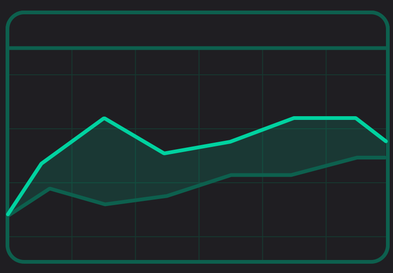
ADQM CONTROL

Arenadata QuickMarts Control



Arenadata QuickMarts Control (ADQM Control) is an observability platform for Arenadata QuickMarts (ADQM) clusters that allows you to significantly simplify the administration and optimize the operation of ADQM databases.

Version 1.3.1 Language: EN

Contents

1. **Concepts**
   - ADQM Control overview
2. **Planning guide**
   - Software requirements
3. **Get started**
   - Installation
     - Online installation of ADQM Control
       - Create a cluster
       - Add services
       - Add hosts to a cluster
       - Add components
       - Configure services
       - Configure a cluster
       - Integrate with an ADQM cluster
       - Install a cluster
       - Install monitoring
     - Offline installation of ADQM Control
       - Create a cluster
       - Add services
       - Add hosts to a cluster
       - Add components
       - Configure services
       - Configure a cluster
       - Integrate with an ADQM cluster
       - Import ET settings
       - Install a cluster
       - Install monitoring
   - Connect to ADQM Control
4. **Access management**
   - Manage users
   - SSL encryption
5. **Cluster maintenance**
   - Cluster actions
   - Service actions
   - Configure an external database
6. **Web user interface**
   - Alert settings
   - View a dashboard
   - Cluster metrics
   - Tables
   - Queries
   - Data cleanup
7. **References**
   - Configuration parameters
   - Query exception codes
   - Monitoring metrics
8. **Release notes**
   - ADQM Control releases

ADQM Control overview

Elena Dvoryadkina

Contents

To Table of Contents

Features

Architecture

Features

Arenadata QuickMarts Control (ADQM Control) is an observability platform for Arenadata QuickMarts (ADQM) clusters that allows you to significantly simplify the administration and optimize the operation of ADQM databases.

Key objectives of ADQM Control are the following:

- Monitor the health of an ADQM cluster. ADQM Control can work with multiple ADQM clusters simultaneously.
- Collect and analyze system metrics and metrics of ADQM services.
- Visualize the states of ADQM cluster hosts as a heat map.
- Generate alerts. An alert is a notification message that describes a problem detected in an ADQM cluster and provides recommendations how to fix it. ADQM Control groups alerts by triggers that cause them:
  - threshold-based alerts that are generated when some metric in the system has reached the specified threshold;
  - event-based alerts that are generated when some event has happened in the system.

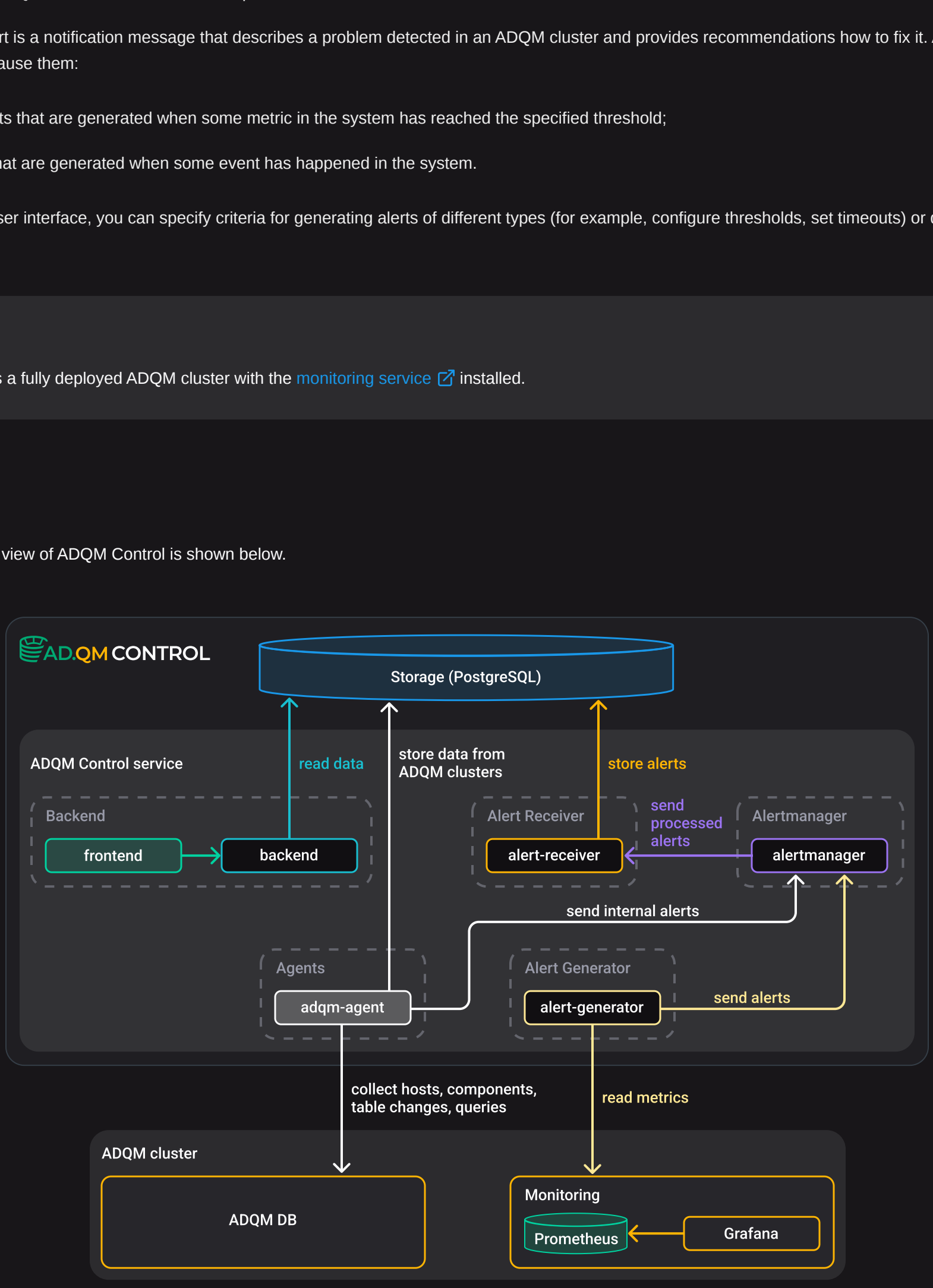
  In the ADQM Control user interface, you can specify criteria for generating alerts of different types (for example, configure thresholds, set timeouts) or disable tracking of some events.

NOTE

ADQM Control requires a fully deployed ADQM cluster with the monitoring service installed.

Architecture

The high-level architecture view of ADQM Control is shown below.



ADQM Control architecture

Main points of the scheme:

- ADQM Control is the main service that implements the full functionality of ADQM Control. It includes the following components:
  - Alert Generator. The alert-generator service reads metrics from a Prometheus instance of an ADQM cluster, generates alerts if necessary (taking into account criteria and timeouts for creating alerts specified in ADQM Control settings), sends alerts to Alertmanager.
  - Agents. The adqm-agent service receives data about hosts, components, queries, and table changes from the ADQM cluster's ADQMDB service and stores this information in the database (Storage), and also generates internal alerts related to ADQM and sends them to Alertmanager.
  - Alertmanager. The alertmanager service handles alerts received from Alert Generator and Agents (filters alerts, mutes alerts of some type, adds additional labels, etc.) and resends processed alerts to Alert Receiver.
  - Alert Receiver. The alert-receiver service receives processed alerts from Alertmanager and stores them in the database.
  - Backend. The backend service communicates with the ADQM Control web interface (frontend in the scheme) via REST API — handles user requests from the frontend, saves specified ADQM Control settings, retrieves alerts and data about the ADQM cluster from the storage to transmit them to the web interface.
- Storage is a PostgreSQL database for storing alerts, as well as data about users, tables, queries, and settings. You can set up it in one of the following ways:
  - install the Arenadata PostgreSQL service in the ADQM Control cluster;
  - specify an external PostgreSQL cluster using ADQM Control configuration parameters.

Software requirements

Elena Dvoryadkina

Contents

To Table of Contents

Operating system

ADCM

Operating system

ADQM Control supports only x86_64 architecture and the following operating systems:

- ALT Linux 10.2 SP
- Astra Linux SE 1.7.6 Orel with Axiom JDK
- Astra Linux SE 1.7.6 Voronezh
- Astra Linux SE 1.8.4 Orel and Voronezh
- RED OS 7.3 Certified edition
- RED OS 8.0 Certified edition
- RHEL 9
- Ubuntu 22.04 LTS

ADCM

It is required to use ADCM 2.7.1 or higher.

Installation

AD.QM CONTROL

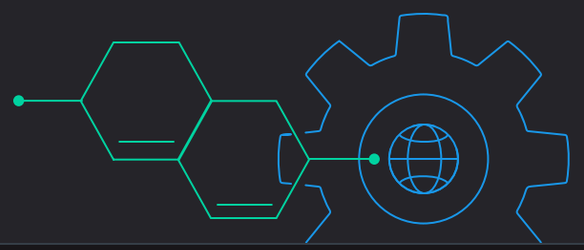
To Table of Contents

This tutorial guides you through the process of installing an Arenadata QuickMarts Control (ADQM Control) cluster. To install it, use Arenadata Cluster Manager (ADCM) — a separate Arenadata product designed for simple, convenient, and fast software deployment and exploitation.

You can install ADQM Control via ADCM in two ways.

Online installation

on hosts with access to Internet



Offline installation

on hosts without access to Internet



Online installation

To Table of Contents

This topic describes **online** installation of ADQM Control. It can be applied to the hosts with access to Internet.

Before installing ADQM Control according to the current guide, you should first install ADCM and prepare an ADQM cluster with the monitoring service. If you have not done this already, perform the steps 1-4 listed below (each of these steps is described in detail in the ADQM documentation).

Step 1.

Install ADCM

Step 2.

Prepare hosts

Step 3.

Install ADQM cluster

Step 4.

Install monitoring in ADQM

Step 5.

Install ADQM Control

Step 6.

Install monitoring in ADQM Control

Create a cluster

Elena Dvoryadkina, Daria Barysheva

AD.QM CONTROL

Contents

To Table of Contents

Step 1. Get the Enterprise version of a cluster bundle

Step 2. Upload a cluster bundle to ADCM

Step 3. Create a cluster based on the uploaded bundle

In ADCM, a **cluster** means a set of services running on a set of hosts.

Clusters distributions for ADCM are made in the form of so-called bundles. Generally a bundle is a regular archive that includes a description and software logic of cluster deployment and exploitation.

The steps for adding a cluster to ADCM are listed below.

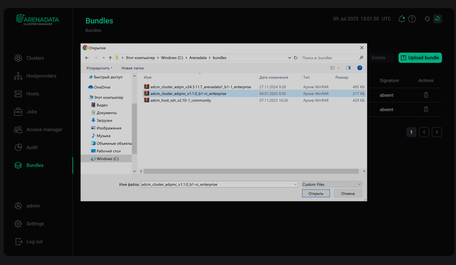
Step 1. Get the Enterprise version of a cluster bundle

Please get a cluster bundle from the Arenadata support team.

Step 2. Upload a cluster bundle to ADCM

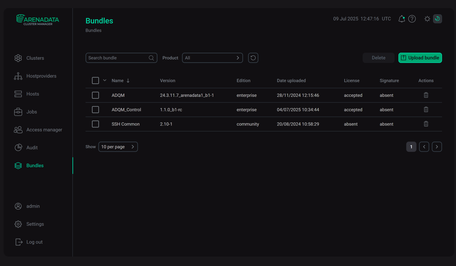
To upload a bundle to ADCM, follow the steps:

1. Select the ***Bundles*** item in the left navigation menu and click ***Upload bundle***.
2. Select a bundle in the Open File Dialog.



Upload a bundle

3. As a result of the performed actions, a bundle is displayed on the ***Bundles*** page.



The result of successful uploading a bundle

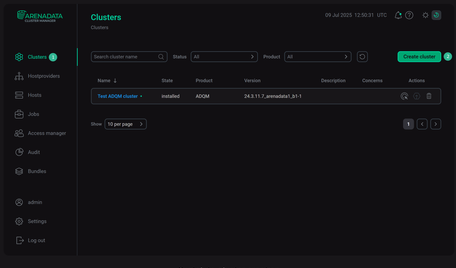
Step 3. Create a cluster based on the uploaded bundle

IMPORTANT

When creating a cluster via the ADCM web interface, you generate a new ADCM cluster instance. It means only adding the cluster to ADCM — at this stage no services are deployed.

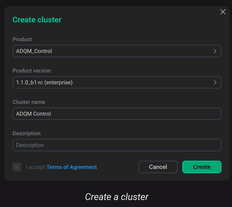
The steps for cluster creation are given below:

1. Select the ***Clusters*** item in the left navigation menu and click ***Create cluster***.



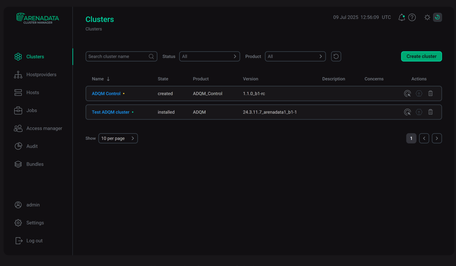
Switch to cluster creation

2. In the opened dialog, describe a new cluster instance:
   - Select an uploaded cluster bundle in the ***Product*** field.
   - Select a bundle version in the ***Product version*** field. Several versions become available in the case of different versions of the same bundle being uploaded.
   - Enter a cluster name in the ***Cluster name*** field.
   - Enter a cluster description in the ***Description*** field if necessary.
   - Set the ***I accept Terms of Agreement*** flag to sign the license agreement. This flag is displayed only for Enterprise bundles. To read the agreement text, refer to the ***Terms of Agreement*** link. You can also sign the license agreement on the ***Bundles*** page by clicking the desired product bundle.
   - Click ***Create***.



Create a cluster

3. As a result, the created cluster instance is displayed on the ***Clusters*** page.



The result of successful cluster creation

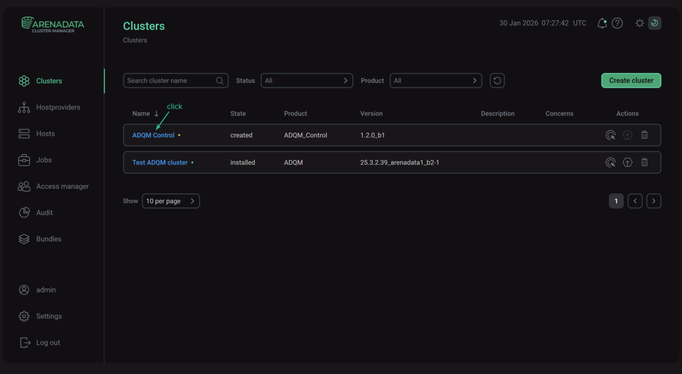
Add services

Elena Dvoryadkina, Daria Barysheva

To Table of Contents

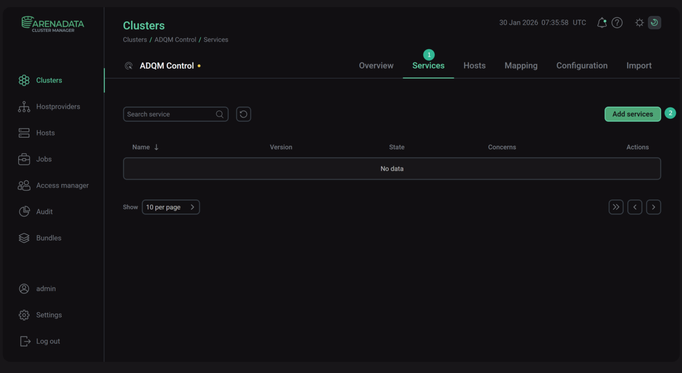
In ADCM, a **service** means a software that performs some function. The steps for adding services to a cluster are listed below:

1. Select a cluster on the ***Clusters*** page. To do this, click a cluster name in the ***Name*** column.



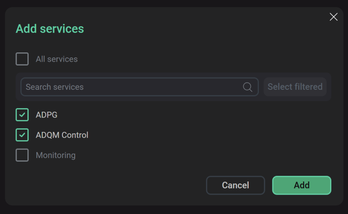
Select a cluster

2. Open the ***Services*** tab on the cluster page and click ***Add services*** (***Add service*** in first versions of ADCM).



Switch to adding services

3. In the opened dialog, select services that should be added to the cluster and click ***Add***.



Select services

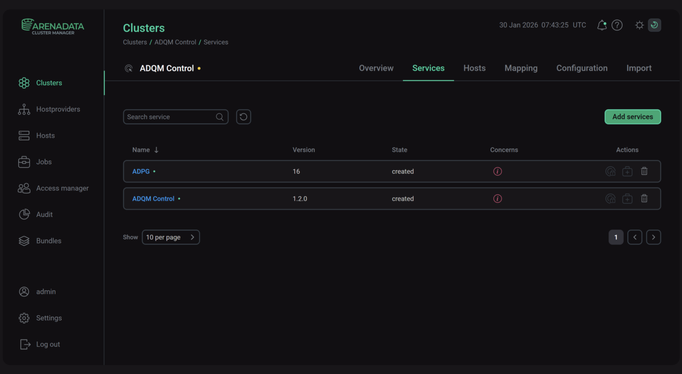
The brief description of available services is listed below.

Services that can be added to the ADQM Control cluster

Service	Mandatory	Purpose
ADQM Control	Yes	Main service that provides the full functionality of the ADQM Control product
ADPG	No	Arenadata Postgres database to store ADQM Control-specific data
Monitoring	No	Service for collecting and storing ADQM Control cluster monitoring metrics based on the Prometheus monitoring system and Grafana service

For an ADQM Control cluster, it is required to select a PostgreSQL database to store ADQM Control data. To do this, you can add the ADPG service to the ADQM Control cluster or specify an external PostgreSQL server through the corresponding settings of the ADQM Control service.

4. As a result, the added services are displayed on the ***Services*** tab.



The result of successful adding services to a cluster

Add hosts to a cluster

Elena Dvoryadkina, Daria Barysheva

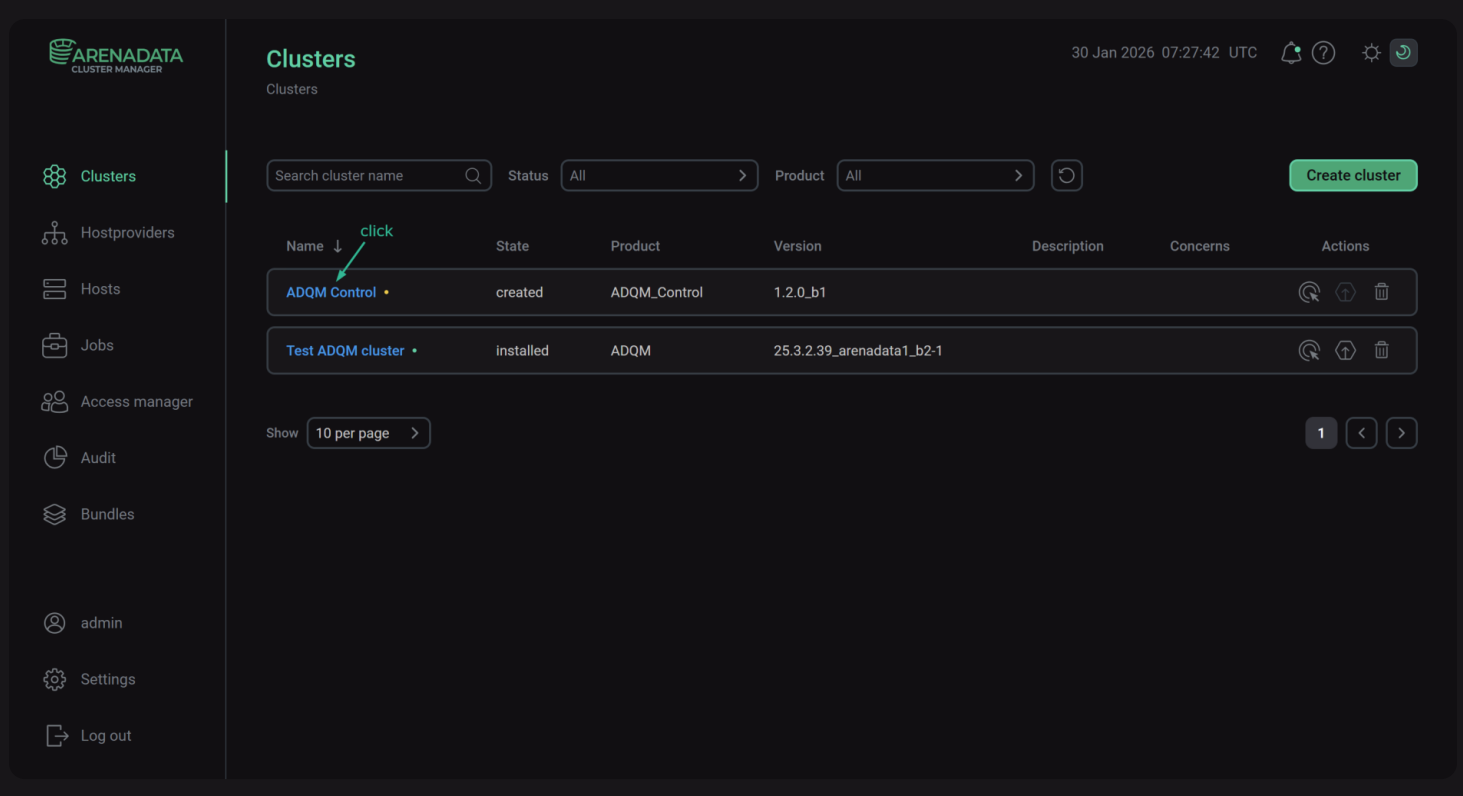
To Table of Contents

NOTE

To set up and install an ADQM Control cluster via ADCM, you should first prepare hosts for the cluster and link these hosts to ADCM. For a detailed description of how to do this, see Prepare hosts in the ADQM documentation.

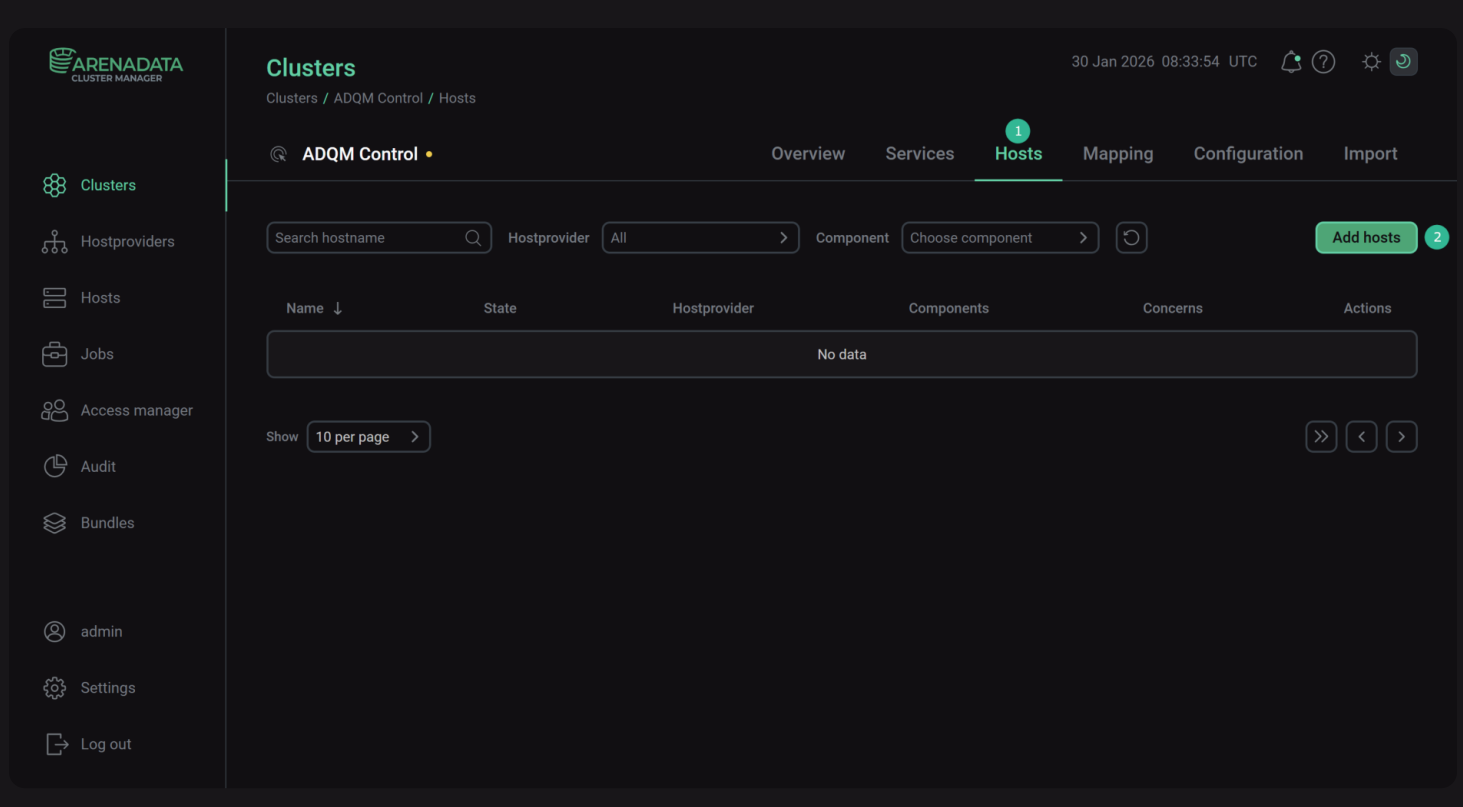
The steps for adding hosts to a cluster are listed below:

