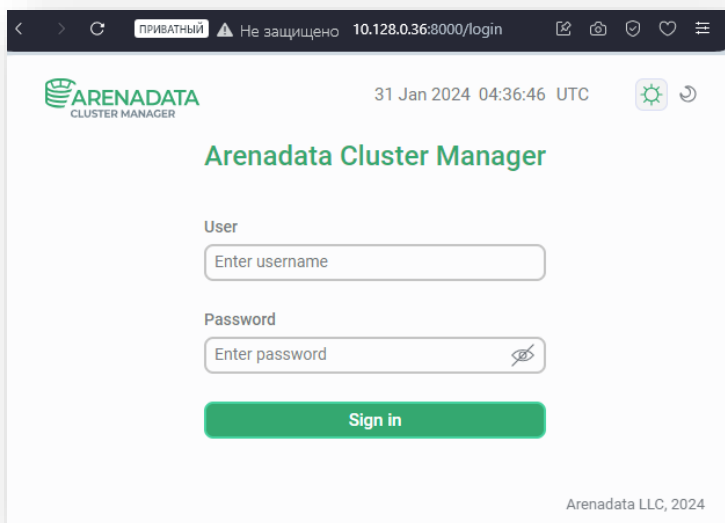
- Проверяем работу ADCM в docker:

```
sudo docker container ls
```

Вывод:

```
5190a5c19a10  hub.arenadata.io/adcm/adcm:<version>  "/etc/startup.sh"  3 hours ago  Up 3 hours  0.0.0.0:8000->8000/tcp  adcm
```

- Проверяем работу ADCM через Web-UI по адресу `http://<ip_address_of_server>:8000`
- (Login и Password – admin):



Изменения версии и удаление ADCM

- Останавливаем docker-контейнер ADCM:

```
sudo docker stop adcm
```

- Для обновления ADCM контейнера, необходимо скачать новый образ из репозитория, удалить старый контейнер и заново развернуть обновленный контейнер (при необходимости удалить содержимое каталога /opt/adcm/):

```
sudo docker container rm adcm
```

```
sudo docker pull hub.arenadata.io/adcm/adcm:<version>
```

```
sudo docker create --name adcm -p 8000:8000 -v /opt/adcm:/adcm/data hub.arenadata.io/adcm/adcm:<version>
```

```
sudo docker start adcm
```

Демонстрация

- Развертывание ADCM **версии 2.X** на узле ads-a-XX-adcm.
- Проверка работоспособности.

Установка и настройка ADS

Настройка инфраструктуры

Настройка инфраструктуры продемонстрирована на примере **SSH Common Bundle**

Общая схема

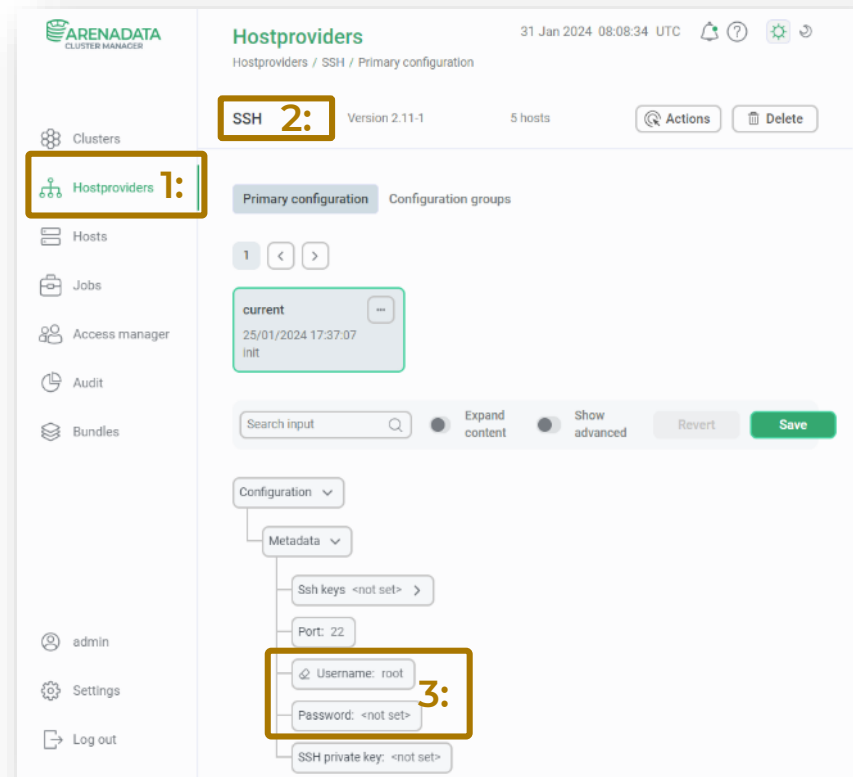
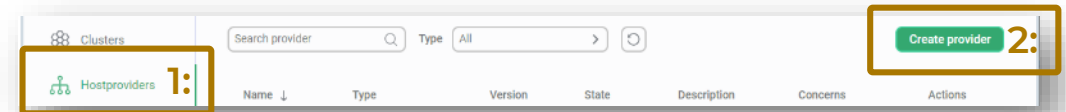
- Загрузить необходимый бандл
- Выбираем и устанавливаем провайдера
- Добавляем узлы
- Настраиваем доступ для хостов
- Проверяем настройки средствами ADCM

Настройка инфраструктуры

- Выбираем Хост провайдер SSH Common Bundle
- Хост провайдер *ssh* – это обычный архив, содержащий описание и программную логику подключения к хосту по протоколу *ssh*
- На основе провайдера создаем хосты соответствующие узлам:
 - `ads-a-XX-node-1.ru-central1.internal`
 - `ads-a-XX-node-2.ru-central1.internal`
 - `ads-a-XX-node-3.ru-central1.internal`
 - `ads-a-XX-adcm.ru-central1.internal` (мониторинг Prometheus)
- Настраиваем способ подключения (логин/пароль или по ключу)
- Если выбран облачный провайдер, то узлы создаются в облаке с указанными параметрами VM (CPU,Memory,...)

Основные этапы

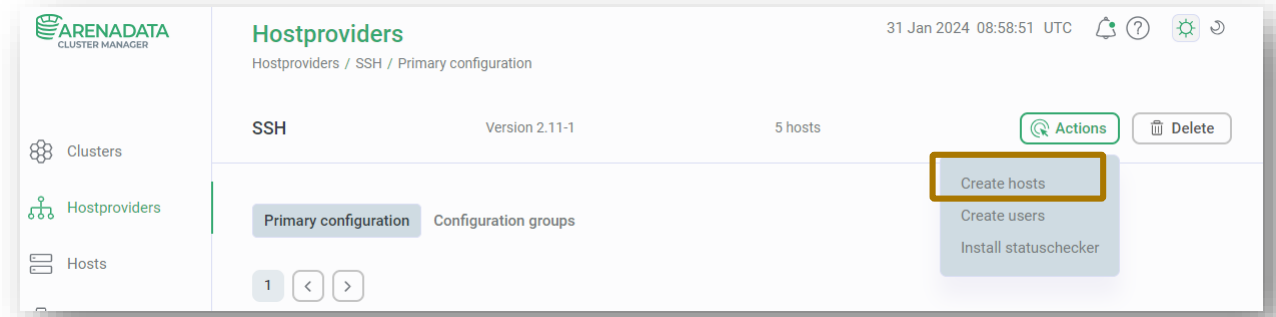
- Скачиваем SSH Common Bundle (https://store.arenadata.io/#products/arenadata_cluster_manager) и загружаем в ADCM
- Создаем хост-провайдер:
Hostproviders → Create provider → SSH Common
- Указываем настройки SSH-провайдера:
 - Username
 - Password



Основные этапы. Создание группы хостов

- Создаем группу хостов:

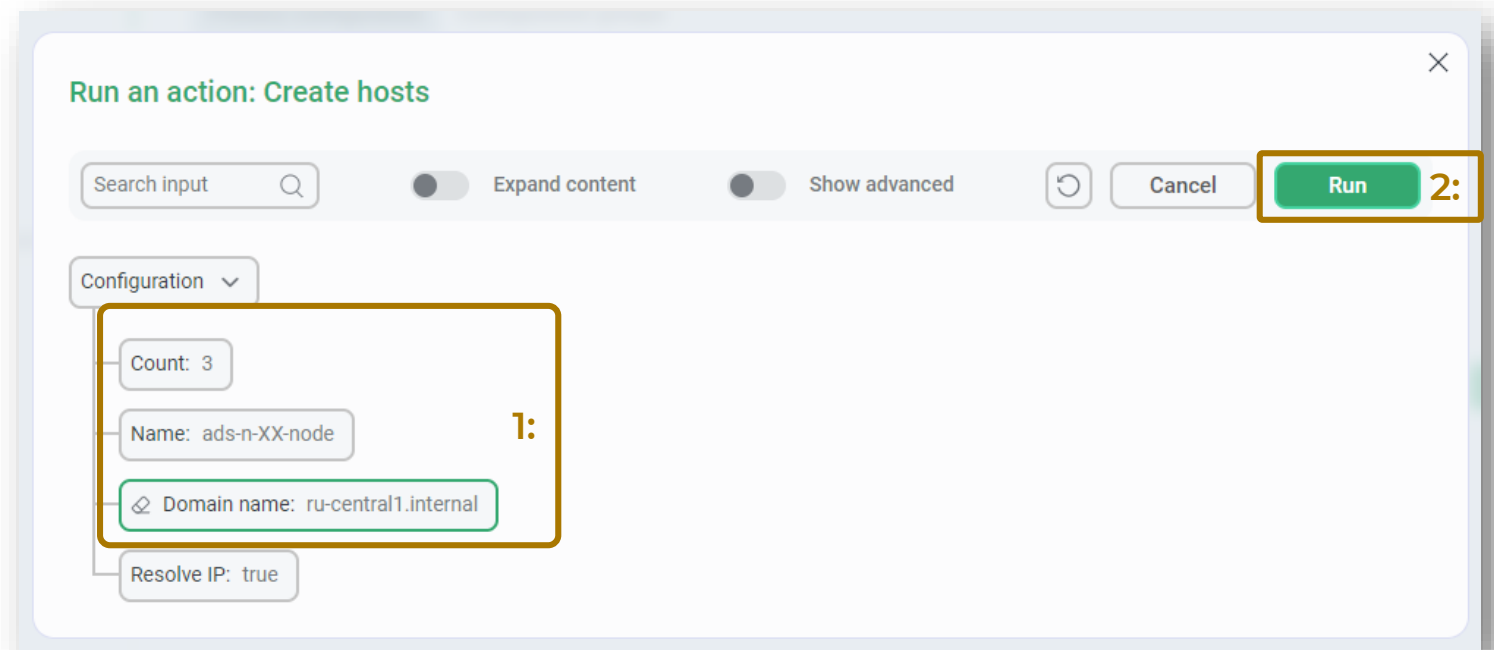
Hostproviders → SSH Common → Action → Create hosts



- Указываем настройки (количество, имя группы, domain):