1. Select a cluster on the ***Clusters*** page. To do this, click a cluster name in the ***Name*** column.



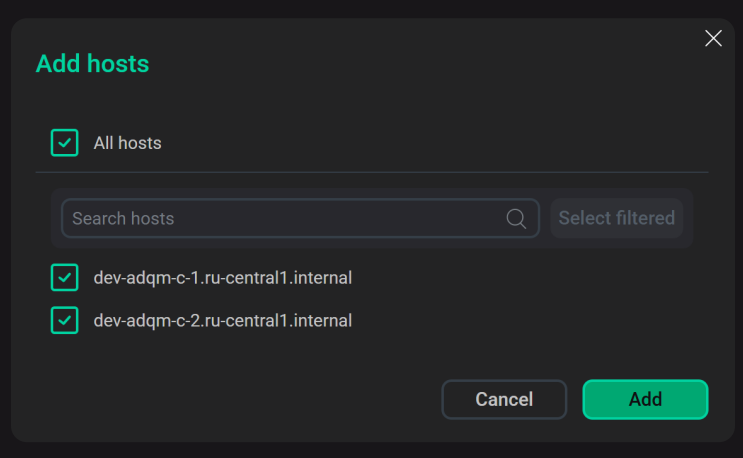
Select a cluster

2. Open the ***Hosts*** tab on the cluster page and click ***Add hosts***.



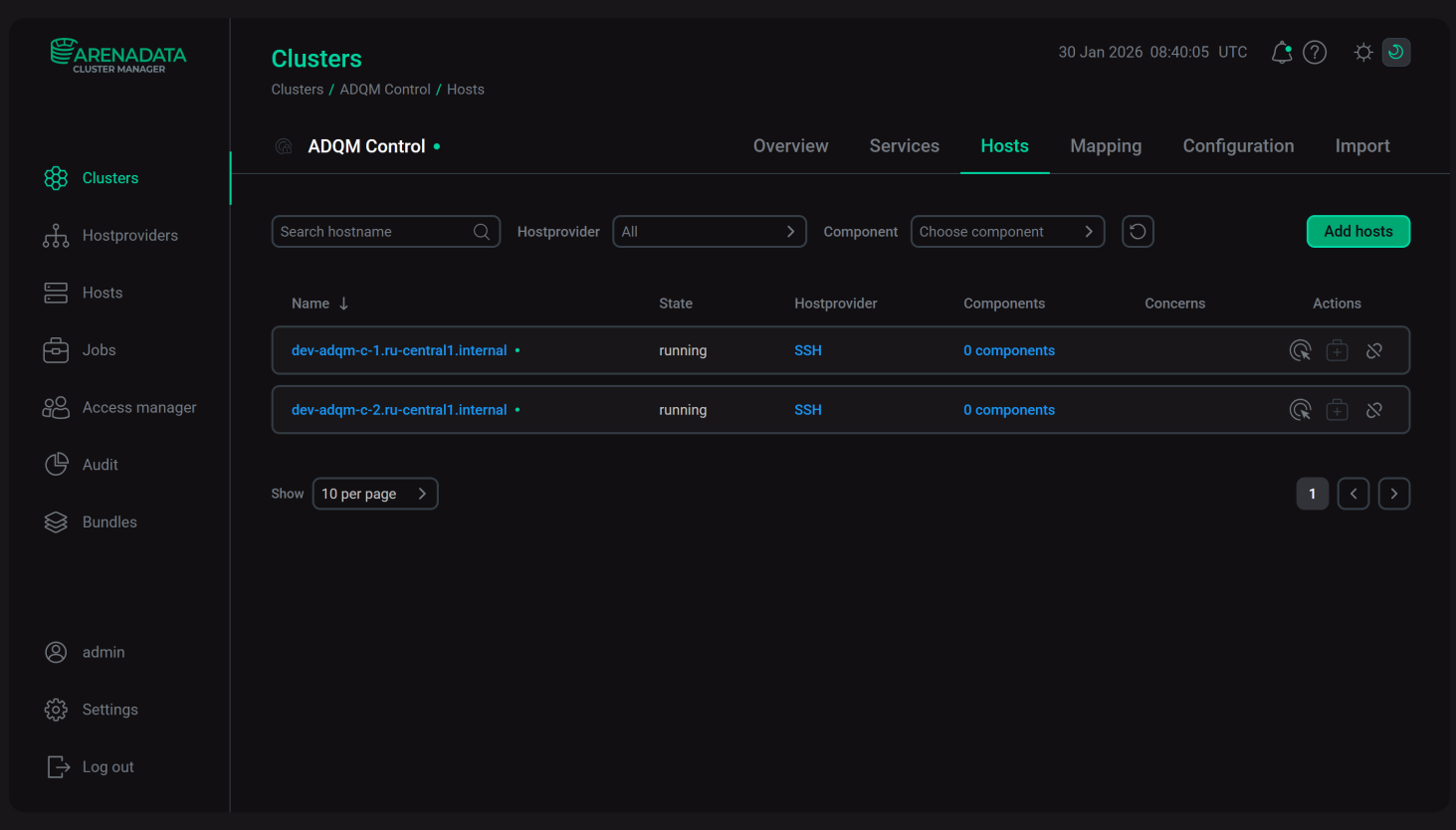
Switch to adding hosts

3. In the opened dialog, select hosts that should be added to the cluster and click ***Add***.



Select hosts

4. As a result, the added hosts are displayed on the ***Hosts*** tab.



The result of successful adding hosts to a cluster

Add components

Elena Dvoryadkina, Daria Barysheva

ADQM CONTROL

To Table of Contents

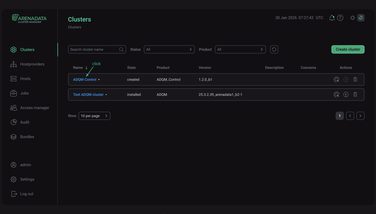
In ADCM, a **component** means a part of a service that should be deployed on one or several cluster hosts.

IMPORTANT

Initially, there are no components on any of cluster hosts. You should map components to hosts manually.

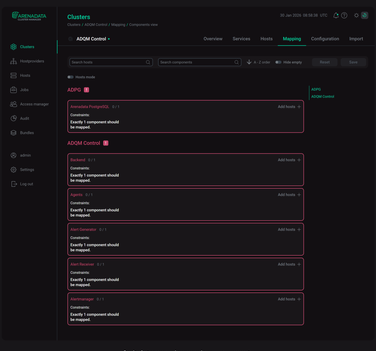
The steps for adding components to hosts are listed below:

1. Select a cluster on the ***Clusters*** page. To do this, click a cluster name in the ***Name*** column.



Select a cluster

2. Open the ***Mapping*** tab on the cluster page.



Switch to mapping service components

All components of ADQM Control services are mandatory (highlighted in red). The ADCM UI also displays how many components should be installed.

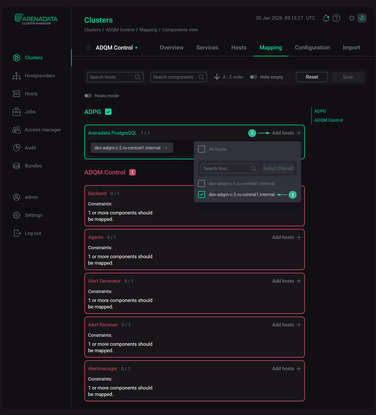
CAUTION

Without assigning a necessary number of hosts to mandatory components, the component mapping cannot be saved.

Components of ADQM Control services

Service	Component	Notes
ADQM Control	Backend	Implements the product's web interface
	Agents	Collects information on states of hosts and metrics from ADQM clusters
	Alert Generator	Generates alerts and sends them to the alert manager
	Alert Receiver	Receives alerts from the alert manager and writes them to the data storage
	Alertmanager	Handles alerts (mutes alerts of some type, filters alerts, adds additional labels) and resends processed alerts to the alert receiver
ADPG	Arenadata PostgreSQL	Data storage

3. In the window that opens, map components to hosts. To assign hosts for each component, click ***Add hosts*** and select hosts in the pop-up window.

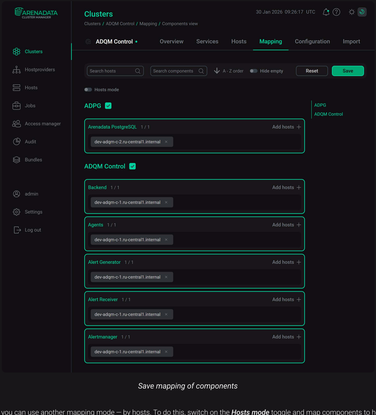


Select hosts for a component

TIP

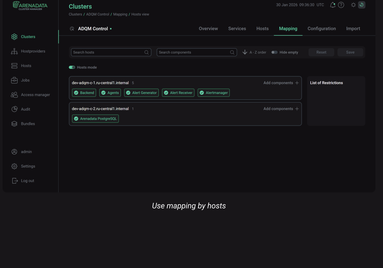
It is recommended to install the Arenadata PostgreSQL component on a dedicated host, separately from the components of the ADQM Control service.

4. After component distribution is completed, click ***Save***.



Save mapping of components

Starting with ADCM 2.2.0, you can use another mapping mode — by hosts. To do this, switch on the ***Hosts mode*** toggle and map components to hosts by clicking ***Add components*** for each host sequentially.

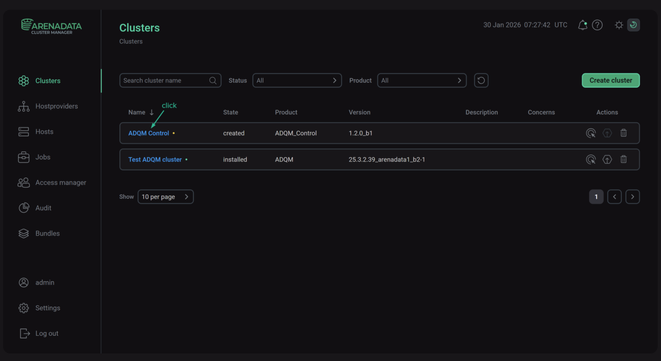


Use mapping by hosts

Configure services

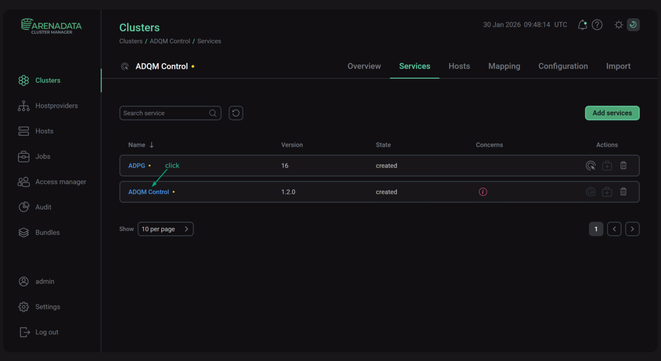
Elena Dvoryadkina, Daria Barysheva

1. Select a cluster on the ***Clusters*** page. To do this, click a cluster name in the ***Name*** column.



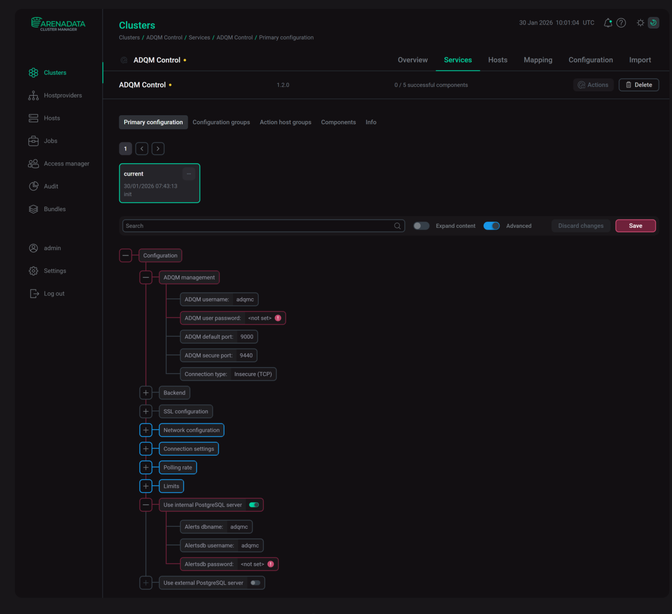
Select a cluster

2. Open the ***Services*** tab on the cluster page, find the row that contains the desired service, and click its name in the ***Name*** column.



Switch to configuring a service

3. In the window that opens:
   - Open the ***Primary Configuration*** tab.
   - Switch on the ***Show advanced*** toggle.
   - Fill in configuration parameters of the selected service. The fields highlighted in red are mandatory.
   - Click ***Save***.



Configure a service

ADQM user credentials

In the ***ADQM username*** and ***ADQM user password*** parameters of the ADQM Control service, specify the name and password of an ADQM user to be used by ADQM Control agents for connecting to ADQM. For this, you can use the `default` ADQM user account with an assigned password or create a separate user account in ADQM (recommended), for example:

```
CREATE USER adqmc ON CLUSTER default_cluster IDENTIFIED WITH sha256_password BY 'qwerty123';
```

```
GRANT ON CLUSTER default_cluster SELECT ON *.* TO adqmc;
```

For more information about all configuration parameters of ADQM Control services, see Configuration parameters.

Configure a cluster

Elena Dvoryadkina, Daria Barysheva

Contents

To Table of Contents

Primary configuration

Configuration groups

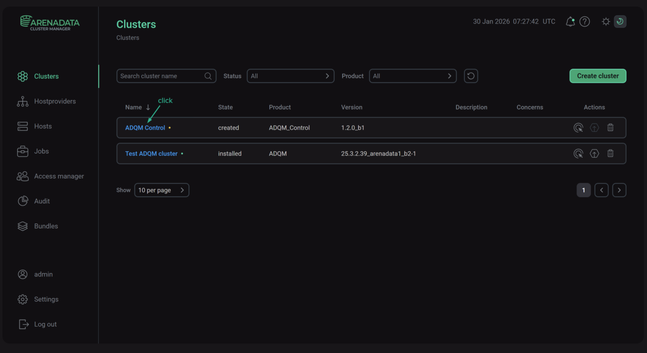
Ansible settings

IMPORTANT

There is usually no need to change cluster configuration parameters. You can leave all parameters at the default values.

After creation a new cluster, you can configure it by performing the following steps:

1. Select a cluster on the ***Clusters*** page. To do this, click a cluster name in the ***Name*** column.



Select a cluster

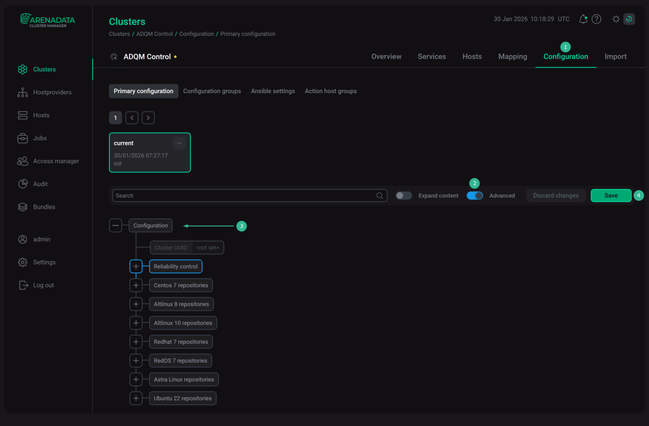
2. Open the ***Configuration*** tab on the cluster page. The ***Configuration*** tab includes the following sections: ***Primary configuration***, ***Configuration groups***, ***Ansible settings***.
3. Fill in all necessary parameters on the selected tab and click ***Save***.

Primary configuration

The ***Configuration*** → ***Primary configuration*** tab allows you to configure the general cluster settings.

ADQM Control cluster configuration parameters are listed below:

- ***Reliability control*** — the section with the following settings (visible if the ***Show advanced*** option is enabled):
  - ***retries*** — number of retries for cluster operations (`3` by default);
  - ***delay*** — delay between retries in seconds (`10` by default);
  - ***timeout*** — timeout for checks of network ports in seconds (`60` by default).
- Paths to the repositories that are used during the installation process — depending on the cluster operating system (Centos, Alt Linux, Red Hat, RED OS, Astra Linux, or Ubuntu):
  - ***ADQMC_arenadata_community*** — repository required to install the Community edition of ADQM Control;
  - ***arenadata_postgres*** — repository required to install Arenadata PostgreSQL;
  - ***prometheus_repo*** — repository required to install Prometheus;
  - ***ADQMC_arenadata_enterprise*** — repository required to install the Enterprise edition of ADQM Control.



Cluster configuration window

Configuration groups

The ***Configuration*** → ***Configuration groups*** tab is designed to set cluster configuration groups.

Ansible settings

The ***Configuration*** → ***Ansible settings*** tab allows you to set Ansible configuration options at the cluster level. The tab is available starting with ADCM 2.2.0.

Ansible settings

Name	Description	Default
forks	The number of parallel processes to spawn when communicating with remote hosts	5

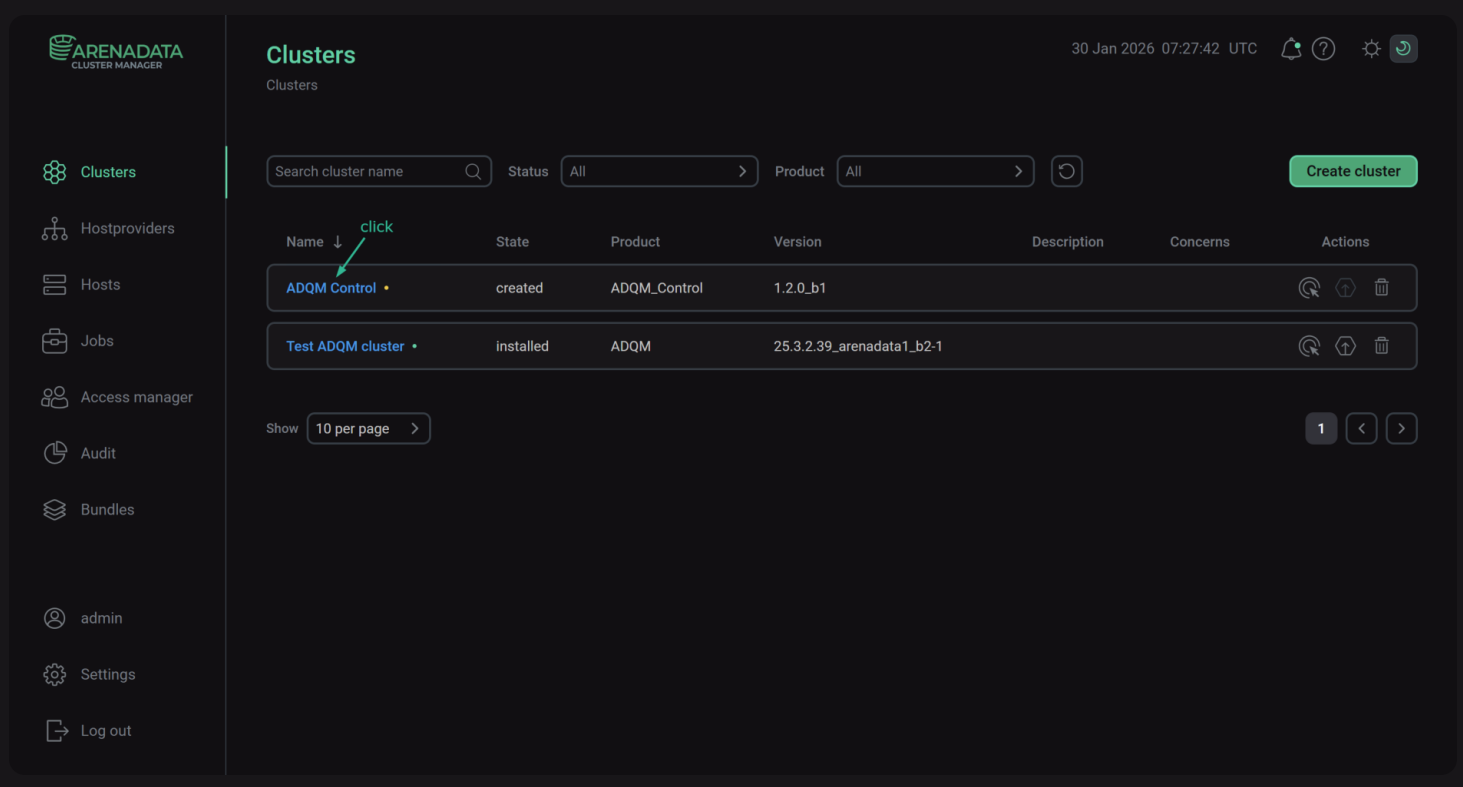
Integrate with an ADQM cluster

Elena Dvoryadkina

To Table of Contents

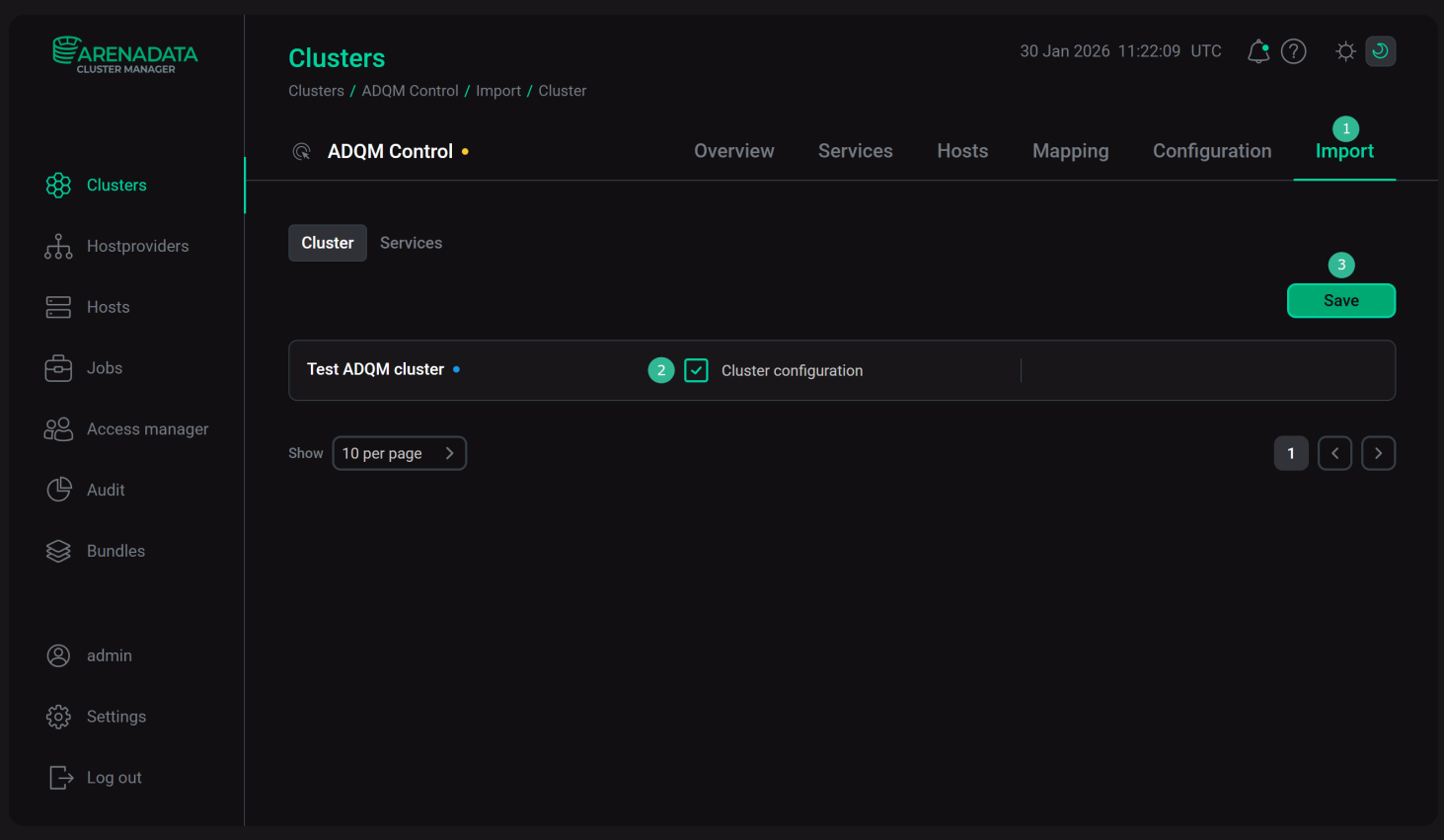
To set up an ADQM Control cluster, you should integrate it with at least one pre-installed ADQM cluster with the Monitoring service configured:

1. On the ***Clusters*** page, click an ADQM Control cluster name in the ***Name*** column.



Select a cluster

2. Open the ***Import*** tab.
3. Select ***Cluster configuration*** for the ADQM cluster to be imported into ADQM Control.
4. Click ***Save***.



Integration with an ADQM cluster

NOTE

A single ADQM Control cluster can also be integrated with multiple ADQM clusters.

Install a cluster

Elena Dorosyadkina, Elena Barysheva

ADQM CONTROL

Contents

To Table of Contents

Step 1. Run installation
- Whole cluster
- Single services

Step 2. View the results of installation

Step 3. Check the cluster state after installation

Step 4. Check the ADQM Control web interface

Step 1. Run installation

There are two ways to install cluster services:

1. **Whole cluster** - in this method all services are installed automatically one by one after applying the *Install* action to a cluster.
2. **Single services** - In this method each service is installed manually after applying one or more actions to it. This way is also suitable for installing new services in the already deployed cluster.

In both cases each service starts automatically after installation.

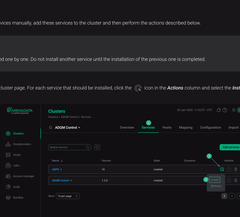
NOTE

In case of errors, you can find logs on the **Jobs** page.

Whole cluster

To install all cluster services within a single action, follow the steps:

1. Apply the *Install* action to the cluster selected on the *Clusters* page by clicking the icon in the **Actions** column.



Switch to the whole cluster installation

2. Wait until the job is completed.

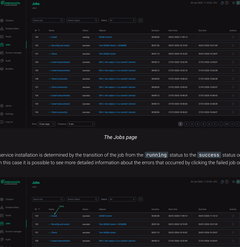
Single services

To install one or several cluster services manually, add these services to the cluster and then perform the actions described below.

CAUTION

Notice that services are installed one by one. Do not install another service until the installation of the previous one is completed.

1. Open the **Services** tab on the cluster page. For each service that should be installed, click the icon in the **Actions** column and select the *Install* action.



Switch to installation of single services

2. Wait until the job is completed. Then check that the service state has changed from `created` to `installed`.
3. Repeat the previous actions for other services that should be installed.

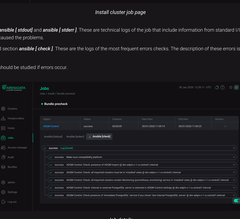
IMPORTANT

When you install ADQM Control services manually, one by one, use the following order:

1. ADPG (if a connection to an external PostgreSQL cluster is not configured)
2. ADQM Control

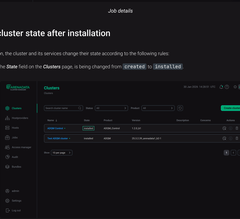
Step 2. View the results of installation

ADCM starts a single job for installation process, as well as for any other task. You can find out about the status of jobs more specifically on the **Jobs** page.

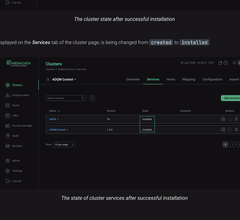


The Jobs page

The successful completion of the service installation is determined by the transition of the job from the `running` status to the `success` status on the **Jobs** page. If the job fails, it switches to the `failed` status. In this case it is possible to see more detailed information about the errors that occurred by clicking the failed job on the **Jobs** page.



Switch to a job page

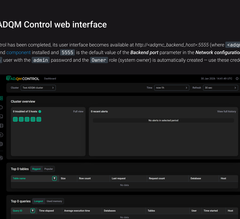


Install cluster job page

A job page contains two sections: ***ansible [stdout]*** and ***ansible [stderr]***. These are technical logs of the job that include information from standard I/O streams stdout/stderr. These logs can help to understand what caused the problems.

There can also be the optional third section ***ansible [check]***. These are the logs of the most frequent errors checks. The description of these errors is more simple and specific, than in two previous technical logs.

The contents of all three sections should be studied if errors occur.

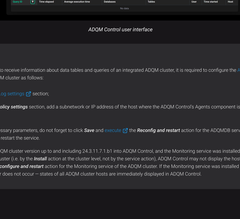


Job details

Step 3. Check the cluster state after installation

As a result of successful installation, the cluster and its services change their state according to the following rules:

- The cluster state, displayed in the *State* field on the **Clusters** page, is being changed from `created` to `installed`.



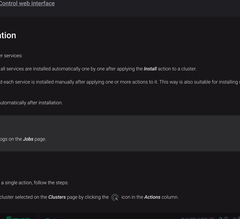
The cluster state after successful installation

- The state of cluster services, displayed on the **Services** tab of the cluster page, is being changed from `created` to `installed`.

The state of cluster services after successful installation

Step 4. Check the ADQM Control web interface

After the installation of ADQM Control has been completed, its user interface becomes available at http://<adqmc_backend_host>:5555 (where `<adqmc_backend_host>` is the IP address of the host with the Backend component installed and `5555` is the default value of the *Backend port* parameter in the *Network configuration* section of the ADQM Control service configuration). The `admin` user with the `admin` password and the `Senior` role (system owner) is automatically created - use these credentials to log in to ADQM Control.



ADQM Control user interface

IMPORTANT

- In order for ADQM Control to receive information about data tables and queries of an integrated ADQM cluster, it is required to configure the ADQMDB service parameters for the ADQM cluster as follows:
  - enable ***query_log*** in the Log settings section;
  - in the ***Default user and policy settings*** section, add a subnetwork or IP address of the host where the ADQM Control's Agents component is installed to the ***Default user IP list***.

  After you have set the necessary parameters, do not forget to click *Save* and execute the ***Reconfig and restart*** action for the ADQMDB service to save configuration changes and restart the service.
- If you have imported an ADQM cluster version up to and including 24.3.1.17.1.b1 into ADCM, and the Monitoring service was installed during the installation of the ADQM cluster (i.e. by the *Install* action at the cluster level, not by the service action), ADQM Control may not display the hosts of that cluster. To fix this issue, run the ***Reconfigure and restart*** action for the Monitoring service of the ADQM cluster. If the Monitoring service was installed into an existing ADQM cluster, this behavior does not occur – states of all ADQM cluster hosts are immediately displayed in ADQM Control.

Install monitoring

Table of contents

Contents

Overview
Step 1. Add the service
Step 2. Add components
Step 3. Configure the service
Step 4. Install the service
Step 5. View results
- Metrics in the Prometheus format
- Prometheus web interface
- Grafana web interface

The Monitoring service deploys the Prometheus server inside ADQM Control to collect and store ADQM Control cluster monitoring metrics, and also supports the ability to use the Grafana UI web application for visualization and analysis of information. This article describes the steps required to add this service to an ADQM Control cluster.

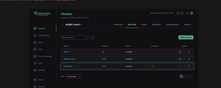
Overview

When adding the Monitoring service to ADQM Control, you can install the node exporter on all or selected hosts of your cluster. It is a monitoring agent that exports host system metrics. Prometheus can then collect these metrics from the ADQM Control server and from the node exporter. The collected metrics are then available in the Grafana web application, specified in the Grafana configuration. You can also use the Prometheus and Grafana web interfaces to view and analyze data that the monitoring service collects.

> **NOTE**
>
> - If you already have a Prometheus monitoring system set up (for example, your own Prometheus server or VictoriaMetrics), you can use it to collect ADQM Control metrics. To do this, configure access to ADQM Control metrics in your monitoring system using ADQM Control's Prometheus parameters specified on the Monitoring service page.
> - You can also use the VictoriaMetrics implementation to integrate all metrics from the Prometheus server deployed in ADQM Control into VictoriaMetrics.

Step 1. Add the service

1. In the ADCM interface, open the **Clusters** page and click your ADQM Control cluster name. On the cluster page that opens, switch to the **Services** tab and click **Add services**.



Switch to adding services

2. In the opened dialog, select the **Monitoring** service and click **Add**.



Select the service

As a result, the added service is displayed on the **Services** tab.



The result of successfully adding the service to the cluster

Step 2. Add components

1. On the cluster page, open the **Mapping** tab to proceed to mapping service components to cluster hosts.



Switch to mapping service components

Monitoring service components

Component	Description
Prometheus Server	Stores and allows requesting all metrics for an ADQM Control cluster
Grafana	Visualizes ADQM Control metrics as graphs and charts organized into dashboards
Node Exporter	Collects system metrics and sends them to Prometheus

2. Assign hosts to components of the Monitoring service – click **Add hosts** and select the desired hosts in the pop-up window.



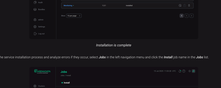
Select a host for a component

> **CAUTION**
>
> It is not recommended to install the Prometheus Server component on hosts with ADQM Control – can separate hosts for it. Otherwise, if an ADQM Control host fails or the load on it is critically high, information about the corresponding problems will not be saved.

3. After the distribution of components is completed, click **Save**.

Step 3. Configure the service

1. Open the **Services** tab on the cluster page and click the **Monitoring** service name in the **Name** column.



Switch to the service configuration

2. On the page that opens, fill in the service configuration parameters.



Configure the Monitoring service

Fields highlighted in red are required.

- **Prometheus settings** → **Password for Grafana connection** – a password of a Grafana user (`admin` or `grafana`) that is used to connect to Prometheus. The password should contain a combination of numbers, symbols, lowercase and uppercase letters and have a minimum length of 8.
- **Grafana settings** → **Grafana administrator's password** – a password of the `admin` Grafana user. The password should contain a combination of numbers, symbols, lowercase and uppercase letters and have a minimum length of 8.

All monitoring parameters are described in the **Monitoring** section of the Configuration parameters article.

3. After specifying all necessary parameters, click **Save**.

Step 4. Install the service

1. On the **Services** tab, click the icon in the **Actions** column and run the **Install** action.



Switch to the service installation

2. Wait until the installation is completed. Then check that the service state has changed from `created` → `installed`.

Installation is complete

To view the service installation process and analyze errors if they occur, select **Jobs** in the left navigation menu and click the **Install** job name in the **Jobs** list.



Install service job page

Step 5. View results

The Monitoring service starts automatically after installation. To ensure that monitoring works correctly, check that both system metrics and metrics of ADQM Control's services are collected in Prometheus – see Metrics in the Prometheus format. To do this, view metrics of the Prometheus format in the browser or use the Prometheus or Grafana web interface.

Metrics in the Prometheus format

1. In the address bar of your browser, enter an address of a host on which Clickhouse is deployed and add a port number + a value of the **Grafana listen port** parameter located in the **Grafana settings** section of the Monitoring service configuration (the default value is `3000`). For example, `http://135.181.205.13:3000`. To log in, use `admin` as a username, and the **Grafana administrator's password** parameter value (found in the **Grafana settings** section of service configuration parameters) as a password.
2. In the window that opens, navigate to **Home** → **Dashboards** and expand the **ADQM Dashboard ADQM_Control_cluster_name** section. In this section, you can select one of the available dashboards to view service's system metrics and metrics from your ADQM Control cluster.

View ADQM Control metrics in Grafana

Prometheus web interface

1. In the address bar of your browser, enter an address of a host on which Clickhouse is deployed and add a port number + a value of the **Prometheus web port** parameter in the **Service parameters** section of the Monitoring service configuration (the default value is `19090`). For example, `http://135.181.205.13:19090`. To log in, use the Prometheus interface, use a username and password that are also specified in the **Prometheus settings** section – the **Prometheus users in login:hash or Prometheus** setting.
2. In the **Expression** field, you can enter a metric and click **Execute** – values of this metric on all hosts of the ADQM Control cluster will be shown in the interface.

Prometheus web interface

Grafana web interface

1. In the address bar of your browser, enter an address of a host on which Grafana is deployed and add a port number + a value of the **Grafana listen port** parameter located in the **Grafana settings** section of the Monitoring service configuration (the default value is `3000`). For example, `http://135.181.205.13:3000`. To log in, use `admin` as a username, and the **Grafana administrator's password** parameter value (found in the **Grafana settings** section of service configuration parameters) as a password.
2. In the window that opens, navigate to **Home** → **Dashboards** and expand the **ADQM Dashboard ADQM_Control_cluster_name** section. In this section, you can select one of the available dashboards to view service's system metrics and metrics from your ADQM Control cluster.

View ADQM Control metrics in Grafana

Offline installation

To Table of Contents

This topic describes **offline** installation of ADQM Control. It can be applied to the hosts without access to Internet.

For offline installation, you need to install and configure Arenadata Enterprise Tools (ET). ET is a solution that provides the necessary infrastructure for Arenadata product deployment in environments with limited internet access. The solution is a part of the Arenadata data platform. It provides the repositories necessary for installing and updating the platform components.

Before installing ADQM Control according to the current guide, you should first install ADCM, deploy an ET cluster, and prepare an ADQM cluster with the monitoring service. If you have not done this already, perform the steps 1-5 listed below (each of these steps is described in detail in the ADQM documentation). Also, upload a special ADQM Control installation shell script to the ET cluster (you can request it from the Arenadata support team, as well as a shell script for offline installation of ADQM) — see Upload packages for offline installation.

Step 1.

Install ADCM

Step 2.

Prepare hosts

Step 3.

Install ET cluster

Step 4.

Install ADQM cluster

Step 5.

Install monitoring in ADQM

Step 6.

Install ADQM Control

Step 7.

Install monitoring in ADQM Control

Create a cluster

Elena Dvoryadkina, Daria Barysheva

AD.QM CONTROL

Contents

To Table of Contents

Step 1. Get the Enterprise version of a cluster bundle

Step 2. Upload a cluster bundle to ADCM

Step 3. Create a cluster based on the uploaded bundle

In ADCM, a **cluster** means a set of services running on a set of hosts.

Clusters distributions for ADCM are made in the form of so-called bundles. Generally a bundle is a regular archive that includes a description and software logic of cluster deployment and exploitation.

The steps for adding a cluster to ADCM are listed below.

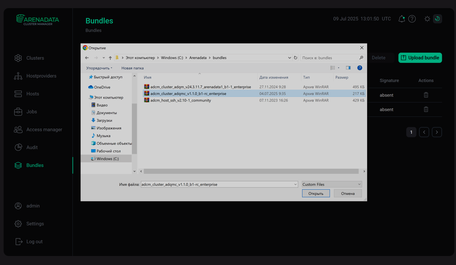
Step 1. Get the Enterprise version of a cluster bundle

Please get a cluster bundle from the Arenadata support team.

Step 2. Upload a cluster bundle to ADCM

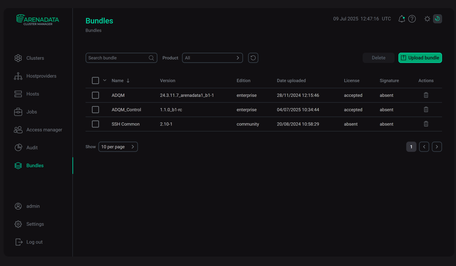
To upload a bundle to ADCM, follow the steps:

1. Select the ***Bundles*** item in the left navigation menu and click ***Upload bundle***.
2. Select a bundle in the Open File Dialog.



Upload a bundle

3. As a result of the performed actions, a bundle is displayed on the ***Bundles*** page.



The result of successful uploading a bundle

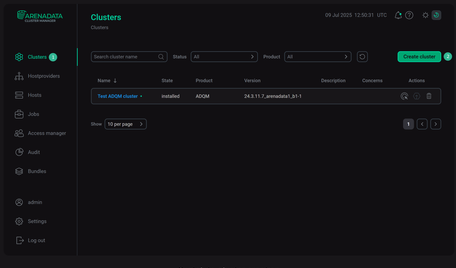
Step 3. Create a cluster based on the uploaded bundle

IMPORTANT

When creating a cluster via the ADCM web interface, you generate a new ADCM cluster instance. It means only adding the cluster to ADCM — at this stage no services are deployed.

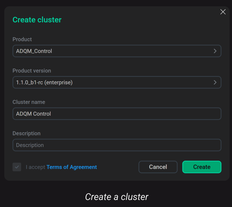
The steps for cluster creation are given below:

1. Select the ***Clusters*** item in the left navigation menu and click ***Create cluster***.



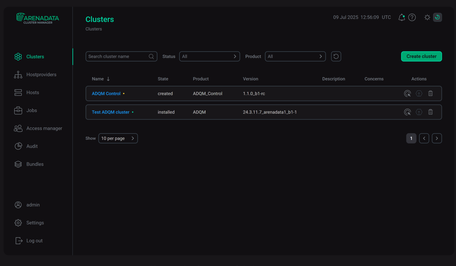
Switch to cluster creation

2. In the opened dialog, describe a new cluster instance:
   - Select an uploaded cluster bundle in the ***Product*** field.
   - Select a bundle version in the ***Product version*** field. Several versions become available in the case of different versions of the same bundle being uploaded.
   - Enter a cluster name in the ***Cluster name*** field.
   - Enter a cluster description in the ***Description*** field if necessary.
   - Set the ***I accept Terms of Agreement*** flag to sign the license agreement. This flag is displayed only for Enterprise bundles. To read the agreement text, refer to the ***Terms of Agreement*** link. You can also sign the license agreement on the ***Bundles*** page by clicking the desired product bundle.
   - Click ***Create***.



Create a cluster

3. As a result, the created cluster instance is displayed on the ***Clusters*** page.



The result of successful cluster creation

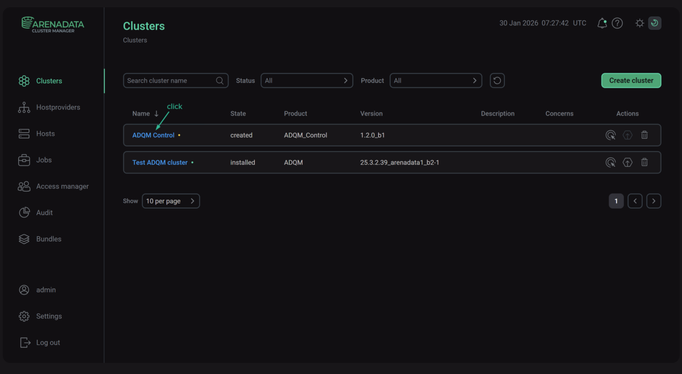
Add services

Elena Dvoryadkina, Daria Barysheva

To Table of Contents

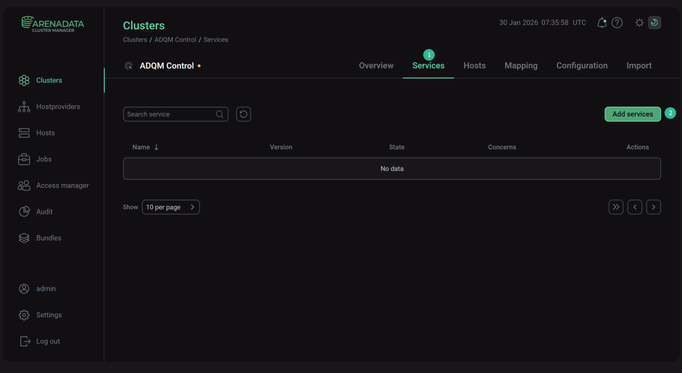
In ADCM, a **service** means a software that performs some function. The steps for adding services to a cluster are listed below:

1. Select a cluster on the ***Clusters*** page. To do this, click a cluster name in the ***Name*** column.



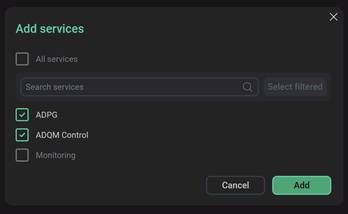
Select a cluster

2. Open the ***Services*** tab on the cluster page and click ***Add services*** (***Add service*** in first versions of ADCM).



Switch to adding services

3. In the opened dialog, select services that should be added to the cluster and click ***Add***.



Select services

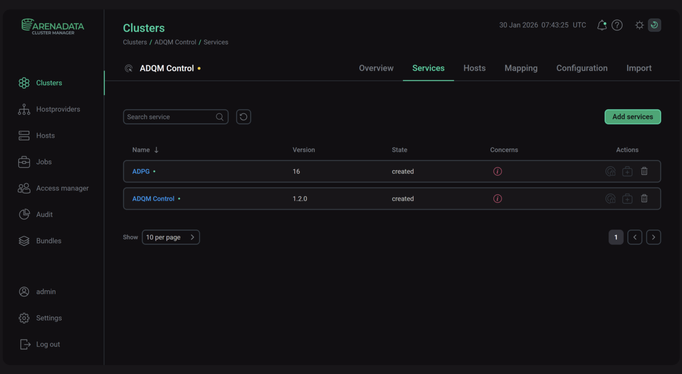
The brief description of available services is listed below.

Services that can be added to the ADQM Control cluster

Service	Mandatory	Purpose
ADQM Control	Yes	Main service that provides the full functionality of the ADQM Control product
ADPG	No	Arenadata Postgres database to store ADQM Control-specific data
Monitoring	No	Service for collecting and storing ADQM Control cluster monitoring metrics based on the Prometheus monitoring system and Grafana service

For an ADQM Control cluster, it is required to select a PostgreSQL database to store ADQM Control data. To do this, you can add the ADPG service to the ADQM Control cluster or specify an external PostgreSQL server through the corresponding settings of the ADQM Control service.

4. As a result, the added services are displayed on the ***Services*** tab.



The result of successful adding services to a cluster

Add hosts to a cluster

Elena Dvoryadkina, Daria Barysheva

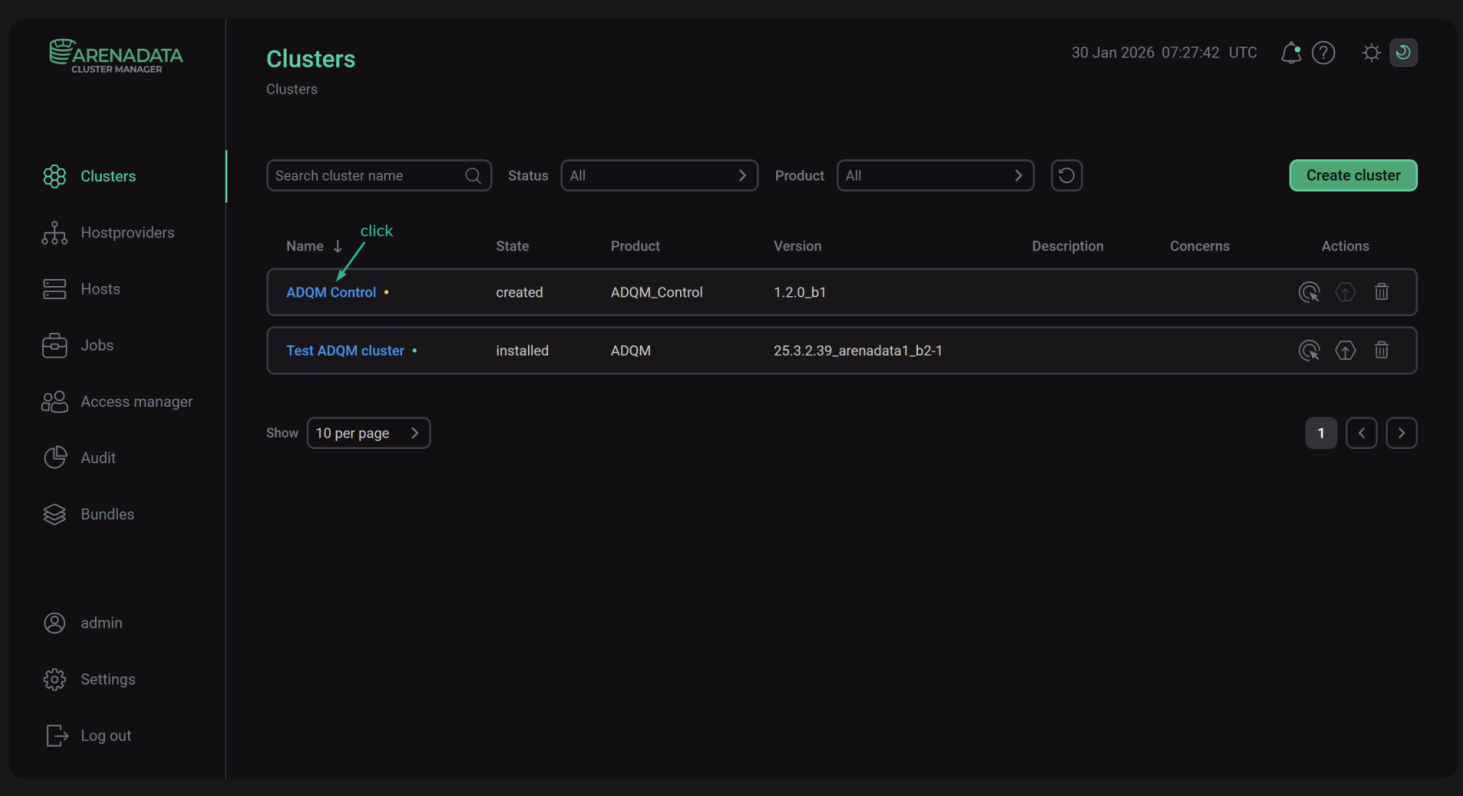
To Table of Contents

NOTE

To set up and install an ADQM Control cluster via ADCM, you should first prepare hosts for the cluster and link these hosts to ADCM. For a detailed description of how to do this, see Prepare hosts in the ADQM documentation.

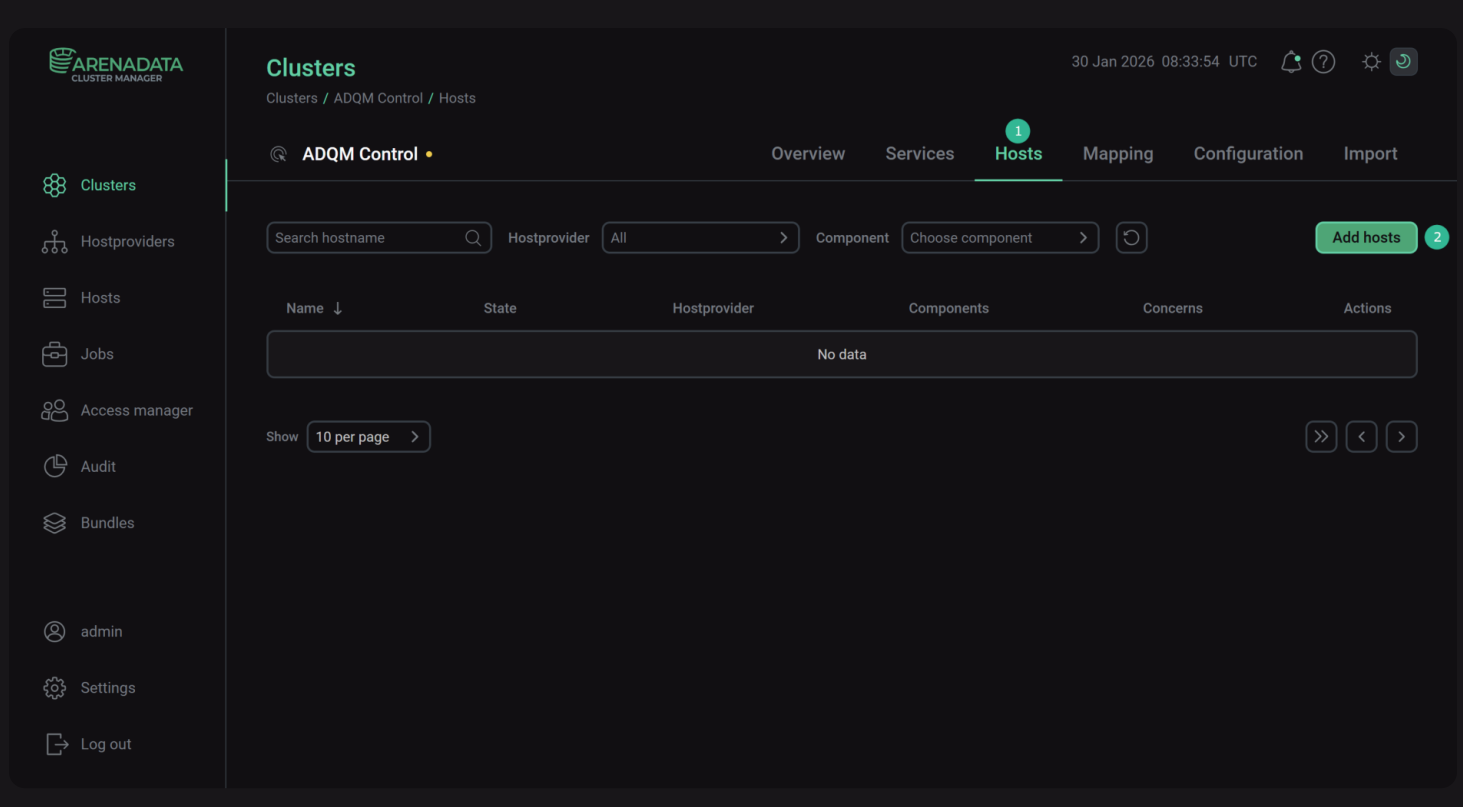
The steps for adding hosts to a cluster are listed below:

1. Select a cluster on the ***Clusters*** page. To do this, click a cluster name in the ***Name*** column.



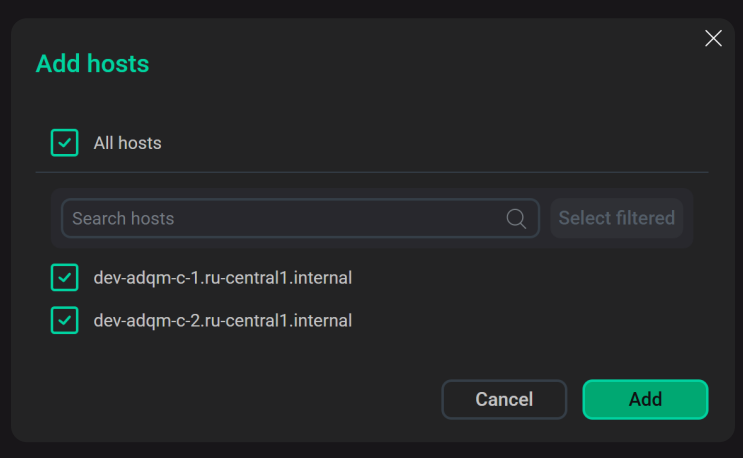
Select a cluster

2. Open the ***Hosts*** tab on the cluster page and click ***Add hosts***.



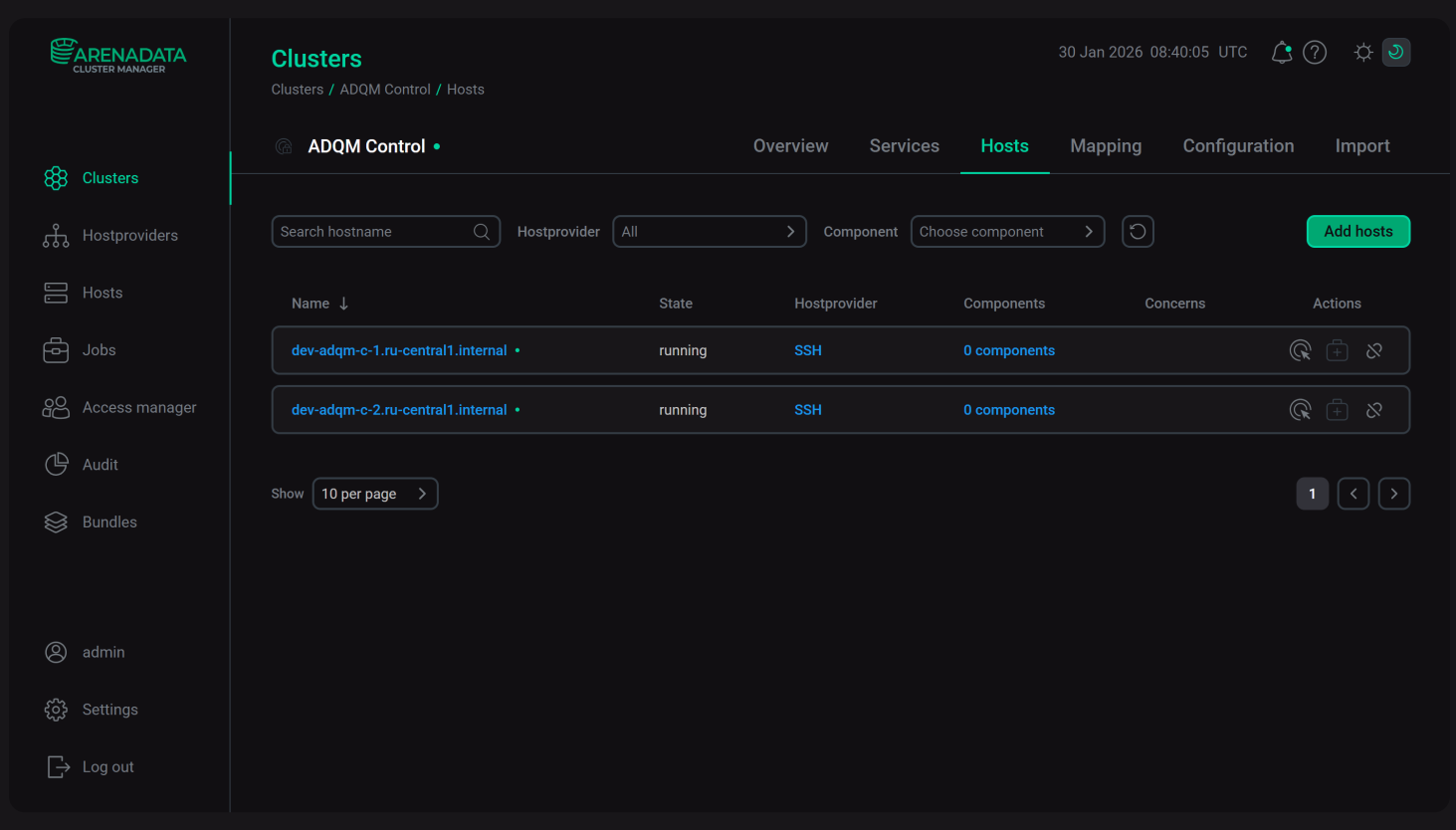
Switch to adding hosts

3. In the opened dialog, select hosts that should be added to the cluster and click ***Add***.



Select hosts

4. As a result, the added hosts are displayed on the ***Hosts*** tab.



The result of successful adding hosts to a cluster

Add components

Elena Dvoryadkina, Daria Barysheva

ADQM CONTROL

To Table of Contents

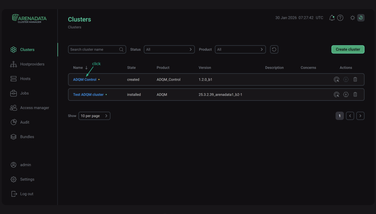
In ADCM, a **component** means a part of a service that should be deployed on one or several cluster hosts.

IMPORTANT

Initially, there are no components on any of cluster hosts. You should map components to hosts manually.

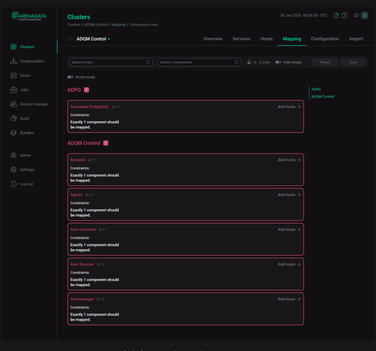
The steps for adding components to hosts are listed below:

1. Select a cluster on the ***Clusters*** page. To do this, click a cluster name in the ***Name*** column.



Select a cluster

2. Open the ***Mapping*** tab on the cluster page.



Switch to mapping service components

All components of ADQM Control services are mandatory (highlighted in red). The ADCM UI also displays how many components should be installed.

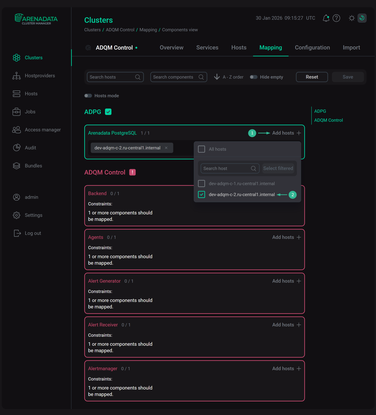
CAUTION

Without assigning a necessary number of hosts to mandatory components, the component mapping cannot be saved.

Components of ADQM Control services

Service	Component	Notes
ADQM Control	Backend	Implements the product's web interface
	Agents	Collects information on states of hosts and metrics from ADQM clusters
	Alert Generator	Generates alerts and sends them to the alert manager
	Alert Receiver	Receives alerts from the alert manager and writes them to the data storage
	Alertmanager	Handles alerts (mutes alerts of some type, filters alerts, adds additional labels) and resends processed alerts to the alert receiver
ADPG	Arenadata PostgreSQL	Data storage

3. In the window that opens, map components to hosts. To assign hosts for each component, click ***Add hosts*** and select hosts in the pop-up window.

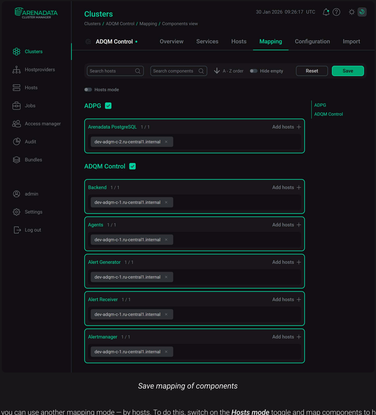


Select hosts for a component

TIP

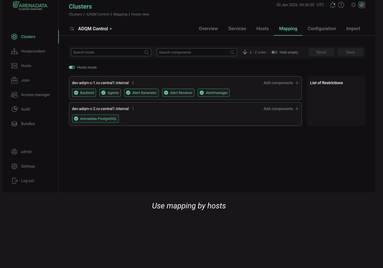
It is recommended to install the Arenadata PostgreSQL component on a dedicated host, separately from the components of the ADQM Control service.

4. After component distribution is completed, click ***Save***.



Save mapping of components

Starting with ADCM 2.2.0, you can use another mapping mode — by hosts. To do this, switch on the ***Hosts mode*** toggle and map components to hosts by clicking ***Add components*** for each host sequentially.

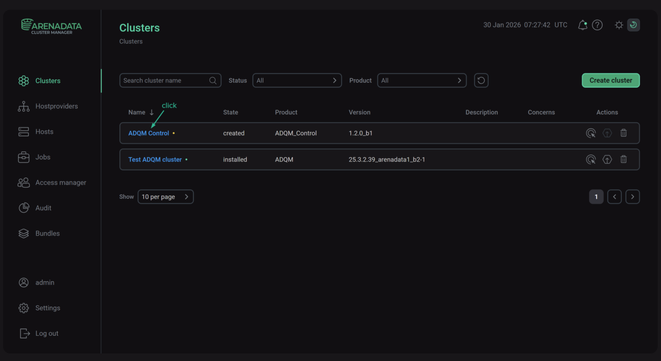


Use mapping by hosts

Configure services

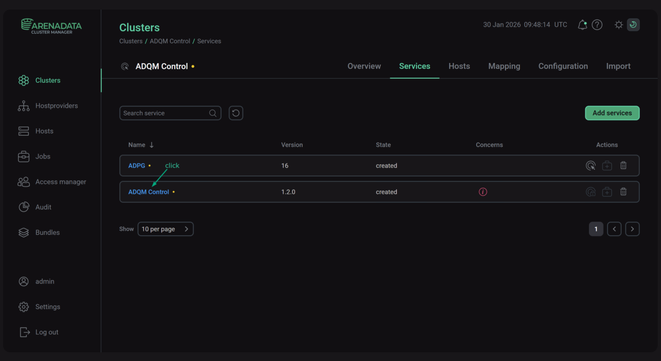
Elena Dvoryadkina, Daria Barysheva

1. Select a cluster on the ***Clusters*** page. To do this, click a cluster name in the ***Name*** column.



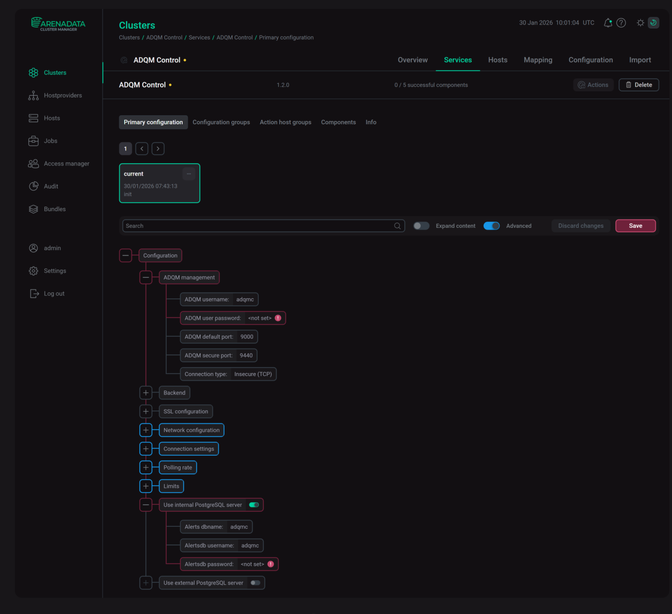
Select a cluster

2. Open the ***Services*** tab on the cluster page, find the row that contains the desired service, and click its name in the ***Name*** column.



Switch to configuring a service

3. In the window that opens:
   - Open the ***Primary Configuration*** tab.
   - Switch on the ***Show advanced*** toggle.
   - Fill in configuration parameters of the selected service. The fields highlighted in red are mandatory.
   - Click ***Save***.



Configure a service

ADQM user credentials

In the ***ADQM username*** and ***ADQM user password*** parameters of the ADQM Control service, specify the name and password of an ADQM user to be used by ADQM Control agents for connecting to ADQM. For this, you can use the `default` ADQM user account with an assigned password or create a separate user account in ADQM (recommended), for example:

```
CREATE USER adqmc ON CLUSTER default_cluster IDENTIFIED WITH sha256_password BY 'qwerty123';
```

```
GRANT ON CLUSTER default_cluster SELECT ON *.* TO adqmc;
```

For more information about all configuration parameters of ADQM Control services, see Configuration parameters.

Configure a cluster

Elena Dvoryadkina, Daria Barysheva

Contents

To Table of Contents

Primary configuration

Configuration groups

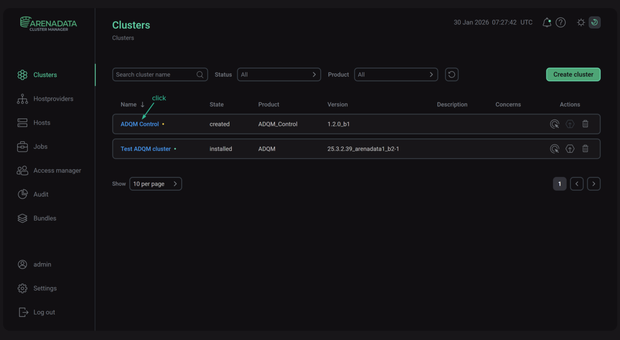
Ansible settings

IMPORTANT

There is usually no need to change cluster configuration parameters. You can leave all parameters at the default values.

After creation a new cluster, you can configure it by performing the following steps:

1. Select a cluster on the ***Clusters*** page. To do this, click a cluster name in the ***Name*** column.



Select a cluster

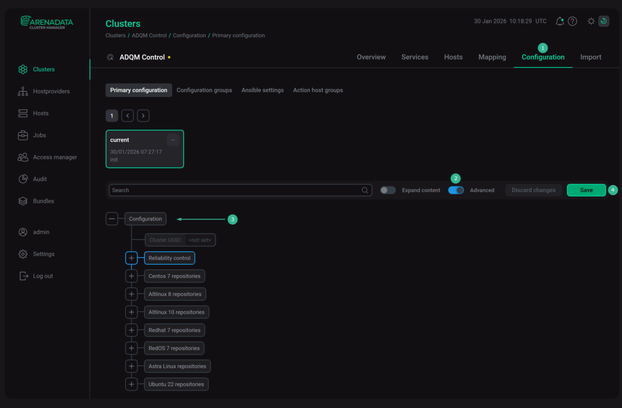
2. Open the ***Configuration*** tab on the cluster page. The ***Configuration*** tab includes the following sections: ***Primary configuration***, ***Configuration groups***, ***Ansible settings***.
3. Fill in all necessary parameters on the selected tab and click ***Save***.

Primary configuration

The ***Configuration*** → ***Primary configuration*** tab allows you to configure the general cluster settings.

ADQM Control cluster configuration parameters are listed below:

- ***Reliability control*** — the section with the following settings (visible if the ***Show advanced*** option is enabled):
  - ***retries*** — number of retries for cluster operations (`3` by default);
  - ***delay*** — delay between retries in seconds (`10` by default);
  - ***timeout*** — timeout for checks of network ports in seconds (`60` by default).
- Paths to the repositories that are used during the installation process — depending on the cluster operating system (Centos, Alt Linux, Red Hat, RED OS, Astra Linux, or Ubuntu):
  - ***ADQMC_arenadata_community*** — repository required to install the Community edition of ADQM Control;
  - ***arenadata_postgres*** — repository required to install Arenadata PostgreSQL;
  - ***prometheus_repo*** — repository required to install Prometheus;
  - ***ADQMC_arenadata_enterprise*** — repository required to install the Enterprise edition of ADQM Control.



Cluster configuration window

Configuration groups

The ***Configuration*** → ***Configuration groups*** tab is designed to set cluster configuration groups.

Ansible settings

The ***Configuration*** → ***Ansible settings*** tab allows you to set Ansible configuration options at the cluster level. The tab is available starting with ADCM 2.2.0.

Ansible settings

Name	Description	Default
forks	The number of parallel processes to spawn when communicating with remote hosts	5

NOTE

During offline installation, these parameters are ignored. The data stored locally in the Enterprise Tools cluster is used instead.

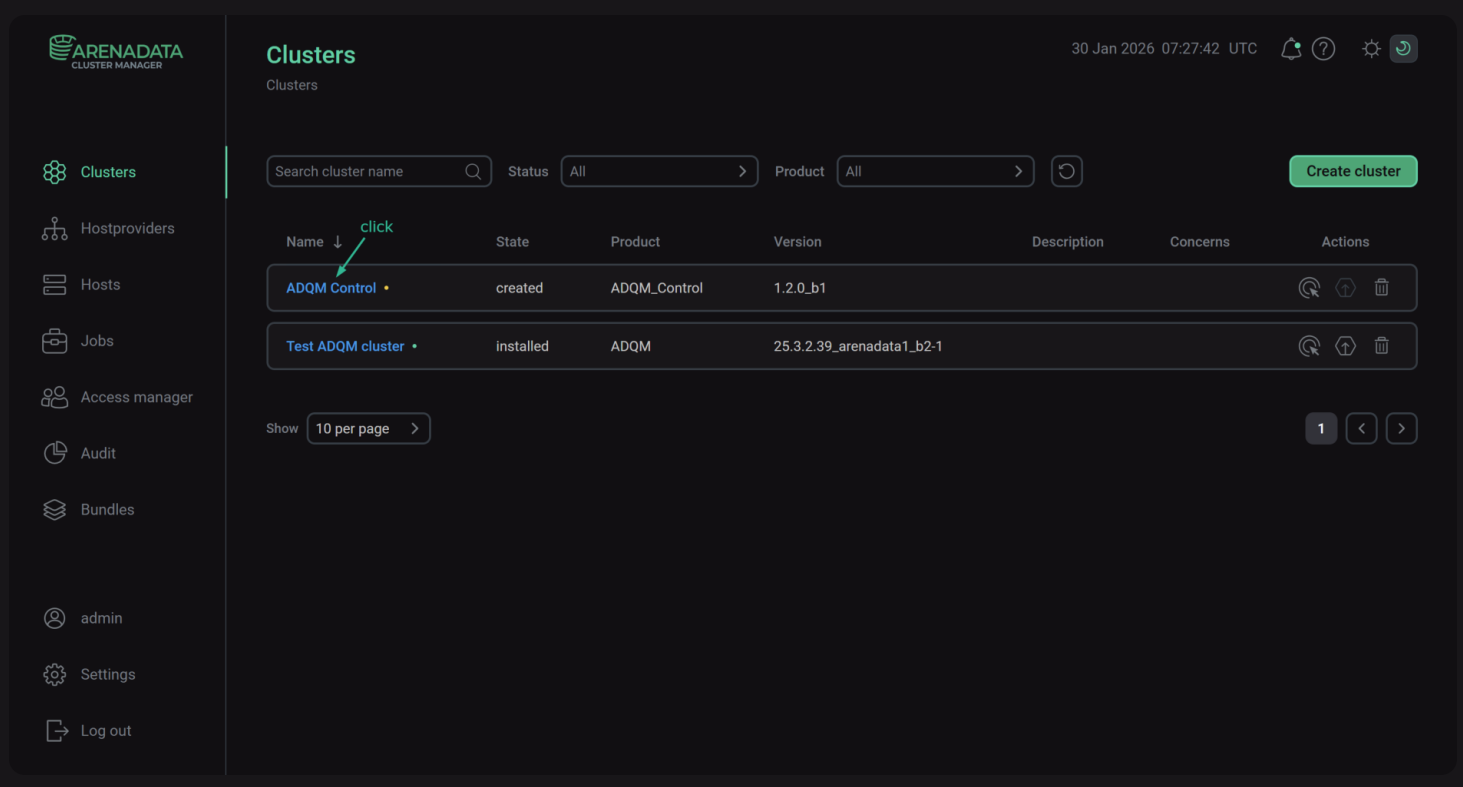
Integrate with an ADQM cluster

Elena Dvoryadkina

To Table of Contents

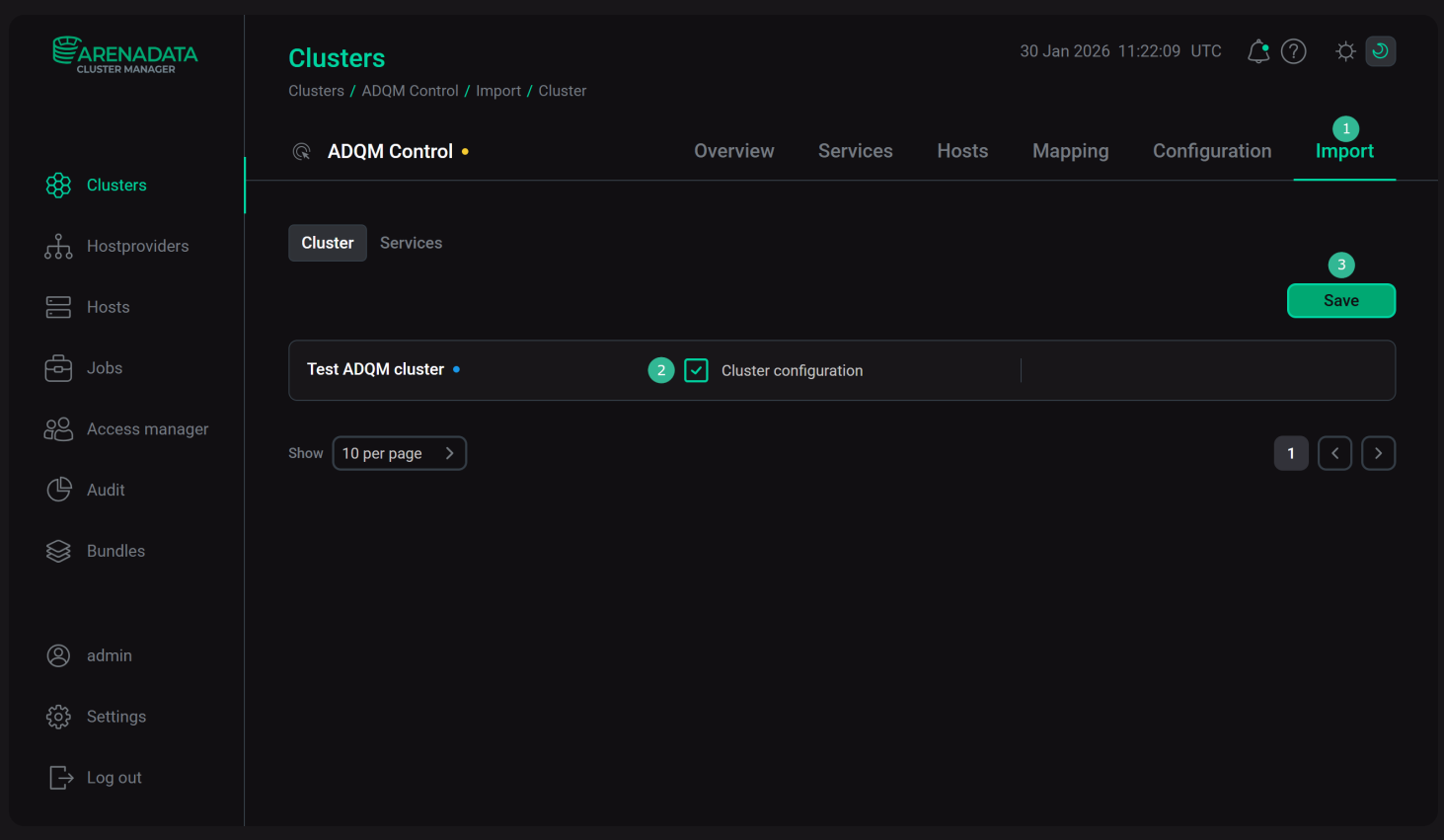
To set up an ADQM Control cluster, you should integrate it with at least one pre-installed ADQM cluster with the Monitoring service configured:

1. On the ***Clusters*** page, click an ADQM Control cluster name in the ***Name*** column.



Select a cluster

2. Open the ***Import*** tab.
3. Select ***Cluster configuration*** for the ADQM cluster to be imported into ADQM Control.
4. Click ***Save***.



Integration with an ADQM cluster

NOTE

A single ADQM Control cluster can also be integrated with multiple ADQM clusters.

Import ET settings

Daria Barysheva, Elena Dvoryadkina

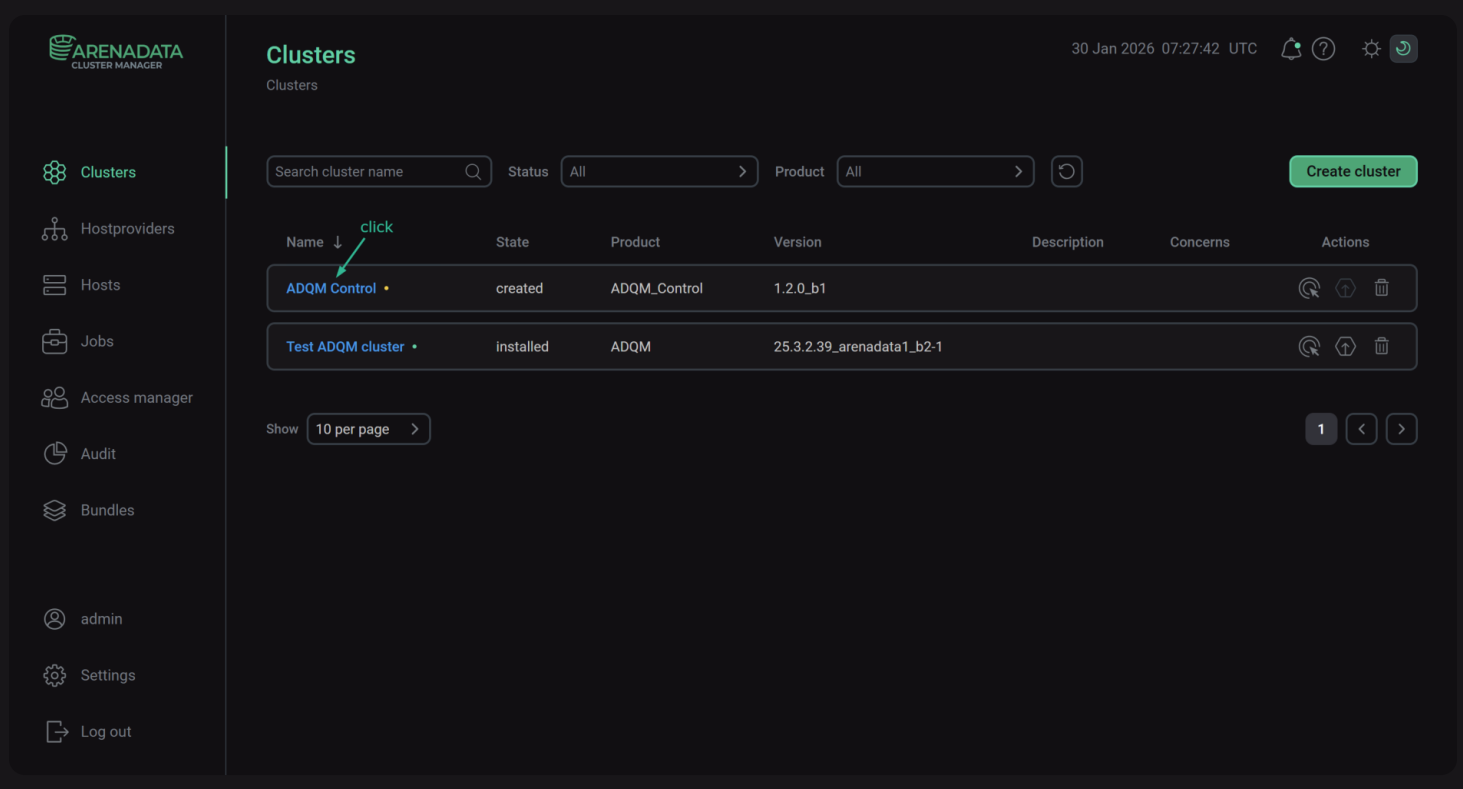
To Table of Contents

IMPORTANT

This step should be performed only in the case of offline installation.