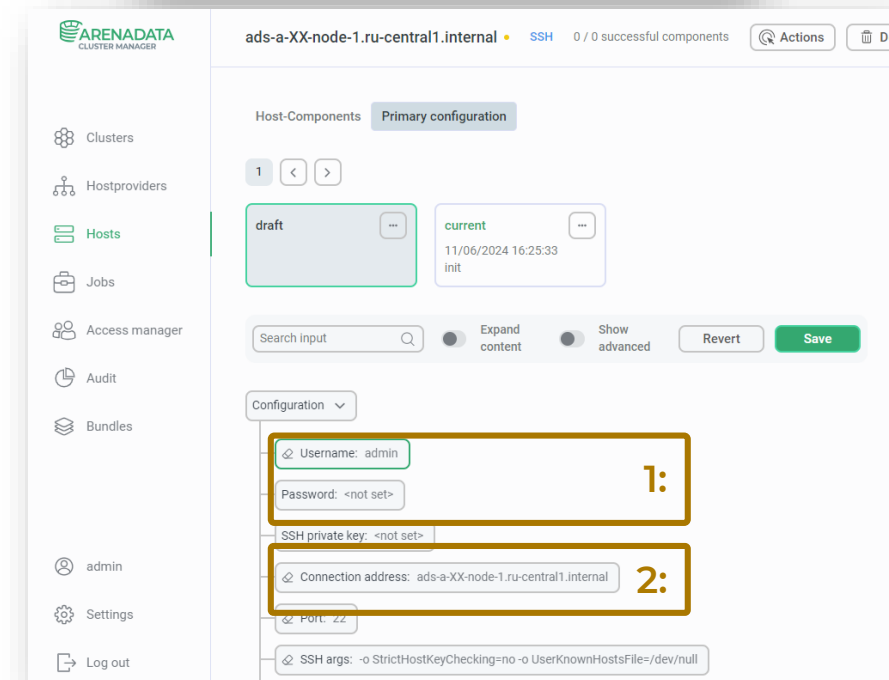
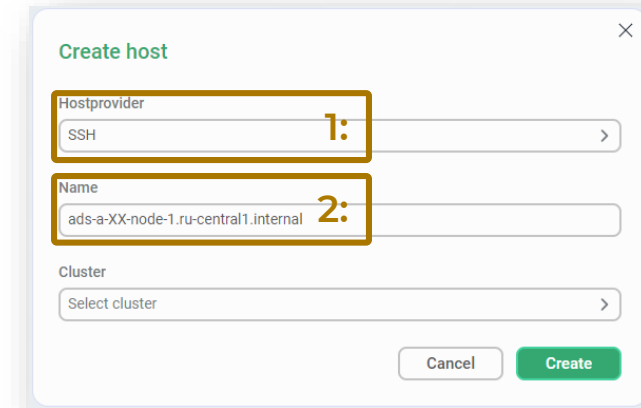
- Узлы будут иметь шаблон:

<name>-<count>.<domain>



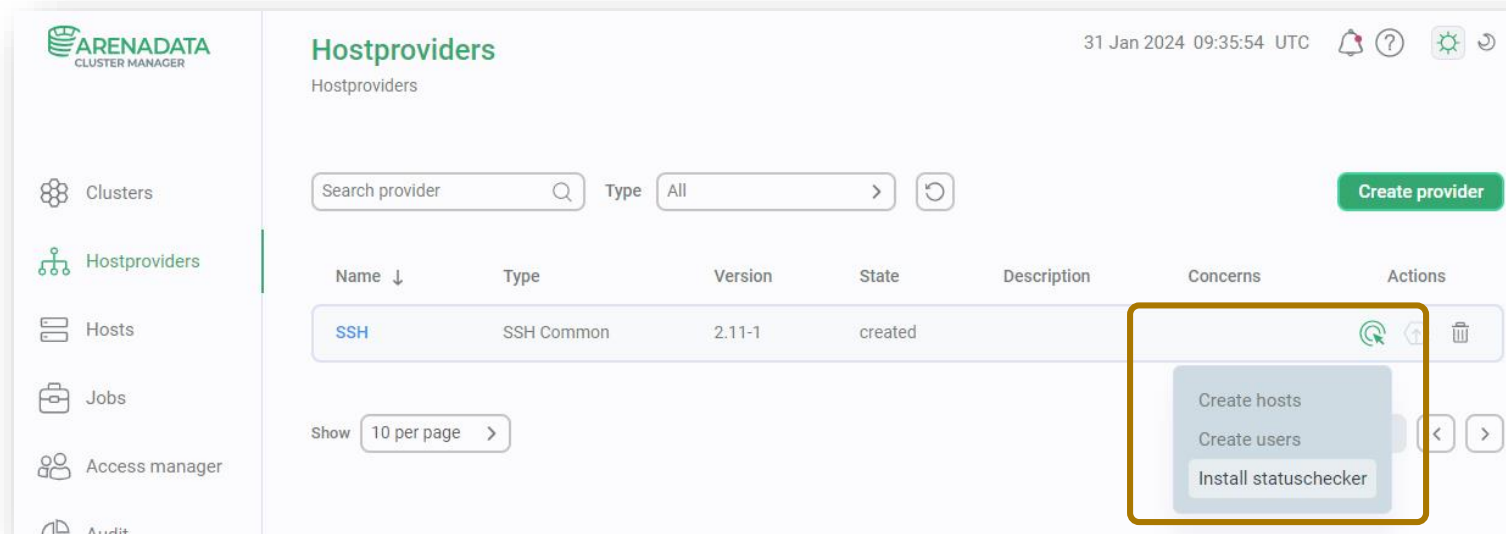
Основные этапы. Создание хоста

- Добавляем хост для узла
Hosts → Create Host
- Заполняем поля (Hostprovider, Name)
- Настраиваем поля для созданного хоста:
Hosts → <хост>
 - *Username/Password*: указываем данные для подключения и/или настраиваем SSH Private Key (опционально)
 - Connection address: указать fully.qualified.domain.name (FQDN) узла
- Прodelываем действия для всех узлов



Основные этапы. Проверка в ADCM

- Для каждого хоста выполнить проверку соединения (**Check connection**) и установить statuschecker (**Install statuschecker**)
- Проверить статусы операций в меню **Jobs**
- Прodelать действия для всех узлов



Лабораторная работа

Учебный кластер (начальная конфигурация) – 4 узла:

- ads-a-XX-adcm.ru-central1.internal
- ads-a-XX-node-1.ru-central1.internal
- ads-a-XX-node-2.ru-central1.internal
- ads-a-XX-node-3.ru-central1.internal

1. Создать хосты для всех узлов, сделав необходимые настройки.
2. Для каждого хоста установить statuschecker (Install statuschecker).

Первичная конфигурация

Основные компоненты кластера:

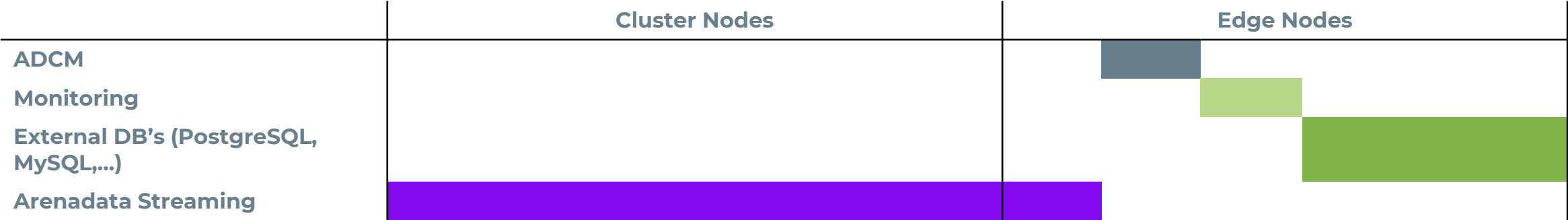
- Apache Zookeeper
- Apache Kafka
- Kafka Connect
- Schema Registry
- Kafka REST Proxy
- ksqlDB
- Kafka-Manager
- Apache Nifi
- Monitoring

Необходимо продумать конфигурацию кластера для указанных компонент:

- Zookeeper (требуется кворум и установки на 3 хостах минимум)
- Kafka Broker (установка на 1-ом и более хостах)
- Kafka Connect (установка на 1-ом и более хостах)
 - Kafka Connect Worker
- Kafka-Manager (установка на 1-ом и более хостах)
- Kafka REST Proxy (установка на 1-ом и более хостах)
- Schema-Registry (установка на 1-ом и более хостах)
- ksqlDB (установка на 1-ом и более хостах)
 - ksqlDB Client
 - ksqlDB Server
- NiFi:
 - Server (установка на 1-ом и более хостах)
 - Registry (установка на 1-ом хосте)
- Monitoring:
 - Prometheus Server (установка на 1-ом хосте)
 - Grafana (установка на 1-ом хосте)
 - Node-exporter (установка на всех хостах)

← Должны располагаться на одном хосте

Узлы кластеров и планирование ролей



Пример распределения компонент для учебного кластера!!!

Кластер	Сервис	Компонент	ads-a- <##> -node-1	ads-a- <##> -node-2	ads-a- <##> -node-3	ads-a- <##> -adcm
ADCM	ADCM	ADCM				1
Arenadata Streaming	Zookeeper	Server	1	1	1	
	Kafka	Kafka Broker	1	1	1	
	Kafka Connect	Kafka Connect Worker	1	1	1	
	Kafka REST Proxy	Kafka REST Proxy			1	
	Schema-Registry	Schema-Registry			1	
	ksqlDB	ksqlDB Client	1	1	1	
		ksqlDB Server		1		
	NiFi	NiFi Server	1	1	1	
		NiFi Registry		1		
	Monitoring	Prometheus Server				1
		Grafana				1
		Node-exporter	1	1	1	
Всего компонент			6	8	8	3

Установка кластера ADS

Общая схема установки кластера ADS:

- ✓ Установить ADCM
- ✓ Подготовить узлы в ADCM
- Создать кластер ADS в ADCM.
- Добавить и настроить сервисы кластера:
 - Kafka
 - Kafka Connect
 - Kafka REST Proxy
 - Monitoring
 - NiFi
 - Schema-Registry
 - Zookeeper
 - ksqlDB
- Добавить в кластер хосты.
- Распределить компоненты на хосты кластера.
- Настроить импорт JMX-метрик.

Основные этапы. Кластер и сервисы

- ✓ Установить ADCM
- ✓ Подготовить узлы в ADCM
- ✓ Создать кластер ADS в ADCM.
- Добавить и настроить сервисы кластера:
 - Kafka
 - Kafka Connect
 - Kafka REST Proxy
 - Monitoring
 - NiFi
 - Schema-Registry
 - Zookeeper
 - ksqlDB
- Добавить в кластер хосты.
- Распределить компоненты на хосты кластера.
- Настроить импорт JMX-метрик.

- Загружаем бандл ADS v2.8+ enterprise (в случае установки Enterprise-версии пакет предоставляется компанией Arenadata) и принимаем условия лицензии.