In order to determine that a new cluster should be installed offline, it is necessary to integrate it with the previously created Enterprise Tools (ET) cluster. Follow the steps:

1. Select a cluster on the ***Clusters*** page. To do this, click a cluster name in the ***Name*** column.

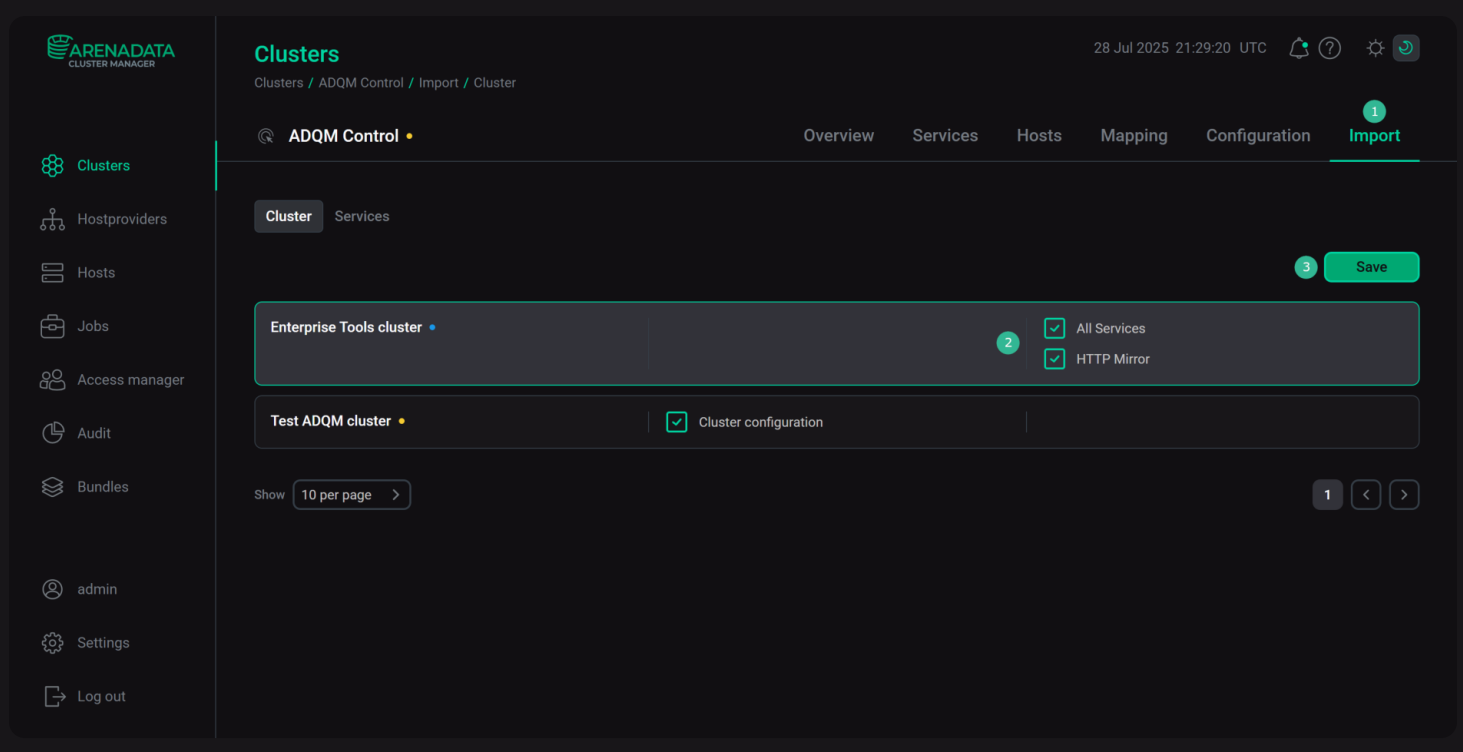


Select a cluster

2. Open the ***Import*** tab on the cluster page.

3. Select the ***HTTP Mirror*** service of the previously created ET cluster.

4. Click ***Save***.



Integration with the ET cluster

NOTE

After successful import, the data stored locally in the Enterprise Tools cluster can be used at the cluster installation stage. So there is no need to extract this data via Internet.

Install a cluster

Elena Dorosyadkina, Elena Barysheva

ADQM CONTROL

Contents

To Table of Contents

Step 1. Run installation
- Whole cluster
- Single services

Step 2. View the results of installation

Step 3. Check the cluster state after installation

Step 4. Check the ADQM Control web interface

Step 1. Run installation

There are two ways to install cluster services:

1. **Whole cluster** - in this method all services are installed automatically one by one after applying the *Install* action to a cluster.
2. **Single services** - In this method each service is installed manually after applying one or more actions to it. This way is also suitable for installing new services in the already deployed cluster.

In both cases each service starts automatically after installation.

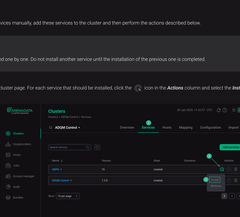
NOTE

In case of errors, you can find logs on the **Jobs** page.

Whole cluster

To install all cluster services within a single action, follow the steps:

1. Apply the *Install* action to the cluster selected on the *Clusters* page by clicking the icon in the **Actions** column.



Switch to the whole cluster installation

2. Wait until the job is completed.

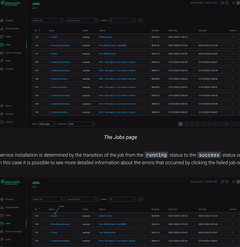
Single services

To install one or several cluster services manually, add these services to the cluster and then perform the actions described below.

CAUTION

Notice that services are installed one by one. Do not install another service until the installation of the previous one is completed.

1. Open the **Services** tab on the cluster page. For each service that should be installed, click the icon in the **Actions** column and select the *Install* action.



Switch to installation of single services

2. Wait until the job is completed. Then check that the service state has changed from `created` to `installed`.
3. Repeat the previous actions for other services that should be installed.

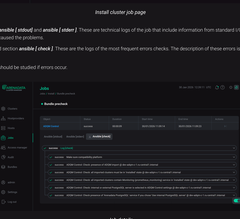
IMPORTANT

When you install ADQM Control services manually, one by one, use the following order:

1. ADPG (if a connection to an external PostgreSQL cluster is not configured)
2. ADQM Control

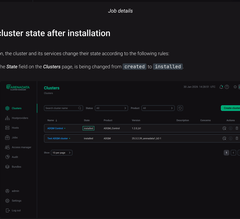
Step 2. View the results of installation

ADCM starts a single job for installation process, as well as for any other task. You can find out about the status of jobs more specifically on the **Jobs** page.

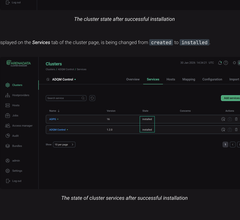


The Jobs page

The successful completion of the service installation is determined by the transition of the job from the `running` status to the `success` status on the **Jobs** page. If the job fails, it switches to the `failed` status. In this case it is possible to see more detailed information about the errors that occurred by clicking the failed job on the **Jobs** page.



Switch to a job page

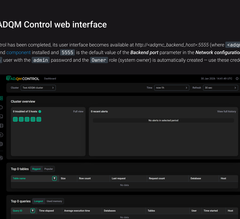


Install cluster job page

A job page contains two sections: ***ansible [stdout]*** and ***ansible [stderr]***. These are technical logs of the job that include information from standard I/O streams stdout/stderr. These logs can help to understand what caused the problems.

There can also be the optional third section ***ansible [check]***. These are the logs of the most frequent errors checks. The description of these errors is more simple and specific, than in two previous technical logs.

The contents of all three sections should be studied if errors occur.

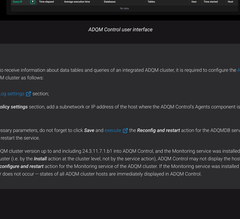


Job details

Step 3. Check the cluster state after installation

As a result of successful installation, the cluster and its services change their state according to the following rules:

- The cluster state, displayed in the *State* field on the **Clusters** page, is being changed from `created` to `installed`.



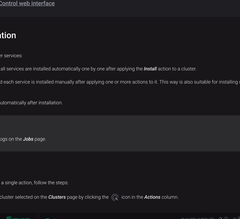
The cluster state after successful installation

- The state of cluster services, displayed on the **Services** tab of the cluster page, is being changed from `created` to `installed`.

The state of cluster services after successful installation

Step 4. Check the ADQM Control web interface

After the installation of ADQM Control has been completed, its user interface becomes available at http://<adqmc_backend_host>:5555 (where `<adqmc_backend_host>` is the IP address of the host with the Backend component installed and `5555` is the default value of the *Backend port* parameter in the *Network configuration* section of the ADQM Control service configuration). The `admin` user with the `admin` password and the `Senior` role (system owner) is automatically created - use these credentials to log in to ADQM Control.



ADQM Control user interface

IMPORTANT

- In order for ADQM Control to receive information about data tables and queries of an integrated ADQM cluster, it is required to configure the ADQMDB service parameters for the ADQM cluster as follows:
  - enable ***query_log*** in the Log settings section;
  - in the ***Default user and policy settings*** section, add a subnetwork or IP address of the host where the ADQM Control's Agents component is installed to the ***Default user IP list***.

  After you have set the necessary parameters, do not forget to click *Save* and execute the ***Reconfig and restart*** action for the ADQMDB service to save configuration changes and restart the service.
- If you have imported an ADQM cluster version up to and including 24.3.1.17.1.b1 into ADCM, and the Monitoring service was installed during the installation of the ADQM cluster (i.e. by the *Install* action at the cluster level, not by the service action), ADQM Control may not display the hosts of that cluster. To fix this issue, run the ***Reconfigure and restart*** action for the Monitoring service of the ADQM cluster. If the Monitoring service was installed into an existing ADQM cluster, this behavior does not occur – states of all ADQM cluster hosts are immediately displayed in ADQM Control.

Install monitoring

Table of contents

Contents

Overview
Step 1. Add the service
Step 2. Add components
Step 3. Configure the service
Step 4. Install the service
Step 5. View results
- Metrics in the Prometheus format
- Prometheus web interface
- Grafana web interface

The Monitoring service deploys the Prometheus server inside ADQM Control to collect and store ADQM Control cluster monitoring metrics, and also supports the ability to use the Grafana UI web application for visualization and analysis of information. This article describes the steps required to add this service to an ADQM Control cluster.

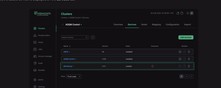
Overview

When adding the Monitoring service to ADQM Control, you can install the node exporter on all or selected hosts of your cluster. It is a monitoring agent that exports host system metrics. Prometheus can then collect these metrics from the ADQM Control server and from the node exporter. The collected metrics are then available in the Grafana web application, specified in the Grafana configuration. You can also use the Prometheus and Grafana web interfaces to view and analyze data that the monitoring service collects.

> **NOTE**
>
> - If you already have a Prometheus monitoring system set up (for example, your own Prometheus server or VictoriaMetrics), you can use it to collect ADQM Control metrics. To do this, configure access to ADQM Control metrics in your monitoring system using ADQM Control's Prometheus parameters specified on the Monitoring service page.
> - You can also use the VictoriaMetrics implementation to integrate all metrics from the Prometheus server deployed in ADQM Control into VictoriaMetrics.

Step 1. Add the service

1. In the ADCM interface, open the **Clusters** page and click your ADQM Control cluster name. On the cluster page that opens, switch to the **Services** tab and click **Add services**.



Switch to adding services

2. In the opened dialog, select the **Monitoring** service and click **Add**.



Select the service

As a result, the added service is displayed on the **Services** tab.



The result of successfully adding the service to the cluster

Step 2. Add components

1. On the cluster page, open the **Mapping** tab to proceed to mapping service components to cluster hosts.



Switch to mapping service components

Monitoring service components

Component	Description
Prometheus Server	Stores and allows requesting all metrics for an ADQM Control cluster
Grafana	Visualizes ADQM Control metrics as graphs and charts organized into dashboards
Node Exporter	Collects system metrics and sends them to Prometheus

2. Assign hosts to components of the Monitoring service – click **Add hosts** and select the desired hosts in the pop-up window.



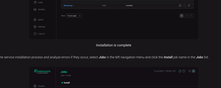
Select a host for a component

> **CAUTION**
>
> It is not recommended to install the Prometheus Server component on hosts with ADQM Control – can separate hosts for it. Otherwise, if an ADQM Control host fails or the load on it is critically high, information about the corresponding problems will not be saved.

3. After the distribution of components is completed, click **Save**.

Step 3. Configure the service

1. Open the **Services** tab on the cluster page and click the **Monitoring** service name in the **Name** column.



Switch to the service configuration

2. On the page that opens, fill in the service configuration parameters.



Configure the Monitoring service

Fields highlighted in red are required.

- **Prometheus settings** → **Password for Grafana connection** – a password of a Grafana user (`admin` or `grafana`) that is used to connect to Prometheus. The password should contain a combination of numbers, symbols, lowercase and uppercase letters and have a minimum length of 8.
- **Grafana settings** → **Grafana administrator's password** – a password of the `admin` Grafana user. The password should contain a combination of numbers, symbols, lowercase and uppercase letters and have a minimum length of 8.

All monitoring parameters are described in the **Monitoring** section of the Configuration parameters article.

3. After specifying all necessary parameters, click **Save**.

Step 4. Install the service

1. On the **Services** tab, click the icon in the **Actions** column and run the **Install** action.



Switch to the service installation

2. Wait until the installation is completed. Then check that the service state has changed from `created` → `installed`.

Installation is complete

To view the service installation process and analyze errors if they occur, select **Jobs** in the left navigation menu and click the **Install** job name in the **Jobs** list.



Install service job page

Step 5. View results

The Monitoring service starts automatically after installation. To ensure that monitoring works correctly, check that both system metrics and metrics of ADQM Control's services are collected in Prometheus – see Metrics in the Prometheus format. To do this, view metrics of the Prometheus format in the browser or use the Prometheus or Grafana web interface.

Metrics in the Prometheus format

1. In the address bar of your browser, enter an address of a host on which Clickhouse is deployed and add a port number + a value of the **Grafana listen port** parameter located in the **Grafana settings** section of the Monitoring service configuration (the default value is `3000`). For example, `http://135.181.205.13:3000`. To log in, use `admin` as a username, and the **Grafana administrator's password** parameter value (found in the **Grafana settings** section of service configuration parameters) as a password.
2. In the window that opens, navigate to **Home** → **Dashboards** and expand the **ADQM Dashboard ADQM_Control_cluster_name** section. In this section, you can select one of the available dashboards to view service's system metrics and metrics from your ADQM Control cluster.

View ADQM Control metrics in Grafana

Prometheus web interface

1. In the address bar of your browser, enter an address of a host on which Clickhouse is deployed and add a port number + a value of the **Prometheus web port** parameter in the **Service parameters** section of the Monitoring service configuration (the default value is `19090`). For example, `http://135.181.205.13:19090`. To log in, use the Prometheus interface, use a username and password that are also specified in the **Prometheus settings** section – the **Prometheus users in login:hash or Prometheus** setting.
2. In the **Expression** field, you can enter a metric and click **Execute** – values of this metric on all hosts of the ADQM Control cluster will be shown in the interface.

Prometheus web interface

Grafana web interface

1. In the address bar of your browser, enter an address of a host on which Grafana is deployed and add a port number + a value of the **Grafana listen port** parameter located in the **Grafana settings** section of the Monitoring service configuration (the default value is `3000`). For example, `http://135.181.205.13:3000`. To log in, use `admin` as a username, and the **Grafana administrator's password** parameter value (found in the **Grafana settings** section of service configuration parameters) as a password.
2. In the window that opens, navigate to **Home** → **Dashboards** and expand the **ADQM Dashboard ADQM_Control_cluster_name** section. In this section, you can select one of the available dashboards to view service's system metrics and metrics from your ADQM Control cluster.

View ADQM Control metrics in Grafana

Connect to ADQM Control

Elena Dvoryadkina

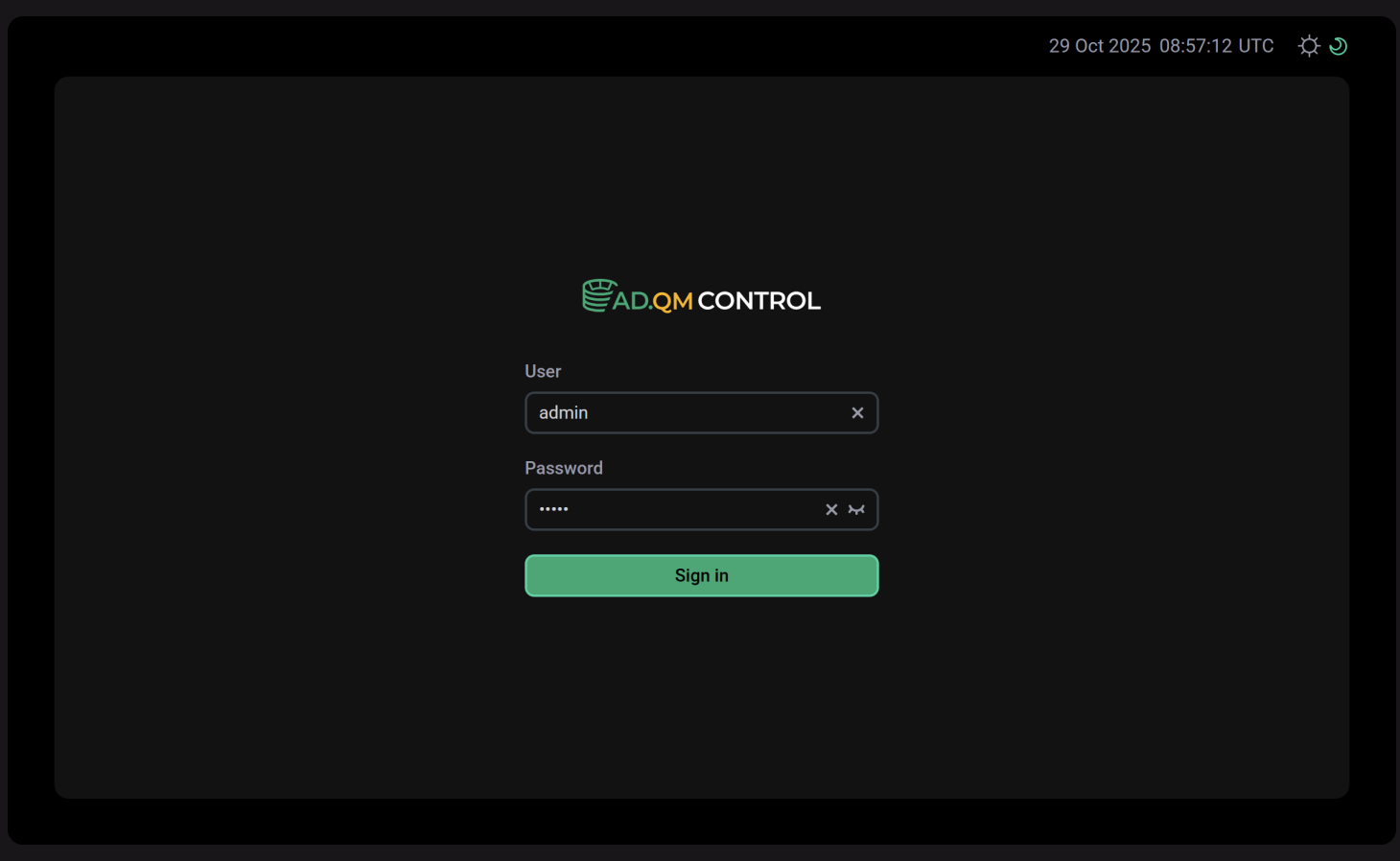
To Table of Contents

To connect to the ADQM Control web interface, follow the steps:

1. In the browser bar, enter the IP address of the host where the Backend component is installed. To specify the port number, use the value of the ***Backend port*** from the ***Network configuration*** section of the ADQM Control service configuration (`5555` by default).

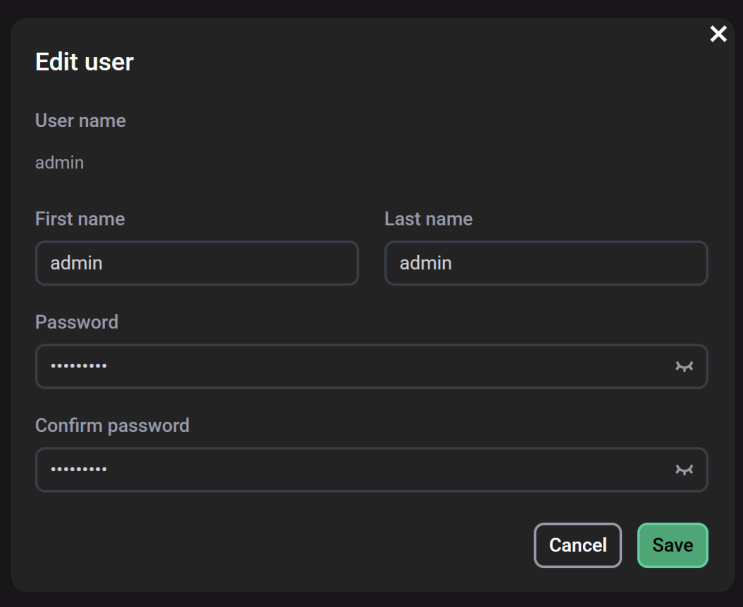
2. To log in for the first time after installation, use the default user credentials:

   - ***User:*** admin
   - ***Password:*** admin



Log in to ADQM Control

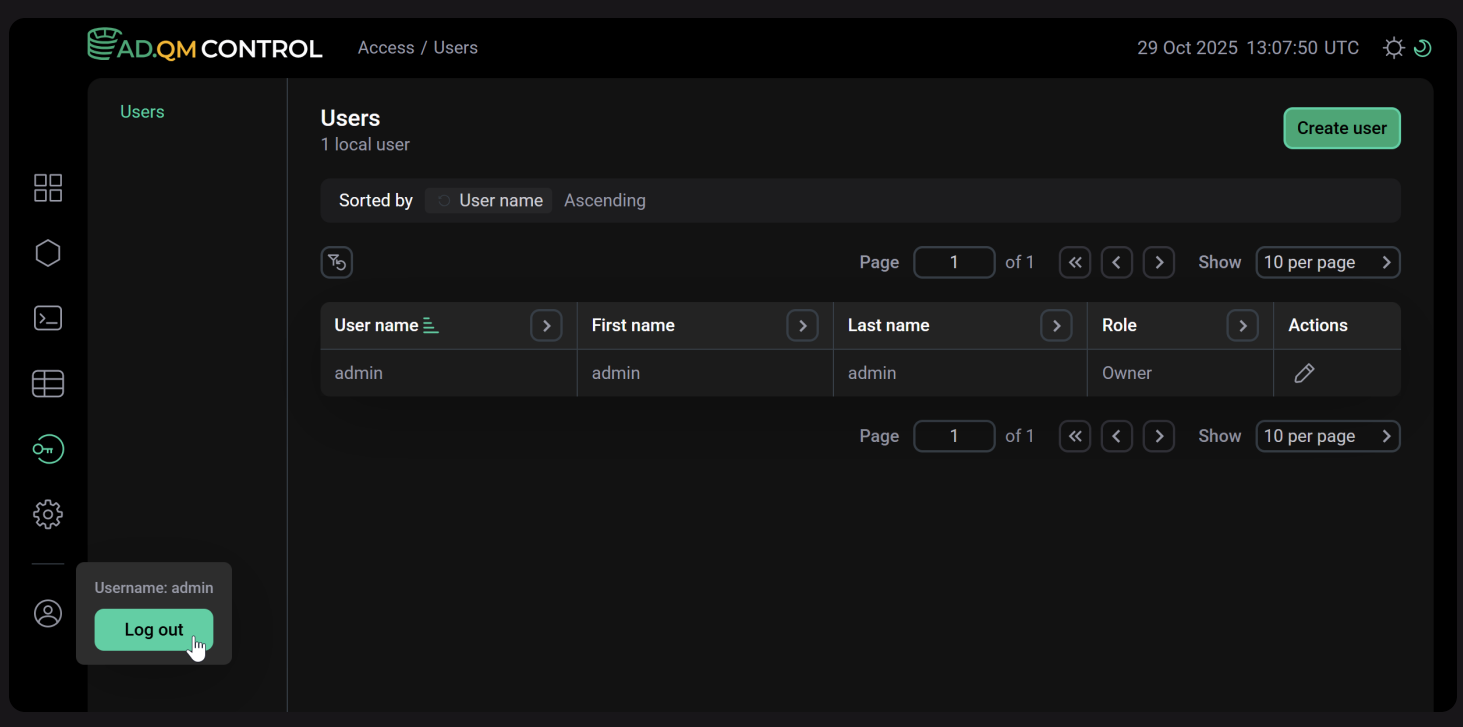
3. It is recommended to assign a new password to the `admin` user instead of the default one. To do this, go to the ***Access*** → ***Users*** page, open the edit window for the `admin` user, set a new password in the ***Password*** and ***Confirm password*** fields, and click ***Save***.



Change a password

On the ***Access*** → ***Users*** page, you can also add new ADQM control users — see the Manage users article for details.

4. To log out from ADQM Control, hover the mouse over the bottom icon in the left navigation menu and click ***Log out*** in the pop-up window (it also displays the current user name).



Log out from ADQM Control

Manage users

Elena Dvoryadkina

Contents

To Table of Contents

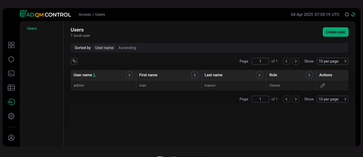
View a list of users
Add a user
Edit a user
Set password requirements

Users log in to ADQM Control under the accounts previously created for them in the system. To manage users, you can use the ***Users*** tab on the ***Access*** page in the ADQM Control web interface.

After an ADQM Control cluster has been installed, the `admin` user with the `admin` password and the `Owner` role (system owner) is automatically created.

View a list of users

The list of ADQM Control users is available on the ***Access*** → ***Users*** page. For each user, it displays the login, first name, last name, and role.



The Users page

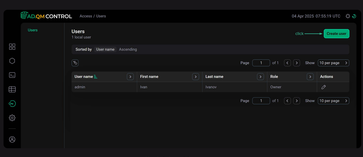
In column headers of the table with a list of users, there are filters that you can use to select specific data. To open a filter, click the › icon.

The ▼ icon means that a filter is defined for a column. To reset all filters, click ⟲.

Add a user

To add a new user, follow the steps:

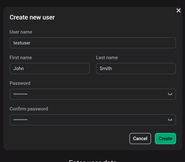
1. On the ***Access*** → ***Users*** page, click ***Create user***.



Switch to adding a user

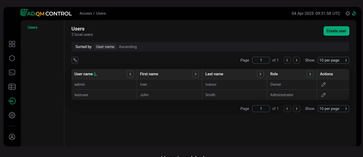
2. In the window that opens, fill in the following fields:
   - ***User name*** – user's login;
   - ***First name*** – user's name;
   - ***Last name*** – user's last name;
   - ***Password*** – user's password.

> **TIP**
>
> By default, the ***Password*** field shows the hint with password requirements (configured on the ***Settings*** → ***Password rules*** page) when you start filling it. To get this information, you can also hover the mouse over the ⓘ icon, which appears to the right of the ***Password*** field name on hover.



Enter user data

3. Click ***Create***. As a result, a new user (with the `Administrator` role – system administrator) is displayed on the ***Access*** → ***Users*** page.

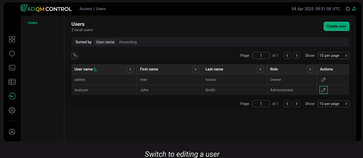


User is added

Edit a user

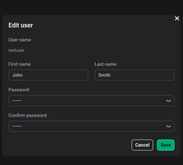
To edit user data, follow the steps:

1. Click the ✎ icon in the ***Actions*** column on the ***Access*** → ***Users*** page.



Switch to editing a user

2. In the window that opens, edit necessary fields. For example, you can specify another password for the user (taking into account the specified password complexity requirements). The user's login in the ***User name*** field cannot be changed.



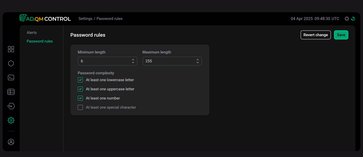
Edit user data

> **NOTE**
>
> Administrators can change only their own passwords. Owners can change passwords for any users.

3. Click ***Save*** to apply changes.

Set password requirements

On the ***Settings*** → ***Password rules*** page, you can set requirements for the complexity of user passwords.



Password rules

If you need to edit the default settings, change the necessary parameters and click ***Save***. Click ***Revert change*** if you need to undo the changes that have not been yet saved by clicking ***Save***.

Parameters to set up password rules

Parameter	Description	Default value
Minimum length	Minimum length of a password	1
Maximum length	Maximum length of a password	255
At least one lowercase letter	Indicates whether a password should contain at least one lowercase character	Disabled
At least one uppercase letter	Indicates whether a password should contain at least one uppercase character	Disabled
At least one number	Indicates whether a password should contain at least one numeric character	Disabled
At least one special character	Indicates whether a password should contain at least one special character (excluding characters from the extended ASCII table)	Disabled

SSL encryption

Table of contents

Create SSL certificates

Configure ADOM Control to enable SSL encryption

- Manage SSL cluster action
- SSL parameters in service configurations
- Example

Test a connection

Disable SSL

To Table of Contents

For secure connections to ADOM clusters and connections of data exchange between servers components, ADOM Control supports SSL encryption. To use this feature in your ADOM Control installation, [activate the SSL support](#).

The example in this article uses an ADOM Control cluster that includes two nodes, on which [components of ADOM Control services](#) are distributed as follows:

- ADOM 1 URL with subject=/C=US/emailAddress=*.adomcontrol.com/CN=*.adomcontrol.com
- ADOM 1 URL with subject=/C=US/emailAddress=*.adomcontrol.com/CN=*.adomcontrol.com

ADOM Control components on the ADOM cluster run under the SSL encryption of connections is enabled according to the tutorial from the [Configure ADOM Control to enable SSL encryption](#) section.

For the environment of the test connections, we will use the [ADOM Control Test Client](#) application (it contains a demo version of the client) and the [ADOM Control Test Server](#) application (it contains a demo version of the server). Both applications are available in the [ADOM Control SDK](#).

Create SSL certificates

To create and register SSL certificates with the command-line tool `openssl`, utility command version 1.1.1:

1. For each node of the ADOM Control cluster, generate a key and create a self-signing request (CSR):

```
$ openssl req -newkey rsa:4096 -nodes -keyout
> -subj "/CN=Your Name/C=US/emailAddress=*.adomcontrol.com"
> -addext "subjectAltName = DNS:*.adomcontrol.com, DNS:*.adomcontrol.com, IP:10.0.0.1, IP:10.0.0.2"
> -addext "extendedKeyUsage = clientAuth, serverAuth"
> -addext "keyUsage = digitalSignature, keyEncipherment"
> -out adomcontrol.com.csr
$ openssl req -newkey rsa:4096 -nodes -keyout
> -subj "/CN=Your Name/C=US/emailAddress=*.adomcontrol.com"
> -addext "subjectAltName = DNS:*.adomcontrol.com, DNS:*.adomcontrol.com, IP:10.0.0.3, IP:10.0.0.4"
> -addext "extendedKeyUsage = clientAuth, serverAuth"
> -addext "keyUsage = digitalSignature, keyEncipherment"
> -out adomcontrol.com.csr
```

The first command creates a private key and a self-signing request with CSRs of a root key, and the second one is similar, but creates a key and a request with CSRs of a child key. Each command creates two files: `adomcontrol.com.key` and `adomcontrol.com.csr`.

2. For each node, create and sign a certificate using the CA root key and private key:

```
$ openssl x509 -req \
> -in adomcontrol.com.csr \
> -CA adomcontrol.com.crt \
> -CAkey adomcontrol.com.key \
> -CAcreateserial -out adomcontrol.com.crt -days 365
Output:
Certificate request self-signature ok
subject=C = US, emailAddress = *.adomcontrol.com, CN = Your Name
$ openssl x509 -req \
> -in adomcontrol.com.csr \
> -CA adomcontrol.com.crt \
> -CAkey adomcontrol.com.key \
> -CAcreateserial -out adomcontrol.com.crt -days 365
Output:
Certificate request self-signature ok
subject=C = US, emailAddress = *.adomcontrol.com, CN = Your Name
```

As a result, the `adomcontrol.com.crt` file is created.

3. For the Windows components of the ADOM Control cluster, copy the files to the folder with the certificates:

- `adomcontrol.com.crt` to `%ProgramData%\Server\`
- `adomcontrol.com.key` to `%ProgramData%\Server\`
- `adomcontrol.com.crt` to `%ProgramData%\Client\`
- `adomcontrol.com.key` to `%ProgramData%\Client\`

4. For the Windows components of the ADOM Control cluster, copy files to the folder with the certificates:

- `adomcontrol.com.crt` to `/etc/adomcontrol-manager/server.crt`
- `adomcontrol.com.key` to `/etc/adomcontrol-manager/server.key`
- `adomcontrol.com.crt` to `/etc/adomcontrol-manager/client.crt`
- `adomcontrol.com.key` to `/etc/adomcontrol-manager/client.key`

5. For the ADOM Control cluster, copy files to the folder of [Linux/Mono] tool:

- `adomcontrol.com.crt` to `/etc/adomcontrol-tool/server.crt`
- `adomcontrol.com.key` to `/etc/adomcontrol-tool/server.key`
- `adomcontrol.com.crt` to `/etc/adomcontrol-tool/client.crt`
- `adomcontrol.com.key` to `/etc/adomcontrol-tool/client.key`

For the test client, we will perform each manager's encryption for the connection and the server of .pem files, so in the same set as by default in the **Certificate file**. Private key file, **Certificate authority file** and **Client certificate file** parameters of the [ADOM Control Test Client](#) application, specify the corresponding paths to these parameters.

Configure ADOM Control to enable SSL encryption

The main settings for SSL encryption in ADOM Control are [configured in the Manage SSL cluster action](#), which provides the following functionality for the cluster:

- [Manage SSL cluster action](#) allows you to enable SSL for multiple components of ADOM Control to connect at once.
- [SSL settings in the configuration of ADOM Control services](#) allow you to configure SSL for the main service components, for example, you can change the default paths to certificate and key files.

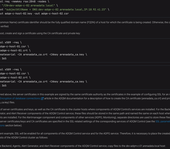
Manage SSL cluster action

1. Click the **Clusters** page, select the [Cluster](#) of the ADOM Control cluster to open the list of [node actions](#), and select **Manage SSL**.



Launch the "Manage SSL" action

ADOM will display a dialog window. After you click the **Enable SSL** button, a window opens allowing you to specify which services of ADOM Control should use secure connections.



Options for enabling SSL in the "Manage SSL" action configuration window

2. Fill in the required parameters for the [Manage SSL action](#) configuration window:

- Parameters in the **SSL** section — to enable SSL encryption of incoming and internal connections for various services of ADOM Control.

Parameters to enable SSL for services

Manage SSL parameter	Parameter description	Service parameter
ADPS — Secure ADPS	Enable SSL encryption of connections to the data storage service of ADOM Control	**ADPS** service (**Enable SSL** parameter)
ADOM Control — Secure Backend	Enable SSL encryption of connections to the Backend component of the Monitoring service of ADOM Control	**ADOM Control** service (**ADOM Control's Backend** — **Enable SSL** parameter
ADOM Control — Secure ADOM Agent	Enable SSL encryption of connections to the ADOM Agent component of the Monitoring service of ADOM Control	**ADOM Control** service (**ADOM Control's ADOM Agent** — **Enable SSL** parameter)
ADOM Control — Secure Alert Generator	Enable SSL encryption of connections to the Alert Generator component of the Monitoring service of ADOM Control	**ADOM Control** service (**ADOM Control's Alert Generator** — **Enable SSL** parameter)
ADOM Control — Secure Alert Receiver	Enable SSL encryption of connections to the Alert Receiver component of the Monitoring service of ADOM Control	**ADOM Control** service (**ADOM Control's Alert Receiver** — **Enable SSL** parameter)
ADOM Control — Secure Monitoring	Enable SSL encryption of connections to the Monitoring component of the Monitoring service of ADOM Control	**ADOM Control** service (**Monitoring**) — **Enable SSL** parameter
Monitoring — Secure Prometheus	Enable SSL encryption of connections to the Prometheus Server component of the Monitoring service	**Monitoring** service (**Prometheus**) — **Enable SSL** parameter
Monitoring — Secure Grafana	Enable SSL encryption of connections to the Grafana component of the Monitoring service	**Monitoring** service (**Grafana**) — **Enable SSL** parameter
Monitoring — Secure Loki	Enable SSL encryption of connections to the Loki component of the Monitoring service	**Monitoring** service (**Node exporter**) — **Enable SSL** parameter

- The **Use service-client** parameter — to specify whether to check the cluster health and availability after applying the settings.

The **Enable SSL** button and the **Use service-client** parameter are disabled by default.

Having set the **Manage SSL** action parameters, click the **Enable SSL** button. ADOM Control will configure SSL encryption for the selected services and restart these services.

To disable SSL encryption for the selected services, click the **Disable SSL** button. After you select [Manage SSL action](#) and click **Disable SSL**, ADOM Control will configure the selected services to not accept connections via secure channels and restart these services.

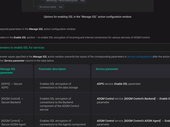
SSL parameters in service configurations

An advanced way to set up secure connections in ADOM Control is to configure SSL settings for each service separately.

1. Open the **Services** tab of any ADOM Control cluster and select the service for which you want to enable SSL encryption — ADPS, ADOM Control or Monitoring.
2. Click the **Configuration** action of the ADPS service and specify the following parameters in the configuration window:

ADPS: SSL Configuration

Parameter	Description	Default value
Enable SSL	Enable SSL encryption of connections to the data storage	Disabled
Certificate file	Path to the file with the SSL certificate (the X.509 format) of your key for the ADPS component. The key file should be located in the specified path on the node of the ADOM Control cluster on which the ADPS component is installed.	ssl.adps.fs.server.crt
Private key file	Path to the file with the private key of the SSL certificate (the X.509 format) of your key for the ADPS component. The key file should be located in the specified path on the node of the ADOM Control cluster on which the ADPS component is installed.	ssl.adps.fs.server.key
Certificate authority file	Path to the file with the public key of the SSL certificate (the X.509 format) of your key for the ADPS component. The key file should be located in the specified path on the node of the ADOM Control cluster on which the ADPS component is installed.	ssl.adps.fs.ca.crt

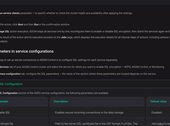


ADPS service configuration parameters to check SSL certificates of connections

3. Click the **Configuration** action of the ADOM Control service configuration, the following parameters are available:

ADOM Control SSL configuration

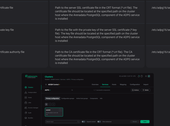
Parameter	Description	Default value
[ADOM Control's Backend] — Enable SSL	Enable secure incoming connections to the Backend component	Disabled
[ADOM Control's ADOM Agent] — Enable SSL	Enable SSL encryption of incoming connections to the Agents component	Disabled
[ADOM Control's Alert Generator] — Enable SSL	Enable SSL encryption of incoming connections to the Alert Generator component	Disabled
[ADOM Control's Alert Receiver] — Enable SSL	Enable SSL encryption of incoming connections to the Alert Receiver component	Disabled
[Alert Manager] — Enable SSL	Enable SSL encryption of incoming connections to the Alert Manager component	Disabled
[ADOM Control Service] — Certificate file	Path to the file with the SSL certificate (the X.509 format) of your key for the ADOM Control component. The key file should be located in the specified path on the node of the ADOM Control cluster on which the ADOM Control component is installed.	ssl.adpsms.fs.server.crt
[ADOM Control Service] — Private key file	Path to the file with the private key of the SSL certificate (the X.509 format) of your key for the ADOM Control component. The key file should be located in the specified path on the node of the ADOM Control cluster on which the ADOM Control component is installed.	ssl.adpsms.fs.server.key
[ADOM Control Service] — Certificate authority file	Path to the file with the public key of the SSL certificate (the X.509 format) of your key for the ADOM Control component. The key file should be located in the specified path on the node of the ADOM Control cluster on which the ADOM Control component is installed.	ssl.adpsms.fs.ca.crt
[Alert Manager] — Certificate file	Path to the file with the SSL certificate (the X.509 format) of your key for the Alert Manager component. The key file should be located in the specified path on the node of the ADOM Control cluster on which the Alert Manager component is installed.	ssl.alertmanager.server.crt
[Alert Manager] — Private key file	Path to the file with the private key of the SSL certificate (the X.509 format) of your key for the Alert Manager component. The key file should be located in the specified path on the node of the ADOM Control cluster on which the Alert Manager component is installed.	ssl.alertmanager.server.key
[Alert Manager] — Certificate authority file	Path to the file with the public key of the SSL certificate (the X.509 format) of your key for the Alert Manager component. The key file should be located in the specified path on the node of the ADOM Control cluster on which the Alert Manager component is installed.	ssl.alertmanager.ca.crt



Configuration parameters of the ADOM Control service to check SSL certificates of connections

ADOM Control: ADOM management

The **Connection type** parameter in the **ADOM management** section of the ADOM Control service configuration allows to enable the use of only secure TCP connections for the ADOM Control service. To do this, set the **Connection type** parameter to **ADPS over TCP/SSL** — the **Connection type** parameter will enable [Secure TCP/IP connection](#) and the [ADPS Control](#) service will be launched. [Secure TCP/IP connection](#).

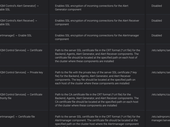


Enable the use of only secure TCP connections to ADOM

Monitoring: SSL configuration

The following parameters in the **SSL configuration** section of the Monitoring service configuration allow you to enable SSL encryption and connections to various components of the Monitoring service of ADOM Control. To enable SSL encryption, set the **Enable SSL** parameter to **Enabled** — the rest of the parameters will be available for editing.

Parameter	Description	Default value
[Prometheus] — Enable SSL	Enable SSL encryption of incoming connections for the Prometheus Server	Disabled
[Prometheus] — Certificate file	Path to the file with the SSL certificate (the X.509 format) of your key for the Prometheus Server	etc/prometheus/prometheus.crt
[Prometheus] — Private key file	Path to the file with the private key of the server SSL certificate (the X.509 format) of your key for the Prometheus Server	etc/prometheus/prometheus.key
[Prometheus] — Certificate authority file	Path to the file with the public key of the SSL certificate (the X.509 format) of your key for the Prometheus Server	etc/prometheus/prometheus.ca.crt
[Grafana] — Enable SSL	Enable SSL encryption of incoming connections for the Grafana component	Disabled
[Grafana] — Certificate file	Path to the file with the SSL certificate (the X.509 format) of your key for the Grafana component	etc/grafana/grafana.server.crt
[Grafana] — Private key file	Path to the file with the private key of the server SSL certificate (the X.509 format) of your key for the Grafana component	etc/grafana/grafana.server.key
[Grafana] — Certificate authority file	Path to the file with the public key of the SSL certificate (the X.509 format) of your key for the Grafana component	etc/grafana/grafana.ca.crt
[Node exporter] — Enable SSL	Enable SSL encryption of incoming connections for the Node exporter component	Disabled
[Node exporter] — Certificate file	Path to the file with the SSL certificate (the X.509 format) of your key for the Node exporter component	etc/node-server.crt
[Node exporter] — Private key file	Path to the file with the private key of the server SSL certificate (the X.509 format) of your key for the Node exporter component	etc/node-server.key

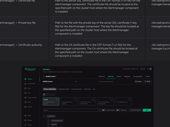


Configuration parameters of the Monitoring service for SSL connections

3. After specifying all the necessary parameters for configuring SSL connections, click **Save** and on **Reconfigure and restart** for services to apply the changes.

Example

1. In the **ADOM management** section of the ADOM Control service configuration, set the **Connection type** parameter to **Secure TCP/IP** (default) so that ADOM Control can connect to ADOM only through the [Secure TCP/IP connection](#) (**ADOM Control** — **Secure TCP/IP connection**).



Option to enable the use of only secure TCP connections to ADOM

Click **Save**.

2. Click **Manage SSL** action for the cluster. In the configuration window that appears, set the **Enable SSL** and click on the following options to encrypt traffic between ADOM components and services:

- **[ADOM Control]** — **Secure Backend**
- **[ADOM Control]** — **Secure ADOM Agent**
- **[ADOM Control]** — **Secure Alert Generator**
- **[ADOM Control]** — **Secure Alert Receiver**
- **[ADOM Control]** — **Secure Monitoring**



Enable SSL in ADOM Control for the "Manage SSL" action action



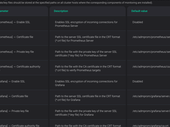
Disable SSL with the "Manage SSL" action

Test a connection

To test the ADOM Control connection in secure mode via HTTPS, you can open any web interface at https://10.0.0.1:8080 or https://10.0.0.2:8080 in the browser (in IP address of the host where the Backend component is installed). These URLs should open the secure ADOM Control over HTTPS.

Disable SSL

To disable SSL, click **Manage SSL** action and click the **Disable SSL** button at the bottom of the [Cluster](#) SSL action in the configuration of a service.



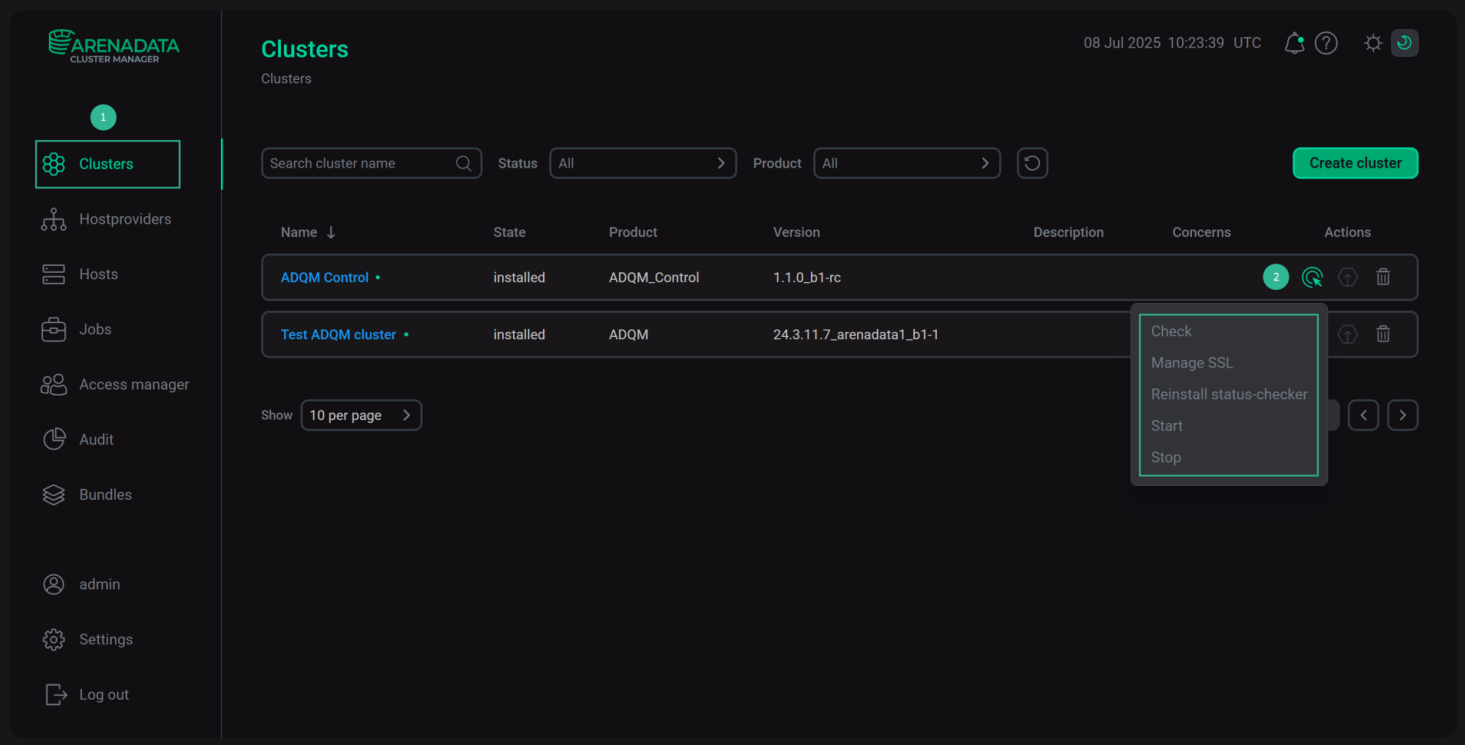
Disable SSL with the "Manage SSL" action

Cluster actions

Elena Dvoryadkina

To Table of Contents

You can manage an ADQM Control cluster on the ***Clusters*** page in the ADCM UI.



ADQM Control cluster in the ADCM UI

Refer to ADCM documentation for an overview of the ***Clusters*** page and common actions.

A set of actions available for an ADQM Control cluster depends on the cluster state:

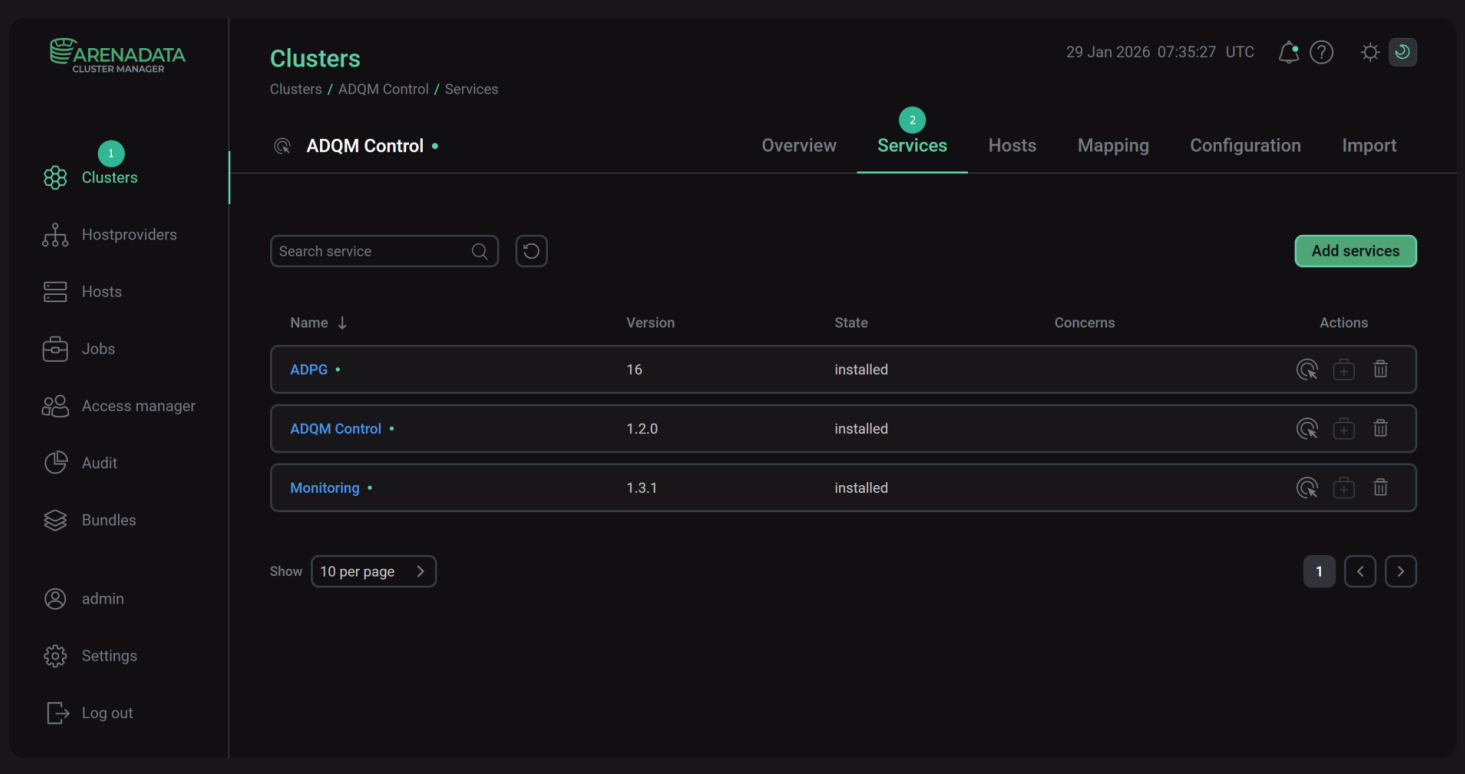
- For a new cluster (the cluster state is `created`), an available action is:
  - ***Install*** — installs all cluster services.
- After a cluster is installed (the cluster state is `installed`), the following actions are available:
  - ***Check*** — checks if all components of the installed cluster work correctly.
  - ***Manage SSL*** — allows you to configure and enable/disable SSL encryption for services of the ADQM Control cluster.
  - ***Reinstall status-checker*** — reconfigures and restarts the status checker for all cluster services. Use this action when migrating a cluster to a new ADCM server.
  - ***Start*** — starts all services of the cluster.
  - ***Stop*** — stops all services of the cluster.