- Создаем кластер:

Clusters → Create Cluster

Заполняем поля и нажимаем **Create**:

Product: **ADS**

Product version: **<#.#>**

Cluster name: **ADS**

- Добавить и настроить сервисы кластера:

- Kafka
- Kafka Connect
- Kafka REST Proxy
- Kafka-Manager
- Monitoring
- NiFi
- Schema-Registry
- Zookeeper
- ksqlDB

Clusters → ADS → Services → Add service

(выбираем только указанные)

Принять лицензионные соглашения (Kafka REST Proxy, Schema-Registry, ksqlDB)

Create cluster

Product: ADS

Product version: 2.8.1_arenadata1_b1-1 (enterprise)

Cluster name: ADS

Description: Description

☐ I accept Terms of Agreement

Cancel Create

Add services

☐ All services

Search services Select filtered

- ☐ [Deprecated] Monitoring clients
- ☒ Kafka
- ☒ Kafka Connect
- ☒ Kafka REST Proxy
- ☒ ksqlDB
- ☐ MINIFI
- ☒ Monitoring
- ☒ NiFi
- ☒ Schema-Registry
- ☒ Zookeeper

☐ Services you selected require you to accept Terms of Agreement

Cancel Next

Add services

Kafka REST Proxy ksqlDB Schema-Registry

☒ I have read text of License Agreement

Accept

This software is made available by Confluent, Inc., under the terms of the Confluent Community License Agreement, Version 1.0 located at <http://www.confluent.io/confluent-community-license>. BY INSTALLING, DOWNLOADING, ACCESSING, USING OR DISTRIBUTING ANY OF THE SOFTWARE, YOU AGREE TO THE TERMS OF SUCH LICENSE AGREEMENT.

Cancel Add

Add services

Kafka REST Proxy ksqlDB Schema-Registry

☒ I have read text of License Agreement

Accept

This software is made available by Confluent, Inc., under the terms of the Confluent Community License Agreement, Version 1.0 located at <http://www.confluent.io/confluent-community-license>. BY INSTALLING, DOWNLOADING, ACCESSING, USING OR DISTRIBUTING ANY OF THE SOFTWARE, YOU AGREE TO THE TERMS OF SUCH LICENSE AGREEMENT.

Cancel Add

Add services

Kafka REST Proxy ksqlDB Schema-Registry

☒ I have read text of License Agreement

Accept

This software is made available by Confluent, Inc., under the terms of the Confluent Community License Agreement, Version 1.0 located at <http://www.confluent.io/confluent-community-license>. BY INSTALLING, DOWNLOADING, ACCESSING, USING OR DISTRIBUTING ANY OF THE SOFTWARE, YOU AGREE TO THE TERMS OF SUCH LICENSE AGREEMENT.

Cancel Add

Основные этапы. Хосты и компоненты

- ✓ Установить ADCM
- ✓ Подготовить узлы в ADCM
- ✓ Создать кластер ADS в ADCM.
- ✓ Добавить и настроить сервисы кластера:
 - ✓ Kafka
 - ✓ Kafka Connect
 - ✓ Kafka REST Proxy
 - ✓ Monitoring
 - ✓ NiFi
 - ✓ Schema-Registry
 - ✓ Zookeeper
 - ✓ ksqldb
- Добавить в кластер хосты.
- Распределить компоненты на хосты кластера.
- Настроить импорт JMX-метрик..

- Добавить в кластер хосты:
- **Clusters → ADS → Hosts → Add hosts**
 - ads-a-XX-node-1.ru-central1.internal
 - ads-a-XX-node-2.ru-central1.internal
 - ads-a-XX-node-3.ru-central1.internal
 - ads-a-XX-adcm.ru-central1.internal
- Распределить компоненты на хосты кластера согласно этапу планирования (см. таблицу):
Clusters → ADS → Mapping
- Проверяем правильность распределения по количеству и составу компонент для каждого хоста (после сохранения нельзя изменить)
- **ОБЯЗАТЕЛЬНО ПРОВЕРЬТЕ ПРАВИЛЬНОСТЬ РАСПРЕДЕЛЕНИЯ КОМПОНЕНТ И СРАВНИТЕ ИХ КОЛИЧЕСТВО В СООТВЕТСТВИИ С ПОСЛЕДНЕЙ СТРОКОЙ ТАБЛИЦЫ!!!**
- После настройки всех ролей сохранить распределение (Save)

Кластер	Сервис	Компонент	ads-a-<##>-node-1	ads-a-<##>-node-2	ads-a-<##>-node-3	ads-a-<##>-adcm
ADCM	ADCM	ADCM				1
Arenadata Streaming	Zookeeper	Server	1	1	1	
	Kafka	Kafka Broker	1	1	1	
	Kafka Connect	Kafka Connect Worker	1	1	1	
	Kafka REST Proxy	Kafka REST Proxy			1	
	Schema-Registry	Schema-Registry			1	
	ksqldb	ksqldb Client	1	1	1	
		ksqldb Server		1		
	NiFi	NiFi Server	1	1	1	
		NiFi Registry		1		
	Monitoring	Prometheus Server				1
		Grafana				1
		Node-exporter	1	1	1	
Всего компонент			6	8	8	3

Основные этапы. Хосты и компоненты

- ✓ Установить ADCM
- ✓ Подготовить узлы в ADCM
- ✓ Создать кластер ADS в ADCM.
- ✓ Добавить и настроить сервисы кластера:

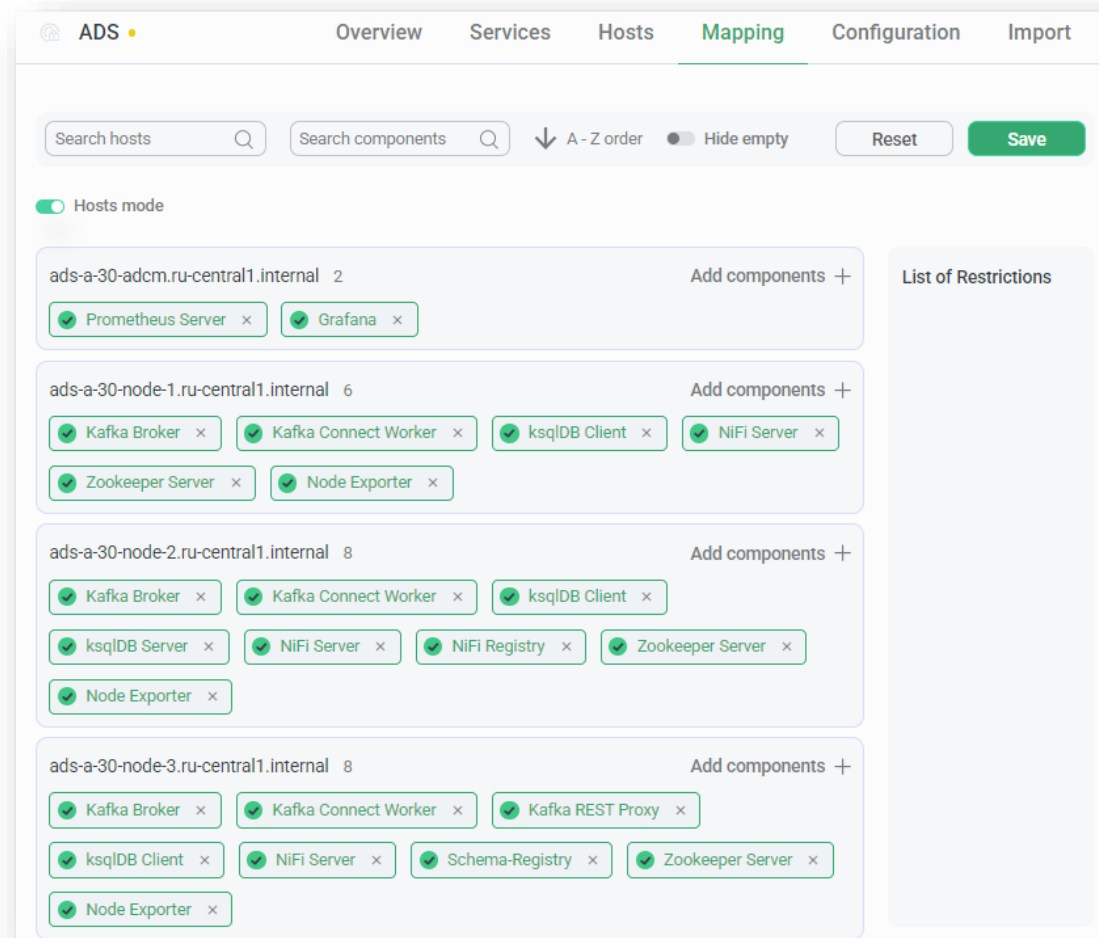
- ✓ Kafka
- ✓ Kafka Connect
- ✓ Kafka REST Proxy
- ✓ Monitoring
- ✓ NiFi
- ✓ Schema-Registry
- ✓ Zookeeper
- ✓ ksqiDB

- Добавить в кластер хосты.
- Распределить компоненты на хосты кластера.
- Настроить импорт JMX-метрик.

- Добавить в кластер хосты:
- **Clusters → ADS → Hosts → Add hosts**
 - ads-a-XX-node-1.ru-central1.internal
 - ads-a-XX-node-2.ru-central1.internal
 - ads-a-XX-node-3.ru-central1.internal
 - ads-a-XX-adcm.ru-central1.internal
- Распределить компоненты на хосты кластера согласно этапу планирования (см. таблицу):

Clusters → ADS → Mapping

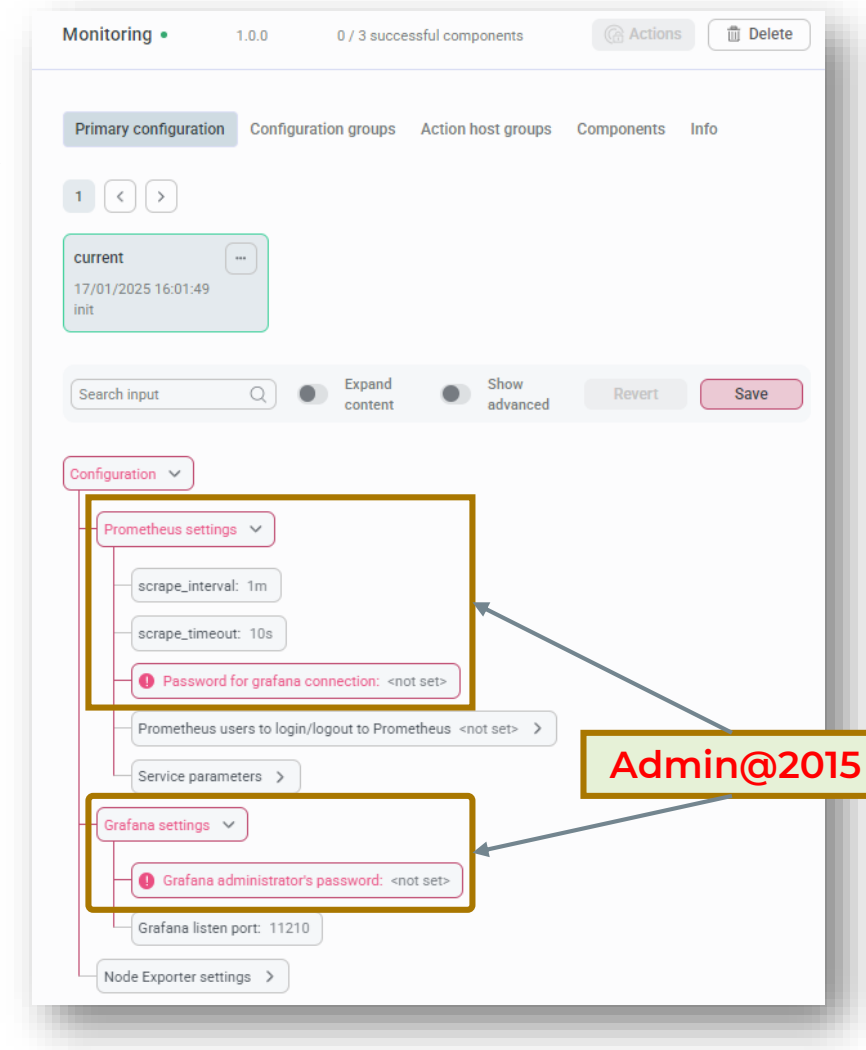
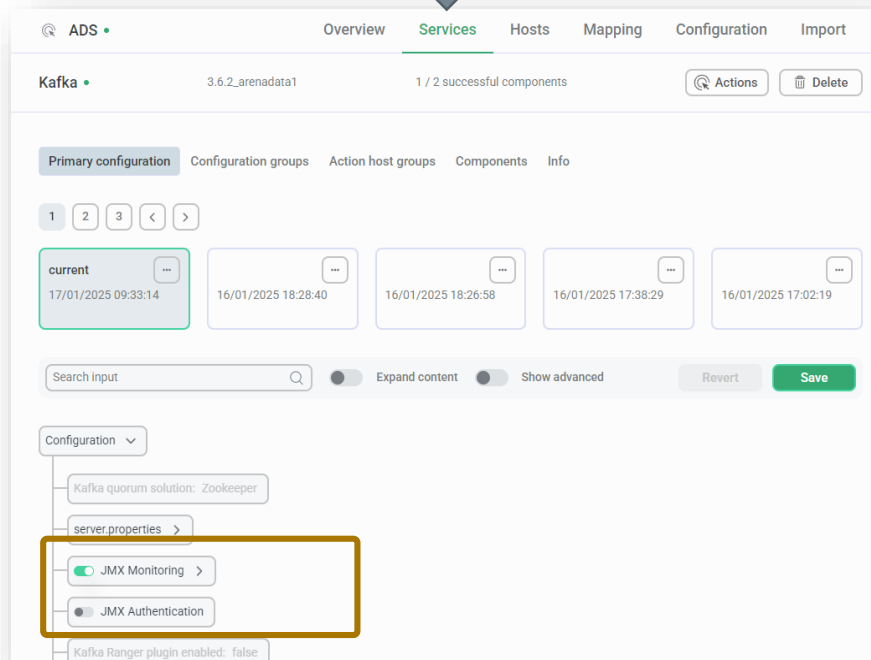
- Проверяем правильность распределения по количеству и составу компонент для каждого хоста (после сохранения нельзя изменить)
- После настройки всех ролей сохранить распределение (Save)



Основные этапы. Мониторинг

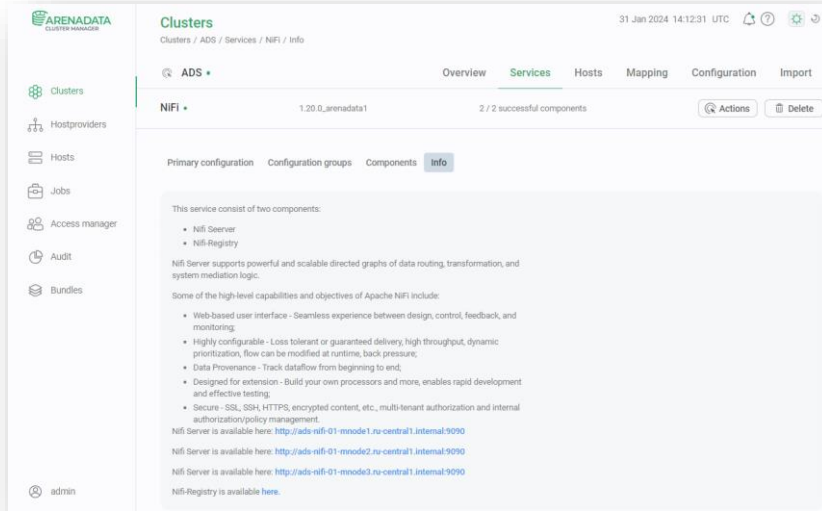
- ✓ Установить ADCM
- ✓ Подготовить узлы в ADCM
- ✓ Создать кластер ADS в ADCM.
- ✓ Добавить и настроить сервисы кластера:
 - ✓ Kafka
 - ✓ Kafka Connect
 - ✓ Kafka REST Proxy
 - ✓ Monitoring
 - ✓ NiFi
 - ✓ Schema-Registry
 - ✓ Zookeeper
 - ✓ ksqldb
- ✓ Добавить в кластер хосты.
- ✓ Распределить компоненты на хосты кластера.
- Настроить импорт JMX-метрик.

- Заполнить Password for grafana connection (пароль пользователя Grafana (admprom_grafana) для подключения к Prometheus)
- Заполнить Grafana administrator's password (пароль администратора Grafana)
- Для сервисов ADS настроить импорт JMX-метрик (например, Kafka)

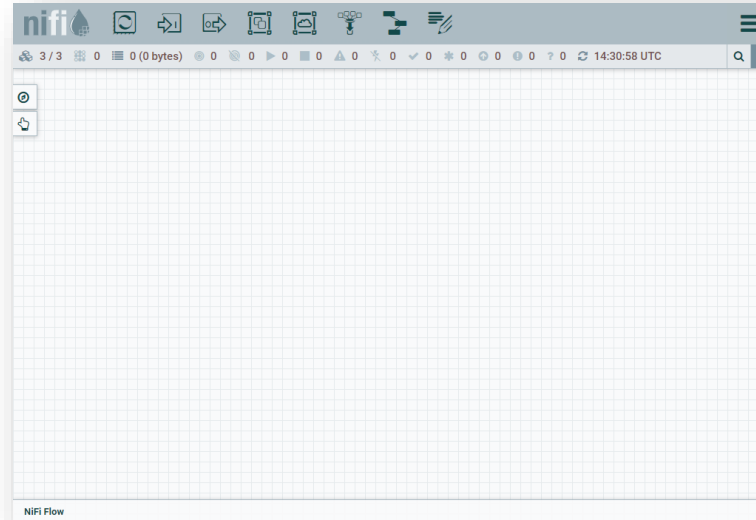


ADS

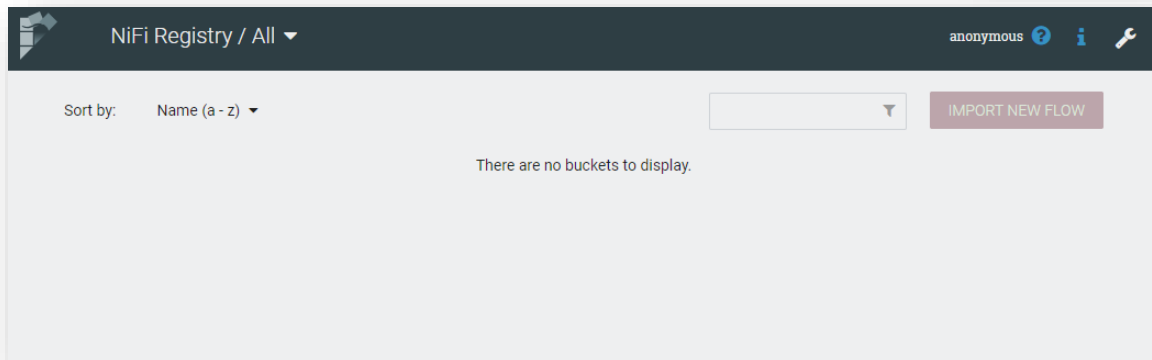
- ADCM: ADS Cluster:



- NiFi Server:
 - <http://<node-1>:9090/nifi>
 - <http://<node-2>:9090/nifi>
 - <http://<node-3>:9090/nifi>



- NiFi Registry:
 - <http://<node>:18080/nifi-registry/>



Конфигурационные файлы сервисов ADS

- Zookeeper (/etc/zookeeper/conf/)

configuration.xml log4j.properties zoo.cfg zookeeper-env.sh zoo_sample.cfg

- NiFi Server (/etc/nifi/conf/)

nifi.properties authorizers.xml bootstrap.conf logback.xml ...

- NiFi Registry

nifi-registry.properties authorizers.xml bootstrap.conf logback.xml ...

- MiNiFi (/usr/lib/minifi-c2/conf/)

c2.properties authorizers.xml bootstrap.conf logback.xml ...

- Kafka, Kafka Connect

/etc/<Service_Name>/conf/*.*

- Kafka REST Proxy, ksqlDB, Schema-Registry

/etc/<Service_Name>/*.*

Лабораторная работа

- Установить кластер Arenadata Streaming на хосты:

ads-a-XX-node-1.ru-central1.internal

ads-a-XX-node-2.ru-central1.internal

ads-a-XX-node-3.ru-central1.internal

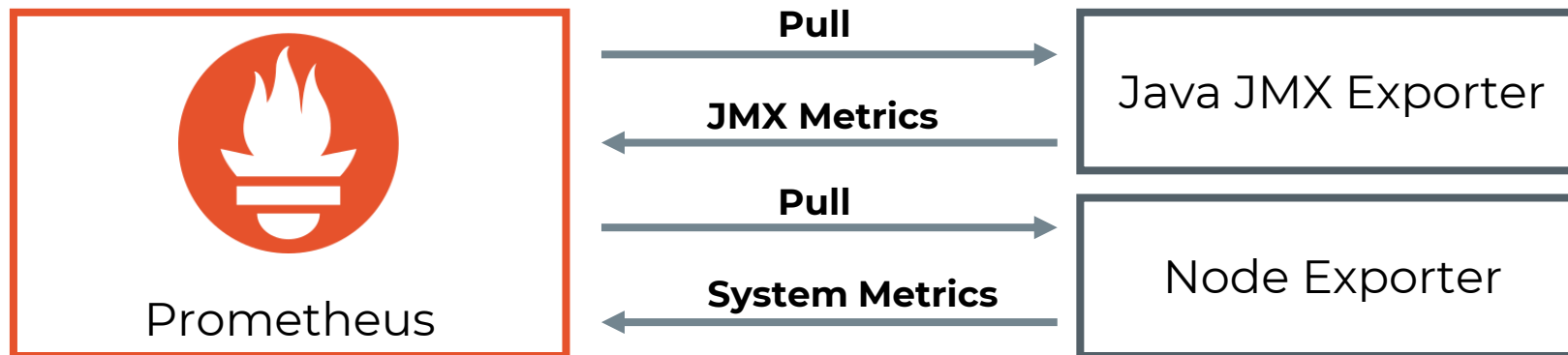
ads-a-XX-adcm.ru-central1.internal

Мониторинг ADS

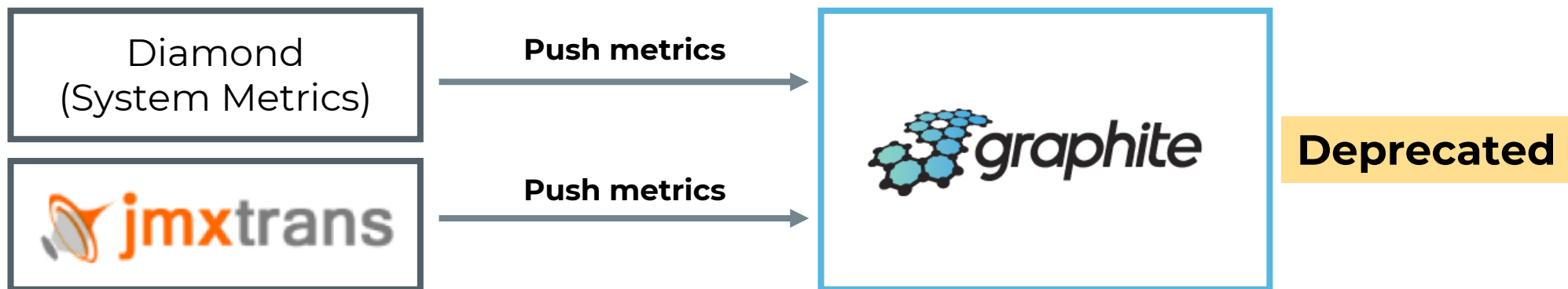
Служба Мониторинга (Prometheus + Grafana)