Service actions

Elena Dvoryadkina

To Table of Contents

ADCM UI provides actions to manage services of ADQM Control. These actions are available on the ***Services*** tab. To open this tab, click an ADQM Control cluster name on the ***Clusters*** page and switch to the ***Services*** tab.



Services of an ADQM Control cluster in the ADCM interface

Refer to ADCM documentation for an overview of the ***Services*** tab and common actions. A set of actions available for working with a service of an ADQM Control cluster depends on the service type.

Actions available for all services:

- ***Install*** (if a service state is `created`) — installs a service.
- ***Check*** — checks the health of a service.
- ***Start*** — starts a service.
- ***Stop*** — stops a service.

Service-specific actions:

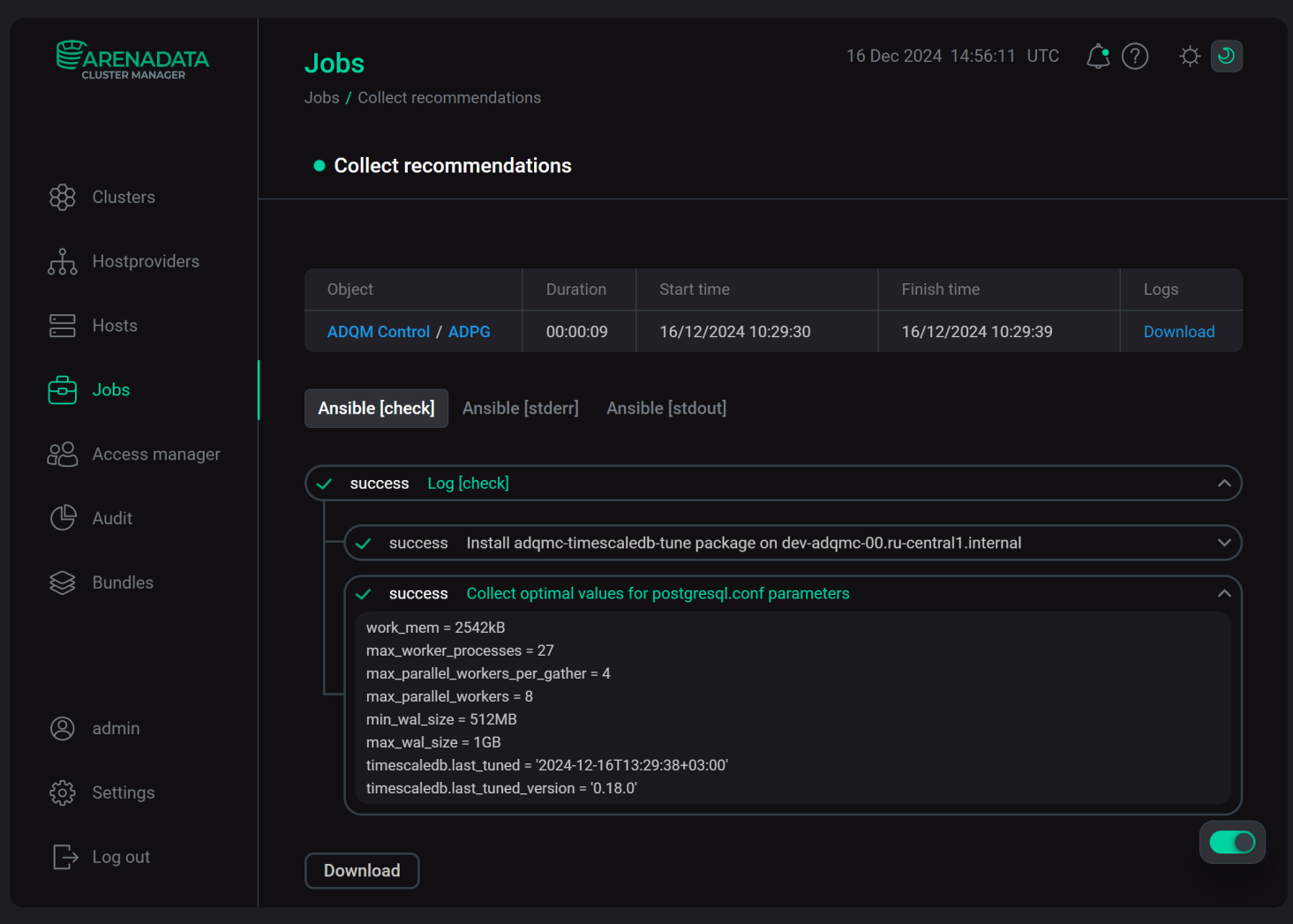
ADQM Control

- ***Reconfigure and restart*** — updates the service configuration files according to settings specified on the service configuration page, and restarts the service.
- ***Reinstall*** — reinstalls the service.

ADPG

- ***Reconfig and restart*** — updates the service configuration and restarts the service.
- ***Collect recommendations*** — collects optimal values for *postgresql.conf* configuration parameters to improve ADPG performance.

  After the action is performed, open the corresponding job on the Jobs page (by clicking ***Collect recommendations*** in the list of jobs) and in the ***Ansible [check]*** section, make sure that the ***Collect optimal values for postgresql.conf parameters*** subtask has been completed successfully. The result of this subtask contains a list of optimal parameter values — assign these values to the corresponding parameters in the ADPG configurations section of the ADPG service configuration page and run the ***Reconfig and restart*** action for the service.



The result of the "Collect recommendations" action

- ***Remove*** — removes the service from the cluster. This action can be used to delete an already installed service, unlike the icon that allows you to remove a non-mapped service (a service whose components have not yet been distributed among cluster hosts).

Monitoring

- ***Add/Remove components*** — opens the component-host mapping interface where you can add, remove, or redistribute service components.
- ***Reconfigure and restart*** — applies configuration to the service according to settings specified on the service configuration page, and restarts the service.
- ***Reinstall*** — reinstalls the service.

Configure an external database

Elena Dvoryadkina

To Table of Contents

For production environments, it is recommended to use an external PostgreSQL database as ADQM Control data storage for high performance and fault tolerance.

TIP

The recommended external service for ADQM Control is Arenadata Postgres (ADPG).

To configure an external database, follow the steps described below:

1. In PostgreSQL, run the following queries to create a user for ADQM Control to connect to an external database and a database owned by that user for use by ADQM Control services (the queries should be run under the default `postgres` user with `superuser` privileges):

```
CREATE ROLE <user_name> LOGIN PASSWORD '<user_password>';
CREATE DATABASE <database_name> OWNER <user_name>;
```

2. Connect to the created database under a superuser and load the following extensions:

```
CREATE EXTENSION IF NOT EXISTS timescaledb;
CREATE EXTENSION IF NOT EXISTS timescaledb_toolkit;
CREATE EXTENSION IF NOT EXISTS pgcrypto;
CREATE EXTENSION IF NOT EXISTS pg_trgm;
```

NOTE

Recommended extension versions:

- `timescaledb` — 2.15.0
- `timescaledb_toolkit` — 1.19.0
- `pgcrypto` — 1.3
- `pg_trgm` — 1.6

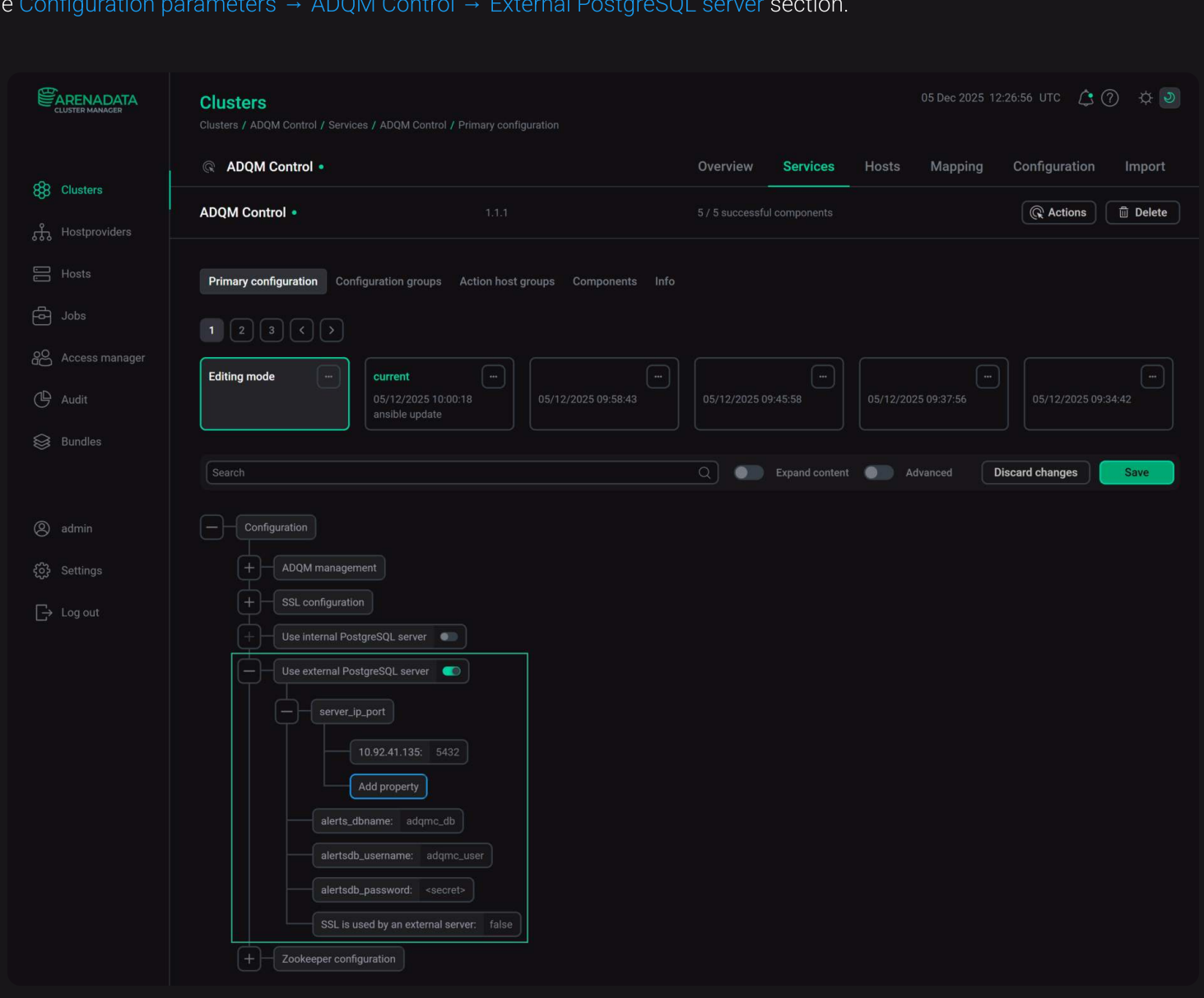
When using higher versions, compatibility testing is required to ensure correct system operation.

3. To enable access to the `<database_name>` database from an ADQM Control host as the `<user_name>` user, add an entry like the following to the *pg_hba.conf* file on the PostgreSQL server side:

```
host <database_name> <user_name> <adqmc_host_address> trust
```

where `<adqmc_host_address>` is an address of the ADQM Control host (for example, if an IPv4 address of the ADQM Control host is `10.92.41.87`, replace `<adqmc_host_address>` with `10.92.41.87/32`). You can find description of the `address` field and other fields of records in the *pg_hba.conf* file in the article The pg_hba.conf File.

4. On the configuration page of the ADQM Control service, activate the ***Use external PostgreSQL server*** toggle and fill in parameters in the list that opens. The assignment of parameters is described in the Configuration parameters → ADQM Control → External PostgreSQL server section.



ADQM Control parameters for connecting to an external PostgreSQL database

5. Click ***Save*** to save the ADQM Control configuration.

6. Install the ADQM Control service using the Install service action or execute the ***Reconfigure and restart*** action for the ADQM Control service if it is already installed.

Configure alerts in ADQM Control via web UI

Elena Dvoryadkina

Contents

To Table of Contents

System alerts
- Configure system alerts
- Types of system alerts

Internal alerts

The ***Alerts*** tab on the ***Settings*** page in the ADQM Control web interface allows you to configure parameters to be criteria for generating alerts.

To change default settings, edit the required fields and click ***Save***. Click ***Revert all*** if you need to cancel changes that have not been yet saved by clicking ***Save***.

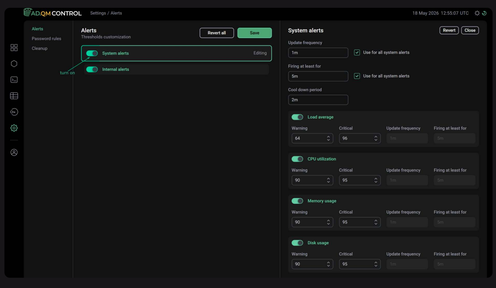
Alerts in ADQM Control are grouped into modules. Currently, ADQM Control can send and allows you to configure alerts of the System alerts and Internal alerts modules.

System alerts

The ***System alerts*** module unites system alerts — alerts that are generated based on values of system metrics that indicate general characteristics of ADQM cluster hosts, usually related to resource consumption (see the System alerts table). For each system metric, you can set thresholds against which ADQM Control compares the current value of the metric and determines whether to generate an alert and its severity level.

Configure system alerts

Use the ***System alerts*** switch on the ***Settings/Alerts*** page to enable/disable the generation of alerts based on system metrics. When the switch is enabled, the ***System alerts*** form on the right allows you to configure settings of system alerts.



Configure system alerts

System alert settings

Parameter	Description	Default value
Update frequency	Specifies how often to compare a metric value with thresholds. Once the metric exceeds a threshold, an alert is generated in ADQM Control (but not sent for processing). The parameter value should be in the range of 1-5 minutes	1m (m — minutes)
Firing at least for	Time during which a metric value should exceed a threshold for the corresponding alert to be sent for processing, after which it will appear in the ADQM Control interface. The parameter value should be in the range of 1-15 minutes	5m (m — minutes)
Cool down period	Period (after sending an alert) during which ADQM Control ignores an update when a metric value no longer exceeds a threshold. If the metric after this period still does not exceed the threshold, then the alert is considered to be no longer valid. The period starts again after each update when the metric exceeds the threshold (during the previous period) — in other words, it is extended. The parameter value should be in the range of 2-15 minutes, greater than the value of the ***Update frequency*** parameter	2m (m — minutes)
Warning	Metric value at which a medium-significance alert is generated. This level of alert importance indicates that there is a potential problem on an ADQM cluster host due to an increase in the value of the corresponding system metric (but this problem has not reached the critical level yet)	See default thresholds in the System alerts table
Critical	Metric value at which a high-significance alert is generated, indicating that a critical problem has been detected on an ADQM cluster host	See default thresholds in the System alerts table

You can set up the ***Update frequency*** and ***Firing at least for*** parameters in two ways:

- enter required values at the top of the ***System alerts*** form and enable the ☑ ***Use for all system metrics*** option to apply the specified values to all system alerts;
- specify parameter values separately for each alert type.

Types of system alerts

The table below describes the types of alerts that ADQM Control can generate while monitoring the corresponding system metrics on ADQM cluster hosts and comparing their values to specified thresholds.

System alerts

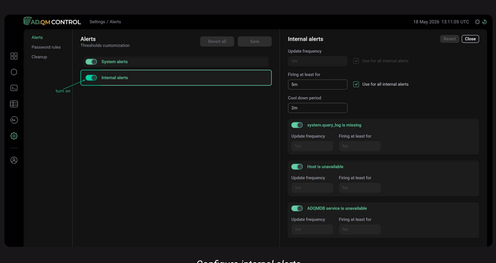
Alert type	Condition for generating an alert	Default thresholds
Load average	The system load average (a value is in the range `[0, 1]` for a single CPU and can be higher for multi-core systems) exceeds a threshold — `(LA15 + LA5)/2 > threshold`. This means either high CPU usage or disk IO activity take too long	Warning — 8, Critical — 16
CPU utilization	The CPU usage (as a percentage) exceeds a threshold	Warning — 90, Critical — 95
Memory usage	The RAM usage (as a percentage) exceeds a threshold	Warning — 90, Critical — 95
Disk usage	The disk capacity (as a percentage) exceeds a threshold	Warning — 90, Critical — 95

For each alert type, there is a switch in the ***System alerts*** form that you can use to disable the generation of alerts based on the corresponding metric.

Internal alerts

In the ***Internal alerts*** module settings, you can enable/disable the generation of alerts in the following cases:

- the `system.query_log` log table with information about executed queries does not exist in an ADQM cluster;
- some host of an ADQM cluster is unavailable;
- the ADQMDB service is unavailable on some host of an ADQM cluster.



Configure internal alerts

As with system alerts, you can control the following parameters for internal alerts:

- ***Update frequency*** — frequency of checking for existence of the `system.query_log` table or availability of hosts/ADQMDB service;
- ***Firing at least for*** — time during which the log table should be missing or an ADQM host/ADQMDB service should be unavailable so that the corresponding alert is added to ADQM Control;
- ***Cool down period*** — period after the alert is sent, during which ADQM Control ignores an update when logging to the `system.query_log` table has started or an ADQM host/ADQMDB service has become available.

View a dashboard

Petr Koutnovec

Contents

Cluster overview
- Heat map
- Recent alerts
- Acknowledged alerts
- Open alert details
- Go to the Cluster metrics page

Top 10 tables

Top 10 queries

The **Dashboard** page of the ADQM Control web interface displays general information about an ADQM cluster state as:

- [Heat map](#) - graphically represents the states of the ADQM cluster hosts.
- [Recent alerts](#) - lists the latest alerts about potential and critical issues detected on the ADQM cluster hosts.
- [Top 10 tables](#) - 10 largest tables by data amount in the ADQM cluster and 10 tables to which the most queries were sent.
- [Top 10 queries](#) - 10 queries that took the most time to execute and 10 queries that used the most memory.

The Dashboard page

In the fields at the top of the screen you can configure the following filters for the data displayed on the page:

- **Cluster** - ADQM cluster for which information should be displayed.
- **Time** - time period for which information should be displayed. Select one of the options from the drop-down list or enter a custom time range (at least 1 hour) on the **Calendar** tab.
- **Refresh** - frequency of data updates.

Cluster overview

Heat map

A heat map visualizes information about the states of all hosts in the ADQM cluster - each host is drawn as a square, and its color indicates the system state at the selected time interval.

Heat map on the Dashboard page

ADQM Control determines a host state and assigns the corresponding color to it in the heat map matrix based on existing alerts about host issues:

- There are no alerts about any issues with the host - **OK** (green).
- Fairly potential issues have been detected on a host (for example, the increase in the values of some system metrics), but they are not yet critical. Appropriate warning alerts for these issues have been generated in ADQM Control - **Degraded** (yellow).
- At least one critical issue has been detected on a host, and the corresponding alert of the high-importance level about it exists in ADQM Control - **Down** (red).
- A host in the ADQM cluster is unavailable, and an alert about this has been generated in ADQM Control (conditions for generating these alerts are configured in [Internal alerts](#)) - **Unknown** (gray).

TIP

From ADQM Control, you can navigate to the [ADQM ZooKeeper](#) UI interface (supported in ADQM since the version [1.7.0](#)), which provides a set of features for working with ADQM databases and tables. To do this, hover the mouse over a host in **Heat map** and click **ADQM ZooKeeper** in the pop-up window.

Recent alerts

The **Dashboard** page also displays the most recent alerts (up to 5) about problems on the ADQM cluster hosts, sorted in descending order by the time the alerts were generated, not by their importance.

Recent alerts on the Dashboard page

Alert severity levels

Severity	Description	Conditions for generating an alert
- warning - critical level of importance	A potential problem has been detected on a host. A critical problem has been detected on a host.	A system metric value exceeds a threshold set via the **Warning** parameter in the [System alerts](#) configuration. A system metric value exceeds a threshold set via the **Critical** parameter in the [System alerts](#) configuration.

You can select one or more hosts in **Heat map** (by clicking) - then the list of recent alerts will include the alerts only for the selected hosts.

Recent alerts about problems on the selected host

To get a full list (history) of alerts for all hosts of the cluster generated during the specified period of time, set **Host** **Full history** in the top right corner within the **Recent alerts** section - it will open the [Alerts history](#) tab of the **Cluster metrics** page.

Acknowledged alerts

The **Acknowledged alerts** option allows you to specify which alerts about problems on cluster hosts ADQM Control should take into account when generating the heat map and the list of recent alerts:

- **Acknowledged alerts** – all alerts generated during the selected time period.
- **Acknowledged alerts** – only those alerts, to which you set an acknowledgment. ADQM Control highlights relevant alerts (requiring your attention to be resolved) with the bell sign in their fields: 

A selected host has no issues that require analysis

Then, the host  appearing on the host, and ADQM Control generated a warning alert. This alert is now displayed in the **Recent alerts** list and the color of the host's pattern in **Heat map**, as shown in the example below.

Example of an unresolved issue on the host

If the disk usage issue has been analyzed and resolved, it is longer necessary to keep the alert about it in the list of acknowledged alerts (on the page with alert details) – see [Alerts history](#). After that, ADQM Control will not take into account this alert when monitoring the cluster health – it will display the host in green in **Heat map** again and will not display the alert in the **Recent alerts** list.

If you  the bell sign next to the alert, it will be removed from the list of acknowledged alerts, and ADQM Control will start to take it into account when generating the heat map and the list of recent alerts about the host again. For example, in the image below, all issues in **Heat map** are red – this means that a real-time critical issue was detected on each of these hosts. However, in the **Recent alerts** list, there are no relevant alerts, because all alerts about issues on the hosts were acknowledged in the past.

The Acknowledged alerts option is enabled

Open alert details

From the **Dashboard** page, you can access detailed information about a problem related to a host in one of the following ways:

- In **Heat map**, hover the mouse over a host → a pop-up window will show alerts about the last two problems found on that host. Click **Open alert** below an alert corresponding to an issue you want to review more information about.

Host issue details

- In the **Recent alerts** list, click the field  in an alert line.

Each of these actions opens a separate page with a detailed description of the corresponding issue. For example, the details of the alert  show which device is screaming for a large time. The **Overall** field summarizes the metrics' threshold values and [anomaly](#) values, based on which the alert was generated.

Host issue details

Use the **Acknowledged alerts** option on this page to change an alert status and mark the corresponding issue as acknowledged (or other words, known or resolved).

Go to the Cluster metrics page

From **Heat map** of the **Dashboard** page, you can go to the [Node overview](#) tab on the **Cluster metrics** page. It also shows the heat map matrix for an ADQM cluster and additionally displays charges of various system metrics for each host over time. To open the **Node overview** tab for a specific host, click on the desired host in the matrix (or select it in the drop-down list). For more information about a specific alert by clicking the corresponding row in the table → alert details will appear below the row.

Heat map on the "Cluster metrics" page

To go to the **Cluster metrics** → **Heat map** tab, do one of the following in **Heat map** of the **Dashboard** page:

- Click  in the pop-up window that appears when you hover the mouse over the selected (by clicking) host.
- Increase the  – change it to make sure the selected time interval is negative value means a decrease in the amount of data in the table).

- **Database** – database to which a table belongs.
- **Host** –  opens the [ADQM Distributed](#) UI interface in the [Cluth Distributed table tracking](#) tab for the host depending on the ADQM cluster is considered a single aggregated table).

Depending on the selected option, **Separated** or **Aggregated** (the following information on ADQM data tables (in addition to **Size** and **Request count**) is provided in this section:

Separated

- **Table name** – table name (only on a table name opens a page with detailed information on table columns).
- **Size** –  – number of data rows in a table on all hosts of the selected time interval.
- **Request count** –  – number of queries to a table on all hosts of the selected time interval.
- **Host** – table information on the ADQM hosts (opens the ADQM Control list of hosts).
- **Last request** – time the last query to a table was run during the selected time interval.
- **Parser** –  – change in table size over the selected time interval (a negative value means a decrease in the amount of data in the table).
- **Database** – database to which a table belongs.
- **Host** –  opens the [ADQM Distributed](#) UI interface in the [Cluth Distributed table tracking](#) tab for the host depending on the ADQM cluster is considered a single aggregated table).

Aggregated

- **Table name** – table name (click on an aggregated table name opens a page with a list of tables in the group, from which you can go to detailed information on columns of a table located on separate hosts).
- **Row count** (on the **Biggest** tab) – total number of data rows in a table across all hosts at the end of the selected time interval.
- **Host** – table information on the ADQM hosts (opens the ADQM Control list of hosts).
- **Last request** – time the last query to a table was run during the selected time interval.
- **Parser** –  – change in the total size of a table over the selected time interval (a negative value means a decrease in the amount of data in the table).
- **Database** – database to which a table belongs.
- **Host** –  opens the [ADQM Distributed](#) UI interface in the [Cluth Distributed table tracking](#) tab for the host depending on the ADQM cluster is considered a single aggregated table).
- **Size** –  – distribution of table data between hosts by data size.
- **Requests initiator** (on the **Popular** tab) –  distribution of queries sent to a table on different hosts.

In the header of the **Table name** field, there is the field , which when clicked allows you to view or change the filter that determines from which ADQM tables the largest or most active tables are selected. The following parameters filters are available:

- **Existing** (default) – top 10 tables are selected from those that exist in the cluster at the end of the specified time interval (tables are considered existing if ADQM Control continues to collect information about them at the end of the time interval). To calculate values of aggregate fields (for example, row count), ADQM Control uses the data collected in the group of the end of the selected time interval.
- **Historical** – top 10 tables are selected from all tables that existed in the cluster during the specified time interval (including tables that were deleted). Values of aggregated table fields are calculated as the maximum value of the corresponding fields in the group of the selected time interval.

In each row of the **Size**, **Row count**, **Request count**, **Disk imbalance** and **Requests imbalance** columns, there is the icon  – that opens a modal window with a graph of changes in the corresponding metrics of an ADQM table. From this window you can go to the [Cluth Distributed table tracking](#) tab – to do this, click a table name in the top left corner of the window.