Мониторинг ADS

Pull-based мониторинг



Push-based мониторинг



Мониторинг ADS



Мониторинг

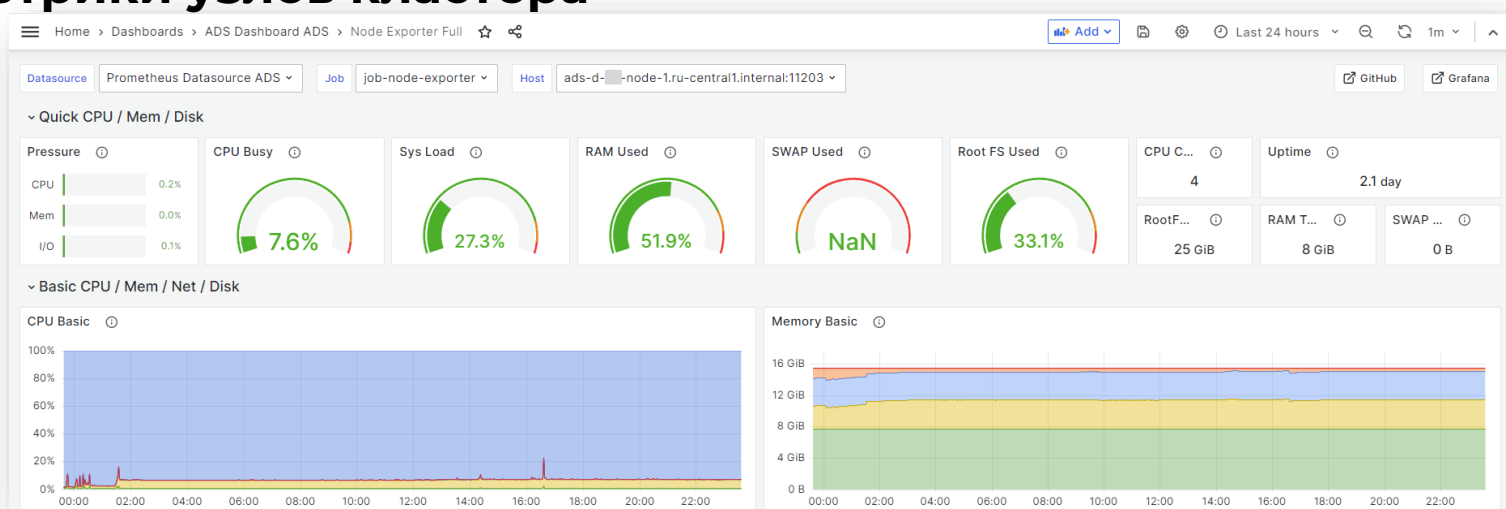
Системные метрики узлов кластера

Метрики сервисов ADS

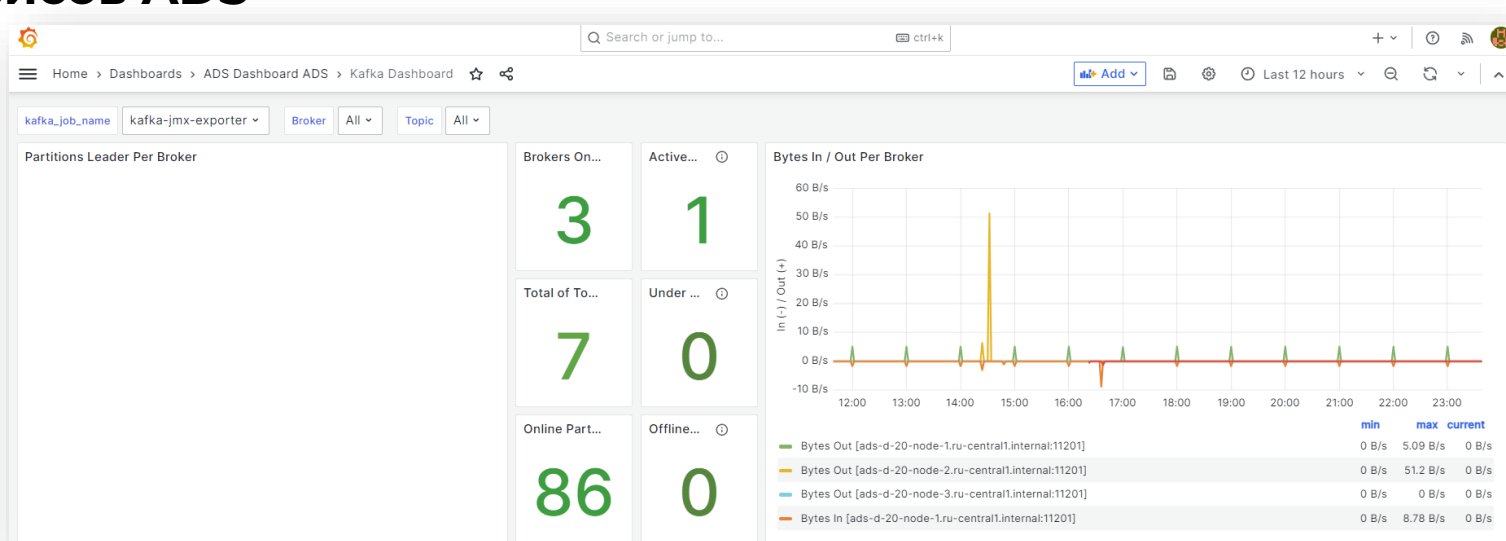
- Сервис Monitoring разворачивает в ADS свой сервер Prometheus, который реализует сбор и хранение метрик мониторинга кластера ADS,
- Сервис Grafana для визуализации и анализа информации.
- В кластер ADS на все хосты устанавливается node exporter — агент мониторинга, считывающий с хостов системные метрики, которые будет собирать Prometheus.
- JMX метрики собираются через JMX Exporter (сборщик значений JMX Mbean).
- Каждый агент JMX Exporter создает HTTP-сервер и предоставляет метрики JMX для Prometheus.
- Агент Java JMX Exporter работает как агент Java в вашем приложении и собирает значения JMX Mbean
Например, для Kafka необходимо добавьте к каждому брокеру Apache Kafka эту переменную среды
`KAFKA_OPTS: "-javaagent:/usr/lib/ads/jmx/jmx_prometheus_javaagent.jar=11201:/usr/lib/ads/jmx/conf/kafka.yml"`

Мониторинг ADS

Системные метрики узлов кластера



Метрики сервисов ADS



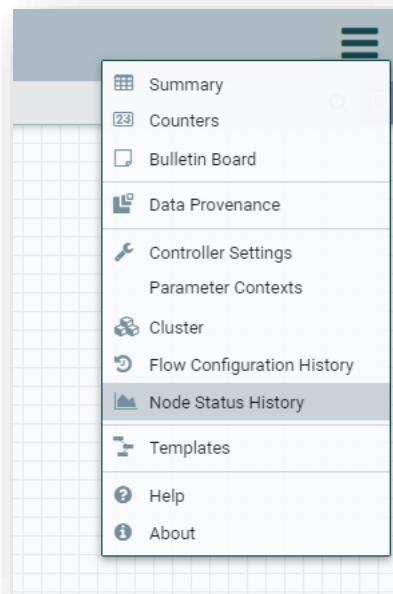
Мониторинг NiFi

- Встроенный мониторинг на основе JMX

- NiFi JMX Metrics Dump

REST API: <http://<node>:9090/nifi-api/system-diagnostics/>

```
1 {
2   "systemDiagnostics": {
3     "aggregateSnapshot": {
4       "totalNonHeap": "226.89 MB",
5       "totalNonHeapBytes": 237916160,
6       "usedNonHeap": "214.26 MB",
7       "usedNonHeapBytes": 224670328,
8       "freeNonHeap": "12.63 MB",
9       "freeNonHeapBytes": 13245832,
10      "maxNonHeap": "-1 bytes",
11      "maxNonHeapBytes": -1,
12      "totalHeap": "1,024 MB",
13      "totalHeapBytes": 1073741824,
14      "usedHeap": "542.79 MB",
15      "usedHeapBytes": 569155424,
16      "freeHeap": "481.21 MB",
17      "freeHeapBytes": 504586400,
18      "maxHeap": "1,024 MB"
```



NiFi JMX Metrics Dump

REST API:

- <http://<node>:9090/nifi-api/system-diagnostics/>

```
1 {
2   "systemDiagnostics": {
3     "aggregateSnapshot": {
4       "totalNonHeap": "226.89 MB",
5       "totalNonHeapBytes": 237916160,
6       "usedNonHeap": "214.26 MB",
7       "usedNonHeapBytes": 224670328,
8       "freeNonHeap": "12.63 MB",
9       "freeNonHeapBytes": 13245832,
10      "maxNonHeap": "-1 bytes",
11      "maxNonHeapBytes": -1,
12      "totalHeap": "1,024 MB",
13      "totalHeapBytes": 1073741824,
14      "usedHeap": "542.79 MB",
15      "usedHeapBytes": 569155424,
16      "freeHeap": "481.21 MB",
17      "freeHeapBytes": 504586400,
18      "maxHeap": "1,024 MB",
19      "maxHeapBytes": 1073741824,
20      "heapUtilization": "53.0%",
21      "availableProcessors": 4,
22      "processorLoadAverage": 0.07,
23      "totalThreads": 91,
24      "daemonThreads": 42,
25      "uptime": "03:58:50.342",
26      "flowFileRepositoryStorageUsage": {
27        "freeSpace": "19.12 GB",
28        "totalSpace": "24.99 GB",
29        "usedSpace": "5.87 GB",
30        "freeSpaceBytes": 20532097024,
31        "totalSpaceBytes": 26830942208,
32        "usedSpaceBytes": 6298845184,
33        "utilization": "23.0%"
34      },
35      "contentRepositoryStorageUsage": [
36        {
37          "identifier": "repo0",
38          "freeSpace": "19.12 GB",
39          "totalSpace": "24.99 GB",
40          "usedSpace": "5.87 GB",
41          "freeSpaceBytes": 20532097024,
42          "totalSpaceBytes": 26830942208,
43          "usedSpaceBytes": 6298845184,
44          "utilization": "23.0%"
45        },
46        {
47          "identifier": "repo1",
48          "freeSpace": "19.12 GB",
```

JSON XML HTML PREVIEW PLAIN

Мониторинг ADS

Кластер Мониторинга (Deprecated)

Кластер Мониторинга (Deprecated)

- Jmxtrans – это сборщик данных, который собирает метрики через JMX
- Diamond – собирает и отправляет системные метрики с узлов
- Graphite – фронтенд, читающий метрики, в основном через API
- Grafana – визуализатор метрик Graphite и возможность создавать алерты

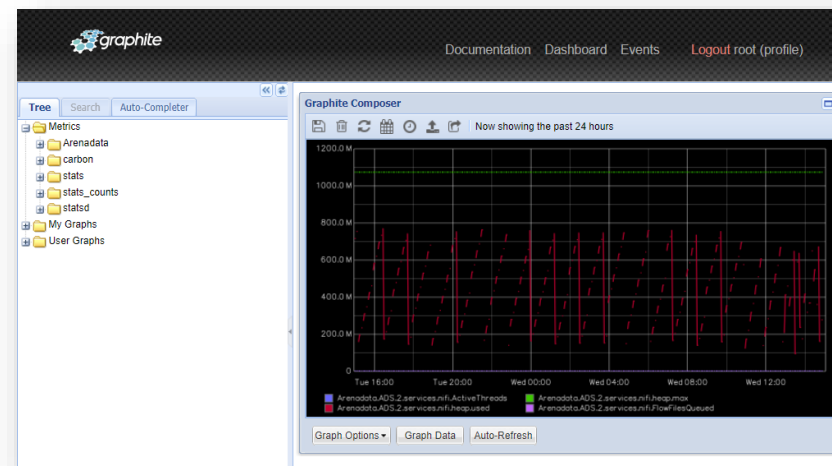
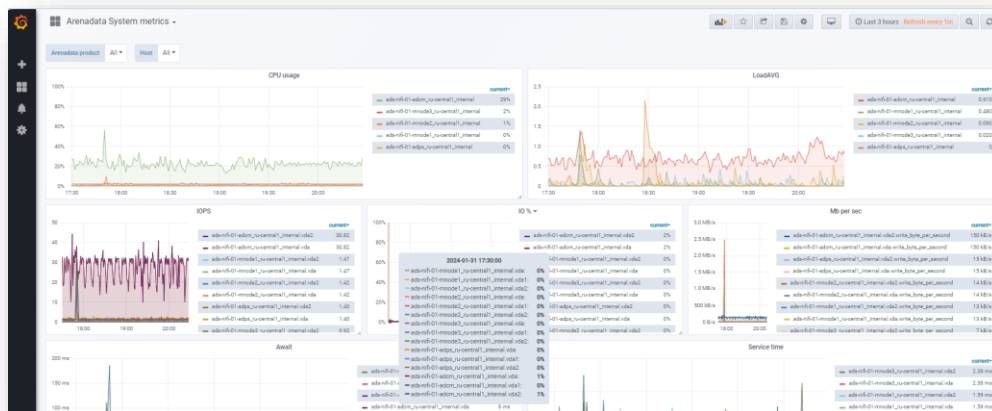
Доступ Graphite: <http://<monitoring host>>

(login: root, password: root)

Доступ Grafana: <http://<monitoring host>:3000>

(login: admin, password: admin)

Начиная с версии 3.6.2.2.b1 не рекомендуется настраивать мониторинг ADS через отдельный кластер мониторинга, так как с этого момента поддержка данной функциональности приостановлена и будет полностью прекращена в будущих релизах ADS. Рекомендуемый способ мониторинга кластера ADS — использовать [сервис Monitoring](#).














Кластер мониторинга Arenadata (Deprecated)

Системные метрики узлов


- CPU
- Memory
- ...

Метрики кластера

 AD.B	Big Data Warehouse	 Greenplum
 AD.H	Cold / Unstructured	 hadoop
 AD.QM	Fast flat marts	 ClickHouse
 AD.PG	OLTP DataBase	 PostgreSQL
 AD.S	ESB / MB/ MQ	 kafka  nifi

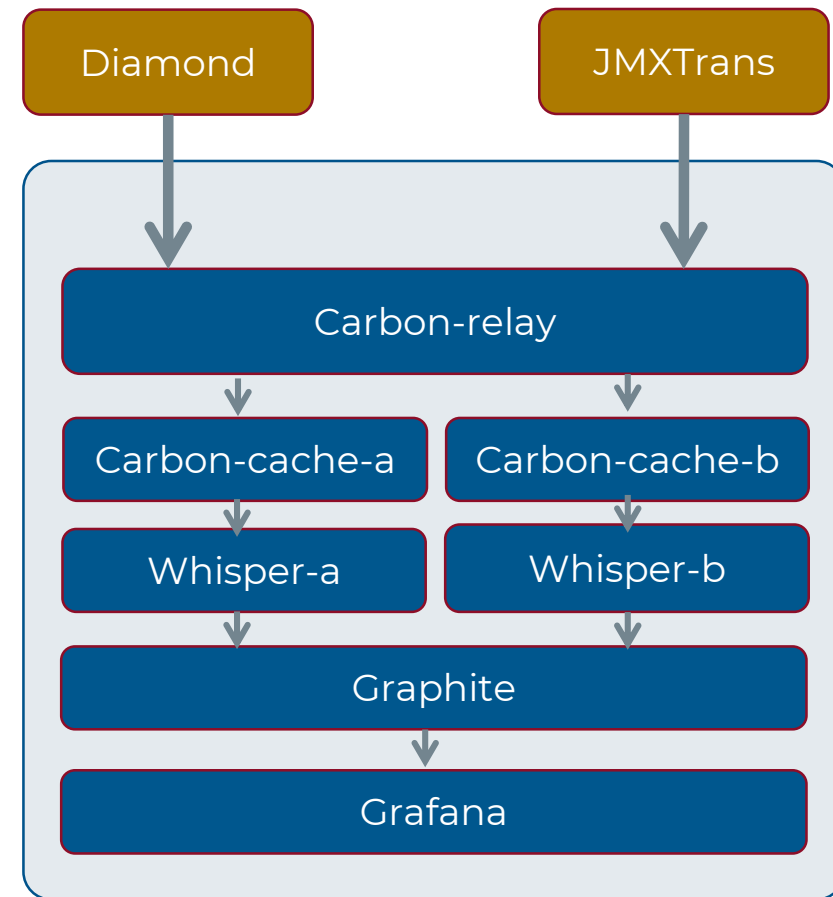
Примеры кластерных метрик:

 AD.S JMX (Java Management Extensions)

 AD.B Запросы к кластеру
(psql -d postgres -c 'select .. ')

Мониторинг. Архитектура (Deprecated)

- Сервис мониторинга модульный, состоит из нескольких компонент:
 - Diamond + JMXTrans – собирает и отправляет системные метрики с узлов и сервисов
 - Carbon-relay – принимает метрики, преобразует в бинарный формат и отправляет Carbon-caches
 - Carbon-caches – принимают метрики и записывают их в Whisper DB
 - Whisper DB – file-based time-series БД
 - Graphite – фронтенд, читающий метрики из Whisper DB и отдающий их в ответ на запросы
 - Grafana – визуализатор метрик Graphite и алерты
- Модульность системы позволяет выполнять масштабирование при росте числа метрик



Diamond

- Установлен на всех узлах кластера ADS.
- `/var/log/diamond/archive.log` – архив всех отосланных метрик.
- `/var/log/diamond/diamond.log` – ошибки и другие сообщения.
- `/usr/share/diamond/collectors/` – скрипты для коллекторов.
- `/etc/diamond/diamond.conf` – настройка адреса отсылки, периодичности и т.д.

...

```
path_prefix = Arenadata.ADS.2.System_metrics
```

...

- Diamond имеет модульную архитектуру, расширяется благодаря коллекторам:

```
/etc/diamond/collectors/
```

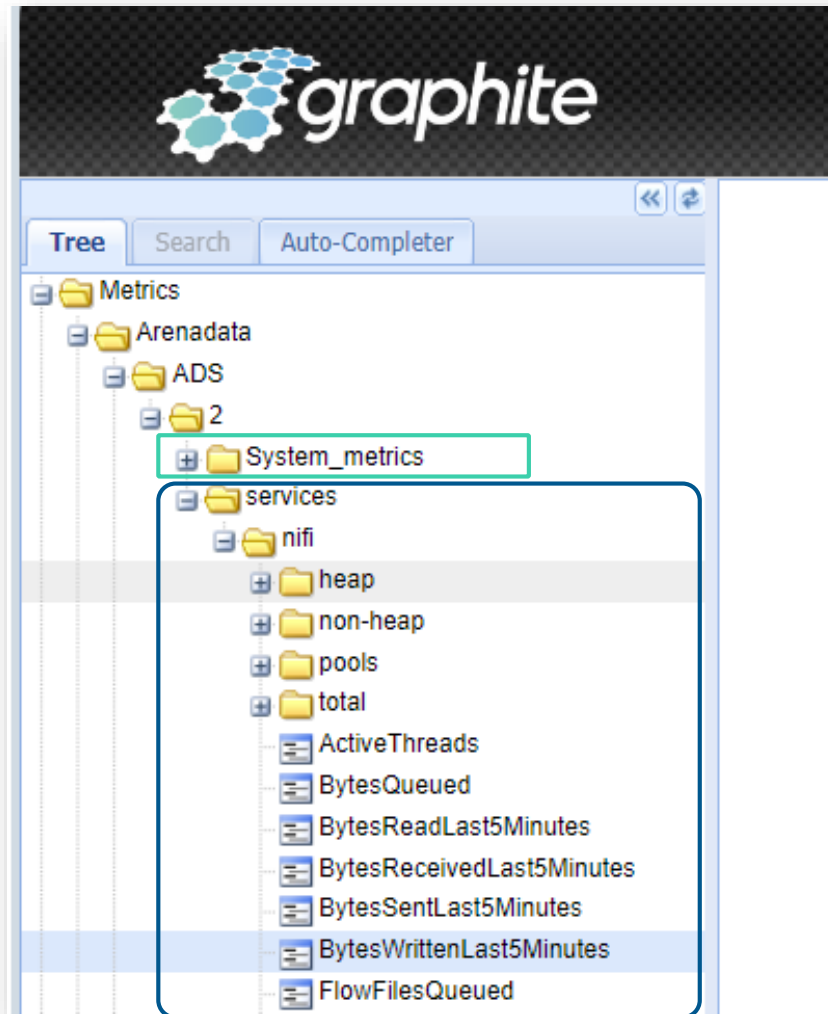
```
├── CPUCollector.conf
├── DiskSpaceCollector.conf
├── DiskUsageCollector.conf
├── FilestatCollector.conf
├── LoadAverageCollector.conf
├── MemoryCollector.conf
├── NetstatCollector.conf
├── NetworkCollector.conf
├── SmartCollector.conf
└── UptimeCollector.conf
```

Мониторинг ADS (Deprecated)

Виды мониторинга:

- Системные метрики узлов кластера
- Метрики сервисов ADS ([список метрик](#))

Graphite

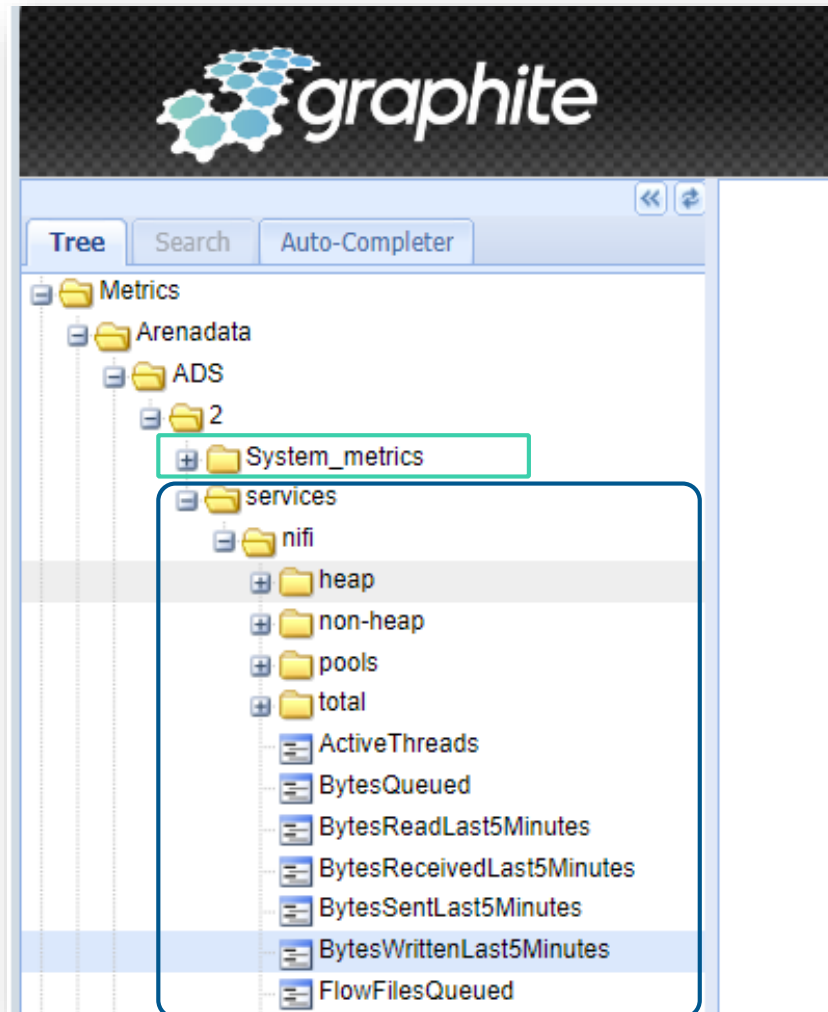


Мониторинг ADS (Deprecated)

Виды мониторинга:

- Системные метрики узлов кластера
- Метрики сервисов ADS ([список метрик](#))

Graphite

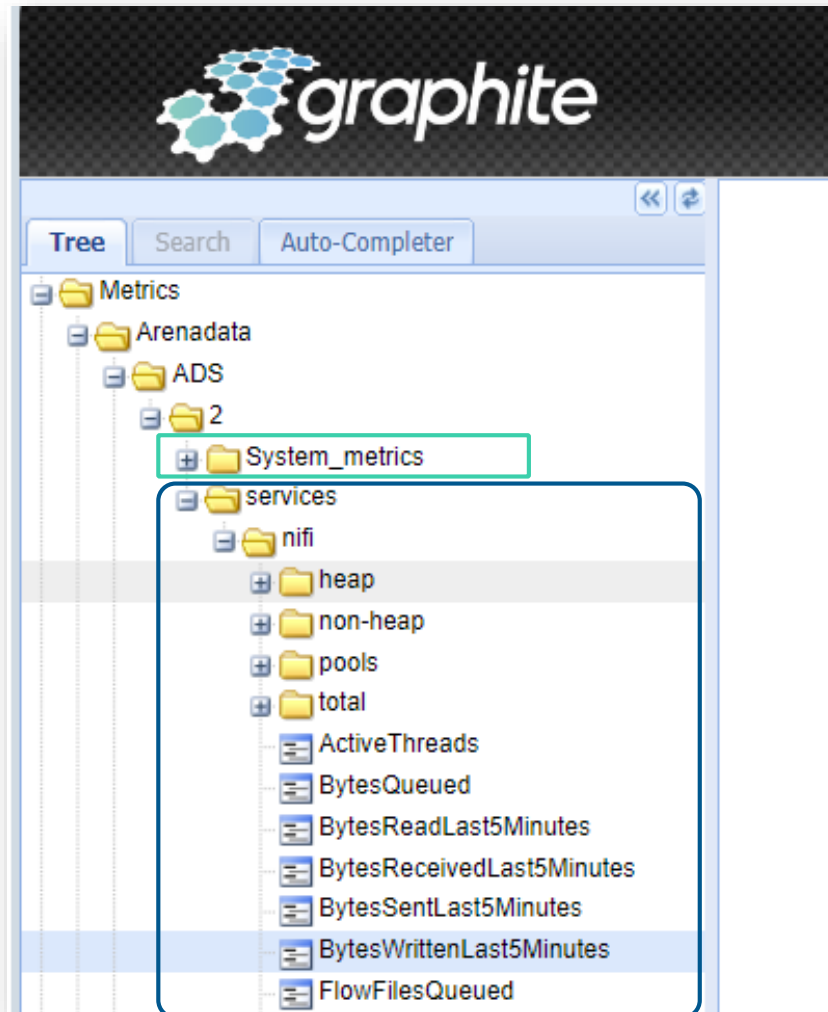


Мониторинг ADS (Deprecated)

Виды мониторинга:

- Системные метрики узлов кластера
- Метрики сервисов ADS ([список метрик](#))

Graphite



Мониторинг ADS NiFi

MetricsReportingTask

- Эта задача создания отчетов передает набор показателей, касающихся JVM и экземпляра NiFi, для отчета, предоставляемого сервисом MetricReporterService.

GraphiteMetricReporterService

- Для работы MetricsReportingTask (задачи отчётности) необходимо настроить службу контроллера GraphiteMetricReporterService (Implementation MetricReporterService), которая передает метрические отчеты для Graphite на определенный порт и с префиксом метрики (Arenadata.ADS.2.services.nifi).

The screenshot displays the NiFi Settings interface. At the top, there are tabs for GENERAL, MANAGEMENT CONTROLLER SERVICES, REPORTING TASKS, REGISTRY CLIENTS, and PARAMETER PROVIDERS. The REPORTING TASKS tab is selected. Below the tabs, a table lists reporting tasks. The first task is 'ADCM Metric Reporter Service' of type 'GraphiteMetricReporterService'. A modal window titled 'Controller Service Details' is open, showing details for 'GraphiteMetricReporterService 1.20.0'. The modal has tabs for SETTINGS, PROPERTIES, and COMMENTS. The SETTINGS tab is active, showing fields for Name, ID, Type, Bundle, Supports Controller Service, and Bulletin Level. The Name is 'ADCM Metric Reporter Service', the ID is '41c3376f-018d-1000-ffff-ffff978421d7', the Type is 'GraphiteMetricReporterService 1.20.0', the Bundle is 'org.apache.nifi - nifi-metrics-reporting-nar', and the Bulletin Level is 'WARN'. The Supports Controller Service section shows a list of supported services, including 'MetricReporterService 1.20.0 from org.apache.nifi - nifi-metrics-reporter-service-api-nar'. A 'Referencing Components' section on the right shows 'Reporting Tasks (1)' with a link to 'ADCM Metrics Reporting Task'. An 'OK' button is at the bottom right of the modal.

Name	Type	Bundle	Status
ADCM Metric Reporter Service	GraphiteMetricReporterService		

Controller Service Details | GraphiteMetricReporterService 1.20.0

SETTINGS | PROPERTIES | COMMENTS

Name
ADCM Metric Reporter Service

ID
41c3376f-018d-1000-ffff-ffff978421d7

Type
GraphiteMetricReporterService 1.20.0

Bundle
org.apache.nifi - nifi-metrics-reporting-nar

Supports Controller Service

- MetricReporterService 1.20.0 from org.apache.nifi - nifi-metrics-reporter-service-api-nar

Bulletin Level
WARN

Referencing Components

- Reporting Tasks (1)
 - ADCM Metrics Reporting Task

OK

Установка и настройка ADS Control (ADSC)

Узлы кластеров и планирование ролей

	Edge Nodes	ET Nodes
ADS Control		
Enterprise Tools		

Для процесса offline-установки необходим отдельный кластер ADS Control и предварительно установленный кластер Enterprise Tools (ET) с загруженным пакетом для offline-установки ADS Control .

1-ый хост – Arenadata Enterprise Tools

2-ой хост – ADS Control

Аппаратные требования к серверам

Arenadata Enterprise Tools, ADS Control
Storage: <ul style="list-style-type: none">• OS + Data: HDD• Volume: 1 TB CPU: 2 RAM: 4 GB

Системные требования: ADS Control, Arenadata Enterprise Tools



- ADS Control (ADSC)

Компонент	Требования
Операционная система	Centos 7.6.1810;
	RHEL 7.6;
Браузеры, поддерживающие сервис ADS Control	AltLinux sp 8.4 — доступно только в Enterprise-версии;
	Astra Linux 1.7 "Опел" SE — доступно только в Enterprise-версии;
	Chrome
	FireFox
	Edge
	Safari
Пользовательский интерфейс поддерживается в текущей стабильной версии этих браузеров и в предыдущей	

- Arenadata Enterprise Tools (ET)

Компонент	Требования
Операционная система	Centos 7.6.1810;
	RHEL 7.6;
	AltLinux sp 8.4 — доступно только в Enterprise-версии;
	Astra Linux 1.7 "Опел" SE — доступно только в Enterprise-версии;
RedOS 7.3.2 — доступно только в Enterprise-версии.	

Настройка инфраструктуры

- Выбираем Хост провайдер SSH Common Bundle
- На основе провайдера создаем необходимые хосты
- Настраиваем способ подключения (логин/пароль или по ключу)
- Если выбран облачный провайдер, то узлы создаются в облаке с указанными параметрами VM (CPU,Memory,...)

Основные этапы. Создание хоста

- Добавляем хост для узла

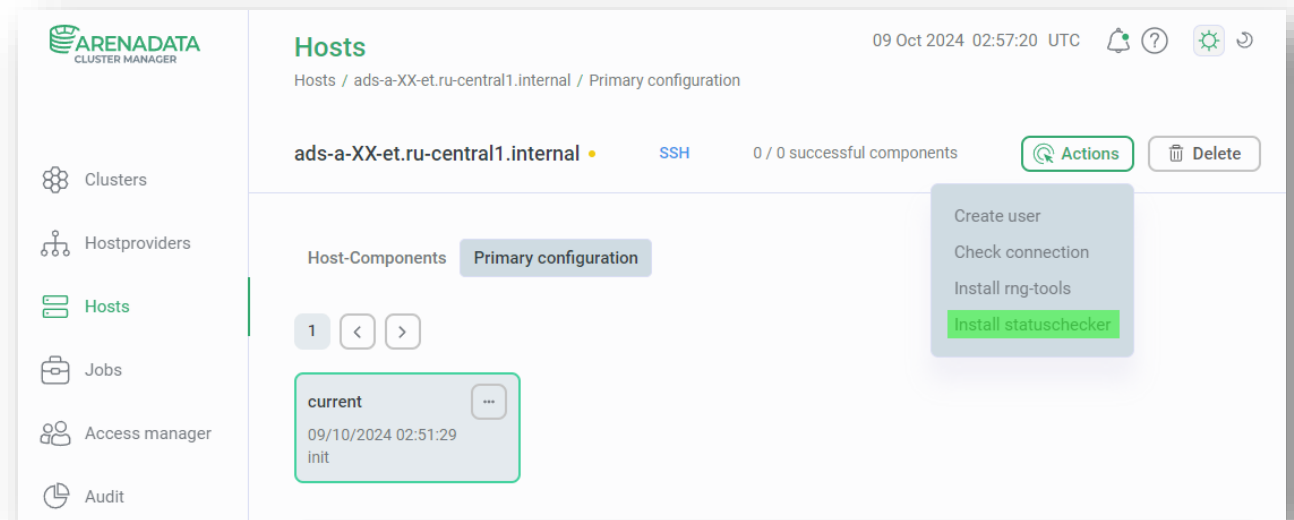
Hosts → Create Host

- Заполняем поля (Hostprovider, Name)

- Настраиваем поля для созданного хоста:

Hosts → <хост>

- *Username/Password*: указываем данные для подключения и/или настраиваем SSH Private Key (опционально)
- Connection address: указать fully.qualified.domain.name (FQDN) узла
- Выполняем проверку соединения (**Check connection**) и устанавливаем statuschecker (**Install statuschecker**)
- Проверить статусы операций в меню **Jobs**
- Прodelать действия для всех узлов



Создание кластера Arenadata Enterprise Tools

- Загружаем советующий бандл и создаем кластер ET:

Create cluster

Product

Arenadata Enterprise Tools

Product version

2023122800-1 (community)

Cluster name

ET

Description

Description

☒ I accept [Terms of Agreement](#)

Cancel

Create

Узлы кластера ET и планирование ролей. Установка Offline

Кластер	Сервис	Компонент	ads-a-<##>-et
ET	HTTP Mirror	HTTP Repo Server	1
	Docker Registry	Docker Registry	1
Всего компонент			2

Пример распределения компонент для учебного кластера!!!

- Добавляем службы HTTP Mirror и Docker Registry и распределяем на хост ads-a-<##>-et:

Add services

☐ All services

Search services

Select filtered

☐ Diamond

☒ Docker Registry

☐ Grafana

☐ Graphite

☒ HTTP Mirror

Cancel

Add

- Скачиваем архив et_pack.sh.xz и распаковываем его в /tmp/ на хосте ads-a-<##>-et

- Перейти к списку кластеров и для ET кластера выбрать действие Offline Install:

ARENADATA CLUSTER MANAGER

Clusters

Clusters / ET / Overview

Clusters

Hostprovide

Hosts

Install

Offline Install

Upload Pack

- Указать путь к offline-паку ET (в локальной файловой системе хоста ET) и нажать кнопку Run

Run an action: Offline Install

Search input

Expand content

Show advanced

Cancel

Run

Configuration

Packfile: /tmp/et_pack.sh

Демонстрация

1. Создать хост для узла `ads-a-XX-et.ru-central1.internal`
2. Установить кластер Arenadata Enterprise Tools на узел `ads-a-XX-et.ru-central1.internal`.

Создание кластера ADS Control

- Загружаем советующий бандл и создаем кластер ADSC:

Create cluster

Product

ADS Control

Product version

2.1.1_b1-1 (enterprise)

Cluster name

ADSC

Description

Description

☒ I accept [Terms of Agreement](#)

Cancel

Create

Узлы кластера ADSC и планирование ролей. Установка Offline

Кластер	Сервис	Компонент	ads-a-<##>-adsc
ADS Control	ADS Control	ADS Control	1
Всего компонент			1

Пример распределения компонент для учебного кластера!!!

- Добавляем службу ADS Control и распределяем на хост ads-a-<##>-adsc:

Add services

☒ All services

Select filtered

☒ ADS Control

Cancel

Add

- Перейти на вкладку **Services** кластера ADSC, открыть конфигурацию сервиса ADS Control и указать пароль для пользователя admin (административный пользователь для входа в Web UI консоль ADS Control):

ADSC

ADS Control 2.1.1

Primary configuration

Configuration groups

Components

Info

1

<

>

draft

current

13/06/2024 09:06:00

init

Search input

Expand content

Show advanced

Configuration

General

Basic authentication

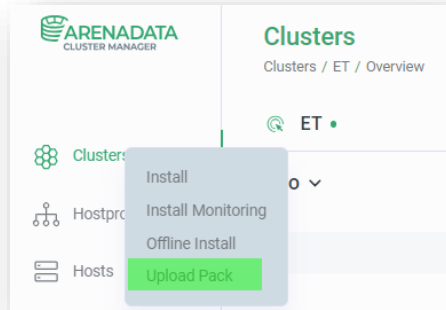
Admin Password: <secret>

Basic authentication users <not set>

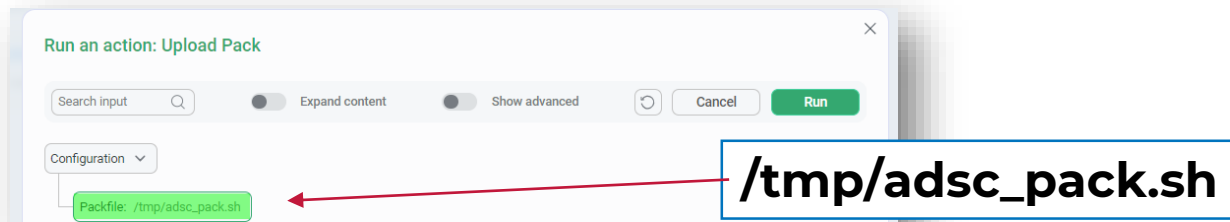
SSL

Установка ADSC Offline

- Скачиваем архив `adsc_pack.sh.xz` и распаковываем его в `/tmp/` на хосте `ads-a-###-et` (offline-пак уже распакован в файл `/tmp/adsc_pack.sh`)
- Перейти к списку кластеров и для ET кластера выбрать действие Upload Pack:

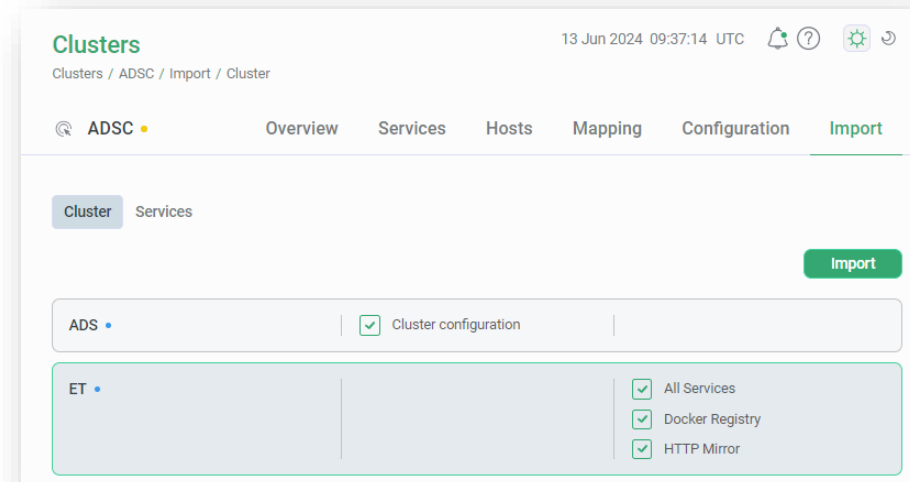


- Указать путь к offline-паку ET (в локальной файловой системе хоста ET) и нажать кнопку Run



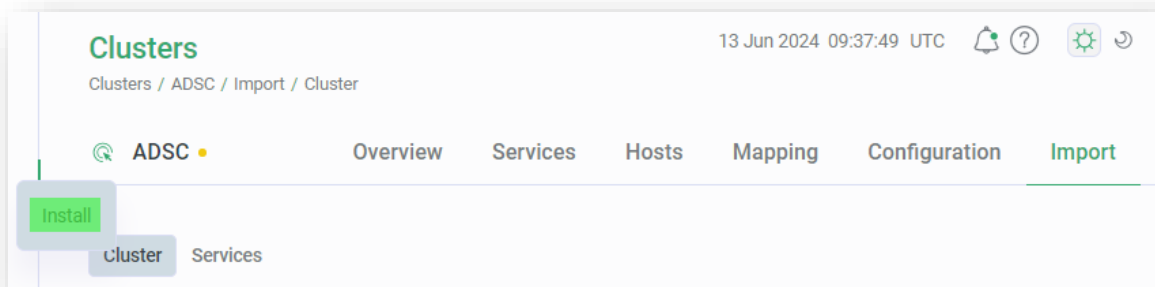
После выполнения Upload Pack, должна появиться директория:
`/opt/arenadata/etools/registry/docker/registry/v2/repositories/hub.adsw.io/adsc-community/`

- Перейти на вкладку **Import** кластера ADSC, импортировать сервисы Docker Registry, HTTP Mirror, импортировать необходимые кластеры ADS и нажать кнопку **Import**:

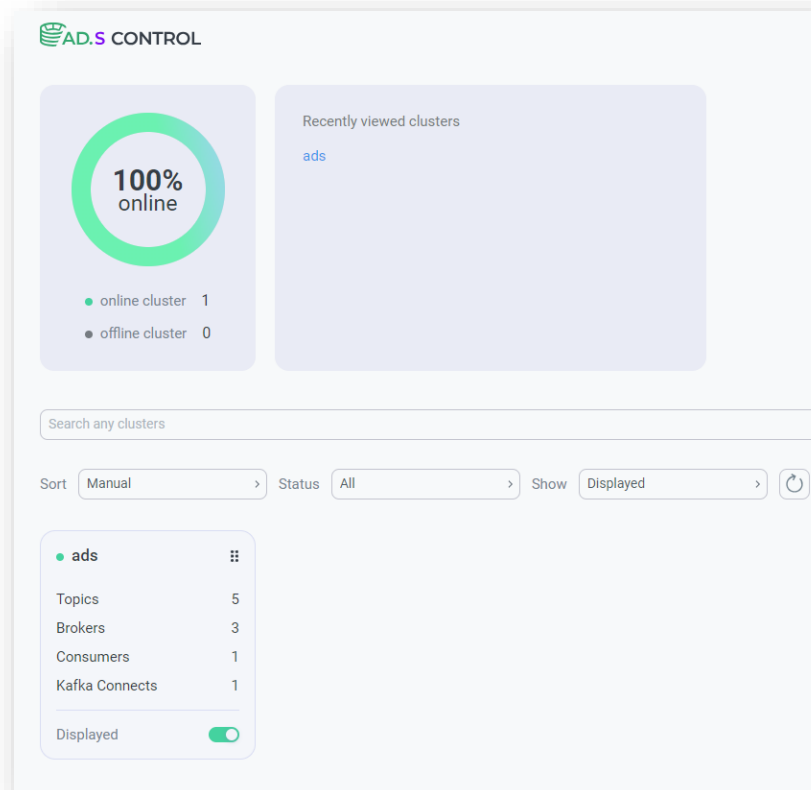


Установка ADSC Offline

- В верхнем левом углу открыть выпадающий список действий кластера ADSC и выбрать кластерное действие **Install**:



- После установки кластера, проверить доступность по адресу <http://ads-a-XX-adsc.ru-central1.internal:8888/>:



Демонстрация

- Создать хост для узла **ads-a-XX-adsc.ru-central1.internal**
- Установить кластер Arenadata Streaming Control на хост **ads-a-XX-adsc.ru-central1.internal**