Sample graph for number of data rows in an ADQM table

NOTE

To get information on all cluster tables for the specified time period, see the [Tables](#) page.

Top 10 queries

The **Top 10 queries** section contains two tabs with information about queries to ADQM cluster databases during a specified time interval:

- **Longest** – top 10 executed (by completion time) queries that ran longest in the cluster during the specified time interval. Duration – duration of a query execution.
- **Used memory** – top 10 queries that required the most memory (amount of memory) to execute during the specified time interval.

List of the longest-running queries

List of queries used the most memory

For each query, the following information is also provided:

- **Query ID** – the query identifier (click on an identifier opens the **Query details** page where you can view the query text).
- **Average duration** (on the **Longest** tab) – average query execution time in milliseconds for queries whose host completely matches (= for parameterized queries, specific parameter values are not taken into account).
- **Result bytes** (on the **Memory** tab) – amount of memory to store the query result.
- **Databases** – names of databases to which the query is sent.
- **Tables** – names of tables to which the query is sent.
- **User** – name of an ADQM user who initiated the query execution.
- **Time started** – the query execution start time.
- **Host** – host where the query was run.

In the header of the **Query ID** field, there is the field  – by clicking on which, you can view or change the filter that determines from which queries the longest or most memory-intensive ones are selected. The following parameters filters are available:

- **Completed** (on the **Longest** tab) – top 10 longest-running queries (i.e. from all queries executed since).
- **All** (on the **Used memory** tab) – top 10 most memory-intensive queries are selected from all queries.
- **Successful** (default filter) – top 10 queries are selected from those that were completed successfully.

NOTE

To view the full history of queries for the required period of time, switch to the [Queries](#) page.

ADQM CONTROL

Contents

To Table of Contents

Heat Map
- Host state
- Alerts related to a host

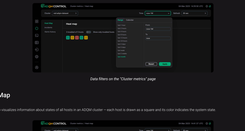
Incidents
- How incidents are generated
- Settings

Alerts history
- Alert filters
- Alert details

The **Cluster metrics** page of the ADQM Control web interface is designed to display information about an ADQM cluster state based on values of metrics collected from the cluster hosts. The page includes the data described below: Heat map, Incidents, and Alerts history.

To select data to be shown on the **Cluster metrics** page, you can use the fields at the top of the screen:

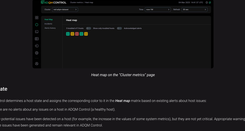
- **Cluster** – ADQM cluster for which information is displayed.
- **Time** – time period for which information is required. In the window that opens when you click the field, you can select an interval from the offered options on the **Range** tab or set a custom time range on the **Calendar** tab.
- **Refresh** – frequency of data updates.



Data filters on the "Cluster metrics" page

Heat Map

A heat map visualizes information about states of all hosts in an ADQM cluster – each host is shown as a square and its color indicates the system state.



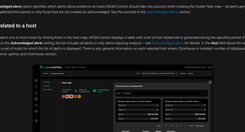
Heat map on the "Cluster metrics" page

Host state

ADQM Control determines a host state and assigns the corresponding color to it in the **Heat map**:

- there are no alerts about a host, and its host is ADQM Control is healthy hosts;
- only potential issues have been detected on a host (for example, the increase in the values of some system metrics), but they are not yet critical. Appropriate warning alerts for these issues have been generated and remain relevant in ADQM Control;
- at least one critical issue has been detected on a host, and the corresponding alert is in ADQM Control list.

The **Show only troubled hosts** switch at the top of the **Heat map** tab allows you to configure the heat map so that it displays only troubled hosts.

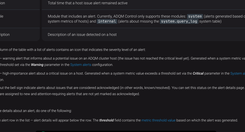


Heat map shows only troubled hosts

The **Acknowledged alerts** option specifies which alerts about problems on hosts ADQM Control should take into account when creating the cluster heat map – all alerts generated during the selected time period or only those that are not marked as acknowledged. See the example in the Acknowledged alerts section.

Alerts related to a host

When you select one or more hosts by clicking them in the heat map, ADQM Control displays a table with a list of host-related alerts generated during the specified period of time. Alerts are displayed in descending order of the **Date start** field value. You can also search for alerts by the **Module** and **Host** fields. Click the **Reset** button to clear the search results. You can change a set of hosts for which list of alerts is displayed. There is also general information on each selected host where Clickhouse is installed: number of databases, number of tables, server uptime, and ClickHouse version.



Alerts related to a host

The table with a list of alerts contains the following information.

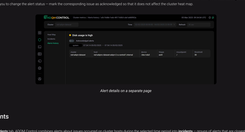
Field	Description
Date start	Timestamp when an alert was generated (in the `2017-07-??T?? hh:mm:ss` format)
Duration	Total time that a host been alert-remained active
Module	Module that includes an alert. Currently, ADQM Control only supports three modules: `system` (alerts generated based on system metrics of hosts) and `internal` (alerts about missing the `system.query_log` system table)
Description	Description of an issue detected on a host

The first column of the table with a list of alerts contains an icon that indicates the severity level of an alert:

- a high-importance alert about a potential issue on an ADQM cluster host (the issue has not reached the critical level yet). Generated when a system metric value exceeds a threshold set via the **Warning** parameter in the *system.alerts* configuration.
- a high-importance alert about a critical issue on a host. Generated when a system metric value exceeds a threshold set via the **Critical** parameter in the *system.alerts* configuration.
- a low-importance alert about a potential issue (not acknowledged) (in other words, known/resolved). You can see this status on the alert details page. The icons are assigned to new and attention-requiring alerts that are not yet marked as acknowledged.

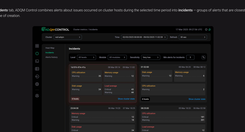
To get more details about an alert, do one of the following:

- Click an alert row in the list – alert details will appear below the row. The **threshold** field contains the metric threshold value based on which the alert was generated.



Alert details within the list

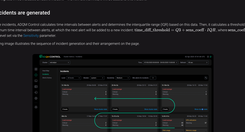
- Click the icon in the alert row – this will take you to a separate page with alert details (the page has a unique address that you can share). The **Acknowledged alerts** option allows you to change the alert status – mark the corresponding issue as acknowledged so that it does not affect the cluster heat map.



Alert details on a separate page

Incidents

On the **Incidents** tab, ADQM Control combines alerts about issues occurred on cluster hosts during the selected time period into **incidents** – groups of alerts that are closest to each other in time of creation.



Incidents on the "Cluster metrics" page

An incident is displayed as a block that includes:

- Incident duration.
- Incident time interval – from the time the first alert in the incident was generated until the time the last alert in the incident was deactivated (the metric-related issue was resolved).
- metrics for which thresholds were exceeded on cluster hosts, with information about how many warnings and high-importance alerts were generated.
- number of hosts where problems were detected.

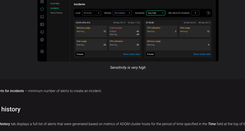
- **Show cluster state** link – refers to the *healthcheck* page with the *Acknowledged alerts* option enabled and the **Time** period equal to the incident interval. If an incident interval is less than 5 minutes, then the period is increased to 5 minutes. For example, if the incident interval is 2 minutes, then the **Time** field in the healthcheck page will contain the interval from 2 minutes before the incident start time to 3 minutes after the incident end time. Thus, the **Dashboard** page you can analyze your cluster state during the time period corresponding to the incident. For example, you can see which hosts had problems, which queries were run on a cluster and how many concurrent queries there were, which tables were the largest in size and which ones were the most queried.

A block highlighted in red indicates an active incident – if a new alert arrives, it will be added to this incident.

How incidents are generated

To generate incidents, ADQM Control calculates time intervals between alerts and determines the interquartile range (IQR) based on this data. Then, it calculates a threshold value that is the minimum time interval between alerts, at which the next alert will be added to a new incident: **time_diff_threshold = Q3 + max_coef * IQR**, where **max_coef** is the sensitivity level set via the *healthcheck* parameter.

The following image illustrates the sequence of incident generation and their arrangement on the page.



Arrangement of incidents on the page

Alerts are grouped into incidents dynamically depending on the specified parameters (see the section below) and the time interval selected in the **Time** field:

- For a fixed time interval in the past, the structure of incidents is generated when the **Incidents** page is opened and remains fixed because the set of alerts for this time interval does not change.
- If you select a time interval relative to the current moment (for example, the interval **now-12h**: **now-2d**: **now-1M**, etc.), the structure of incidents on the page may change, since the set of alerts for the selected interval shift over time, which means the set of active alerts may change (new alerts may occur or the time of previously valid alerts may become outside the current time frame).

Settings

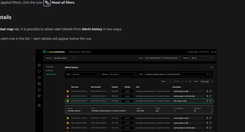
The **Incidents** tab contains the following parameters that affect which alerts are used to build incidents and how alerts are combined:

- **Level** – alert importance level. Possible values: `warning`, `critical`.
- **Module** – module in which alerts belong. Possible values: `system`, `internal`.
- **Sensitivity** – coefficient that affects how much time should pass between an alert and the previous one so that they fall into different incidents. The lower the sensitivity, the longer the time should pass between alerts to place them into different incidents. Possible values: `Very Low`, `Low`, `Medium`, `High`, `Very High`.

Example

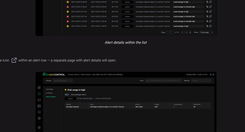
The images below show examples of the alert distribution for the same time interval across incidents depending on the specified sensitivity.

- With the `Very Low` sensitivity, all alerts for the selected time are combined into one incident.



Sensitivity is very low

- With the `Very High` sensitivity, alerts for the same time period are grouped into ten incidents.



Sensitivity is very high

- **Min alerts for incidents** – minimum number of alerts to create an incident.

Alerts history

The **Alerts history** tab displays a full list of alerts that were generated based on metrics of ADQM cluster hosts for the period of time specified in the **Time** field at the top of the screen.

Alerts history on the "Cluster metrics" page

Most of the fields in the **Alerts history** table match the fields that are described above for the **Heat map** tab. Additionally, the following fields are available:

- **Date finished** – timestamp in the `2017-07-??T?? hh:mm:ss` format, when an alert became no longer valid, i.e. the corresponding issue was resolved on the host. If an alert is still active, the field shows a time in the future.
- **Host** – host on which an issue was detected.

The icon in the **Ack** column indicates whether an issue requires attention or is set as acknowledged.

Alert filters

You can use filters located at the top of the **Alerts history** tab to specify which alerts to display in the table:

- **Level** – alert importance level. Possible values:
  - `warning` – potential issues on hosts;
  - `critical` – critical issues on hosts.
- **Module** – module into which alerts are grouped. Possible values: `system`, `internal`.
- **Host** – host for which you want to display alerts on issues found in the system. You can select multiple or all hosts at once.

To reset all applied filters, click the icon – **Reset all filters**.

Alert details

As for the **Heat map** tab, it is possible to obtain alert details from **Alerts history** in two ways:

- Click an alert row in the list – alert details will appear below the row.

Alert details within the list

- Click the icon within an alert row – a separate page with alert details will open.

Alert details on a separate page

Work with tables in ADQM Control

← Main Screens Menu

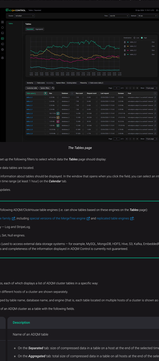
ADQM CONTROL

Contents

To Table of Contents ↑

List of tables
Graphs
Filter and sort data
Table details
Related queries

The **Tables** page of the ADQM Control web interface displays information about data tables in an ADQM cluster connected to the monitoring system. On this page, you can view a list of all cluster tables with general information about them, switch to details on columns of an individual table, and also visually identify changes in sizes of tables using graphs.



The Tables page

At the top of the screen, you can set up the following filters to select data from the **Tables** page should display:

- **Cluster** – ADQM cluster whose data tables are located.
- **Step** – time period for which information is shown should be displayed. In the window that opens when you click the field, you can select an interval from the offered options or the **Range** tab or set a custom time range (at least 1 hour) on the **Calendar** tab.
- **Refresh** – frequency of data updates.

> ADQM Control supports the following ADQM-compatible database table engines (i.e. can store tables based on these engines on the **Tables** page):
>
> - engines of the [MergeTree family](https://clickhouse.com/docs/en/engines/table-engines/mergetree-family/) 🔗 including [special versions of the MergeTree engine](https://clickhouse.com/docs/en/engines/table-engines/mergetree-family/) 🔗 and [Replicated](https://clickhouse.com/docs/en/engines/table-engines/mergetree-family/replication) engines 🔗
> - engines of the Log family — Log and StripeLog
> - the Memory, Merge, Join, Set, Null engines.
>
> For integration table engines (used to access external data storage systems – for example, MySQL, MongoDB, HDFS, Hive, S3, Kafka, EmbeddedRocksDB, RabbitMQ, PostgreSQL), the correctness and completeness of the information displayed in ADQM Control is currently not guaranteed.

List of tables

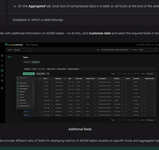
The **Tables** page contains two tabs, each of which displays a list of ADQM cluster tables in a specific way:

- **Separated** – tables located on different hosts of a cluster are shown separately.
- **Aggregated** – tables are grouped by table name, database name and engine (that is, each table located on multiple hosts of a cluster is shown as one entry in the list).

The **Tables** page lists data tables of an ADQM cluster as a table with the following fields:

Field	Description
Table name	Name of an ADQM table
Size	• On the **Separated** tab: size of compressed data in a table on a host at the end of the selected time interval • On the **Aggregated** tab: total size of compressed data in tables on all hosts at the end of the selected time interval
Database	Database to which a table belongs

It is also possible to add more fields with additional information on ADQM tables – to do this, click **Customize table** and select the required fields in the drop-down list.



Additional fields

The **Separated** and **Aggregated** also provide different sets of fields for displaying metrics of ADQM tables located on specific hosts and aggregated tables, respectively.

Separated

Field	Description
Event	Table lifetime (since the start of monitoring the ADQM cluster in ADQM Control)
Row count	Number of data rows in a table on a host at the end of the selected time interval
Request count	Number of queries sent to a table on a host during the selected time interval
Last request	Time the last query to a table on a host was run during the selected time interval
Increase	Change in the total size of a table on a host over the selected time interval (a negative value means a decrease in the amount of data in the table)
Host	ADQM cluster host where a table is located. Clicking on the icon 🔗 opens [ADQM Dashboard](#) 🔗 at http://<*admin_host_IP-address*>:8123/dashboard for ADQM starting with the version [22.3.2.2](#) 🔗 or at http://<*admin_host_IP-address*>:8123/play for earlier versions of ADQM (where <*admin_host_IP-address*> is the value of the **host_name** tag in the *config.xml* file of ADQM Admin). If the system should be configured so that a user chooses the host FQDN instead of address
Engine	Table engine

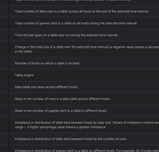
Aggregated

Field	Description
Event	Table lifetime (since the start of monitoring the ADQM cluster in ADQM Control)
Row count	Total number of data rows in a table across all hosts at the end of the selected time interval
Request count	Total number of queries sent to a table on all hosts during the selected time interval
Last request	Time the last query to a table was run during the selected time interval
Increase	Change in the total size of a table over the selected time interval (a negative value means a decrease in the amount of data in the table)
Hosts	Number of hosts on which a table is located
Engine	Table engine
Size, bytes	Data table size, bytes
Size, human-readable	Data table size, human-readable format
Rows, bytes	Size in the number of rows in a data table across different hosts
Requests, bytes	Size in the number of queries sent to a table in different hosts
Size imbalance	Imbalance in distribution of data tables between hosts by data size. Values of imbalance metrics are in the [CPU, USPS] range – the greater the value, the greater the imbalance
Rows imbalance	Imbalance in distribution of data tables between hosts by the number of rows
Requests imbalance	Imbalance in distribution of queries sent to a table on different hosts. For example, for 4 tables with a query distribution of **[1, 1, 1, 1]**, **[2, 2, 0, 0]**, the metric value will be **[0]**, and with a distribution of **[3, 1, 0, 0]**, **[4, 0, 0, 0]** – approximately **[0.5, 1]**.

Graphs

On the **Tables** page, you can get graphs to track the dynamics of changes in metrics of ADQM tables. All the columns of metrics for which graphs are supported, there is the icon 📈. Click the icon – above the list of tables, ADQM Control will display a panel with a graph built based on the corresponding metric of the ADQM table. You can add up to 8 graphs displayed simultaneously.

To see values of metrics at some point in time, hover the mouse over the graph area. The **Resolution** option allows you to choose the level of detail (number of points on the graph): **Low** or **High**.



Sample graphs by size of data tables

To delete a graph, click the icon corresponding to the graph color in the table the inserting 🔗 or the image above) or the icon 🗑 in the legend to the right of the graphs.

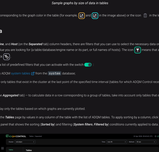
Filter and sort data

In the **Table name**, **Database**, **Engine**, and **Host** (on the **Aggregated** tab) column headers, there are filters that you can use to select the necessary data on the **Tables** page. To set a filter, click the icon 🔍, and enter a value you are looking for (a table database/engine name or host, or a full name). The icon 🔍 means that a filter is defined for the column. To reset all filters by a column, click 🔍.

Clicking on **System.Rows** opens a list of predefined filters that you can activate with the switch 🔘:

- **Show system tables** – display ADQM system tables (all those in the **system** database).
- **Only existing tables** – display only tables that exist in the cluster at the last point of the specified time interval (tables for which ADQM Control received metrics at the end of this time interval).
- **Only existing subtables** (on the **Aggregated** tab) – to calculate data in a row corresponding to a group of tables, take into account only tables that exist in the group at the last point of the specified time interval.
- **Only tables with growth** – display only the tables based on which graphs are currently plotted.

It is also possible to sort data on the **Tables** page by values in any column of the table with the list of ADQM tables. To apply sorting by a column, click the icon 🔍 or ▼ in its header. Above the list of tables, there is a panel that displays the **Sorted by** and filtering **System.Rows**, **Filtered by** options currently applied to data.



Current conditions for sorting and filtering data on the Tables page

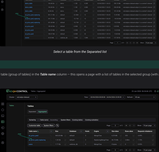
You can also view the list of the largest tables by data amount on the list of the most frequently queried tables in the ADQM cluster on the **Top 10 tables** section on the Dashboard page.

Table details

Open a page with detailed information about an ADQM table or a specific host as follows, depending on which tab on the **Tables** page you are navigating from:

Separated

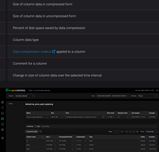
Click a table name in the **Table name** column.



Select a table from the Separated list

Aggregated

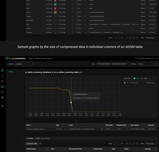
Click a name of an aggregated table (group of tables) in the **Table name** column – this opens a page with a list of tables in the selected group (with a full set of metrics as in the **Separated** list).



Select a group of tables from the Aggregated list

Opening up the **Only existing subtables** switch, click on the icon 🔍 in the header of the column of the group, that is located in the cluster during the specified time interval, or only tables that existed at the end point in time of this interval. The top of the page displays general information about the group of tables (the same as the corresponding row on the **Aggregated** list): **Clusters** table with data on the group of tables in the cluster, **Hosts** table with data on the group of tables in specific hosts of a cluster and metrics of an aggregated table. To do this, use the icon 🔍 in the column corresponding to the required metrics. The aggregated table graph is marked with the 🔍 sign in the legend.

To go to the details of a table located on a specific host, click on the name in the corresponding row of the list.

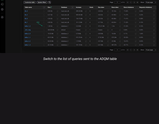


Select a specific table from a group

At the top of the page with ADQM table details, there is information about the table – its name and size, as well as the metrics selected on **Customize table** and the **Separated** tab.

This page also displays a table with the following fields that contain detailed information about each column of the selected ADQM table (to customize the visibility of some fields, use the **Customize table** option):

Field	Description
Column name	Column name at the end of the selected time interval
Size	Size of column data in compressed form
Uncompressed Size	Size of column data in uncompressed form
Compression	Amount of disk space saved by data compression
Type	Column data type
Codec	Data compression codecs 🔗 applied to a column
Comment	Comment for a column
Increase	Change in size of column data over the selected time interval



Information on table columns

You can filter data in the table by the **Column name** field (to set a filter, click 🔍 in the column header) and sort by any column except **Comment** (use the icons 🔍 and ▼ in column headers).

On the page, you can also build graphs to see how the size of compressed or uncompressed data in individual columns of an ADQM table changes over the selected time interval, or to analyze effect of applying data compression on the amount of disk space saved. To do this, use the icon 📈 in the **Size: Uncompressed Size**, or **Compression** column, respectively. The graph area can display up to 7 graphs by individual columns and one graph by the entire table size at a time.

Sample graphs by the size of compressed data in individual columns of an ADQM table

Sample graph by the amount of disk space saved with data compression

> **CAUTION**
>
> In the following cases, graph lines may be interrupted, which can lead to an inaccurate representation of column states and confuse users:
>
> - A column was renamed and then started to be used in the original name – at this time, the column (with any name) continued to exist and store data, and its metrics were collected.
> - A column was dropped and then re-created with the same name – in fact, there are two different columns existing at different time intervals.
>
> The rendering of graphs for such cases will be fixed in future releases of ADQM Control.

Related queries

From the **Tables** page (as well as from the page with ADQM table details), you can access the list of queries sent to a specific ADQM table for a selected time interval. To do this, click the icon 🔍 in the **Table name** cell corresponding to the desired data table – this opens the Queries page with automatically configured filters by table name, database, and host.

Switch to the list of queries sent to the ADQM table

Work with queries in AQOM Control

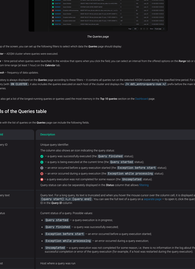
Anna Hovsepyan

Contents

- Fields of the Queries table
- Sort and filter data
- Query details
- Graphs
  - Duration
  - Used
  - Kind

To Table of Contents

The **Queries** page in the AQOM Control web interface displays SQL queries executed in AQOM clusters connected to the monitoring system. On this page, you can get an overview of queries executed in your clusters, sort and filter queries based on various criteria — for example, to analyze query performance, amount of resources used by queries, etc. You may have a look at queries executed by a specific user, queries executed in a specific time interval, queries that read from or write to a specific table, queries that read a certain number of bytes or different types, see how many queries in a specific time period ended with errors, and so on.



The Queries page

At the top of the screen, you can set up the following filters to work with data on the **Queries** page (in just display):

- **Cluster** — filter queries by cluster where they were executed.
- **Time** — time period when queries were launched (i.e. the queries that were given only the time) you can select an interval from the offered options or the **Range** (let a user to set a custom time range) and then press the **Update** button.
- **Refresh** — frequency of data updates.

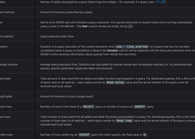
You can also use the **Search** box in the **Queries** page (according to Always filter + to operate) get queries run on the selected AQOM clusters during the specified time period. For each query, the **ID** (UUID) and a short fragment of the query text are shown. You can sort the list of queries in ascending or descending order by clicking the column header.

Fields of the Queries table

The table with the list of queries on the **Queries** page can include the following fields:

Field	Description
Query ID	Unique query identifier. The column will be always shown with following the query status.
Status	— A query was successfully executed (the **Query Finished** status). — A query is currently being executed (the **Query Started** status). — A query ended with an error (the **Exception before start** status). — A query was executed during a query execution error (the **Exception while processing** status). — A query was interrupted by the user (the **Interrupted** status).
Query text	Query status can also be separately displayed in the **Status** column that allows filtering.
Query kind	Query text. Clicking on the query text opens the query details page (see the column), and, if displayed as `(query started)` and **Query kind**). You can see the full text of a query in a query details page. To open it, click the query in the **Query ID** column.
Status	Current status of a query. Possible values: — **Query started** — a query execution is in progress. — **Query Finished** — a query was successfully executed. — **Exception before start** — a query execution started. — **Exception while processing** — an error occurred during a query execution. — **Interrupted** — a query was interrupted by the user.
Host	Host where a query was run.
Exception	Text of an error occurred before or during a query execution.
Exception code	Code of an error occurred before or during a query execution.
Frequency	How many times a query was executed in the cluster during the selected time period or the number of all queries executed. Applies on the page. Statistics are calculated for queries whose text completely matches (i.e. for parameterized queries, queries with different parameter values are grouped).
Databases	Names of databases accessed by a query (there may be multiple — for example, if a query uses `JOIN`).
Tables	Names of tables accessed by a query (there may be multiple — for example, if a query uses `JOIN`).
Used memory	Amount of memory consumed by a query.
User	Name of an AQOM user who initiated a query execution. For queries executed on cluster hosts when running a distributed query, a user is not added. The **User** column shows an empty string (`''`).
Time started	Query execution start time.
Duration	Duration of a query execution in the current execution time (`now() - time started`) of a query that has not yet been completed (with the **Query started** status). For queries with the **Exception before start** status, the duration is not shown.
Average duration	Average query execution time. Statistics are calculated for queries whose text completely matches (i.e. for parameterized queries, queries with different parameter values are grouped).
Read bytes	Total amount of data read from all tables and table functions participated in a query. For distributed queries, this is the number of read bytes on the initiator server. Don't confuse with the **Read rows** column, which shows the number of read rows.
Read rows	Total number of rows read from all tables and table functions participated in a query. For distributed queries, this is the total number of rows read on all servers. Don't confuse with the **Read bytes** column, which shows the number of read bytes.
Written rows	Number of rows written by an `INSERT` query (for other queries, the field value is `0`).
Written bytes	Amount of data written by an `INSERT` query (for other queries, the field value is `0`).
Query type	Query type. Possible values: `ALTER` `ALTER_DELETE` `CREATE` `ALTER_UPDATE` `DELETE` `DROP` `INSERT` `KILL` `OPTIMIZE` `OTHER` `RENAME` `SELECT` `SET` `SYSTEM` `TRUNCATE` `UNDEFINED` `EXISTS` `SHOW` `USE` `DESCRIBE` `EXPLAIN` `CHECK` `SHOW_CREATE_TABLE` — **Read** — for the `SELECT`, `EXPLAIN`, `SHOW`, `DESCRIBE`, `EXISTS`, `CHECK`, and `USE` queries. — **Write and Other** — for all other queries.

To display the fields of the **Queries** table, click **Customize table** and select the desired fields in the drop-down list. The **Query ID** and **Query text** fields are always visible, they cannot be hidden.



Managing fields of the Queries table

Sort and filter data

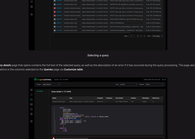
You can sort the data in the table by any field. In some fields the queries are sorted when the mouse is over a column with a hyperlink.

You can also filter queries in the following columns:

- **Query kind** — enter a full or partial query text in the field that appears when you move the mouse over a column with a hyperlink.
- **Databases**, **Tables** — enter a name of a database or table (for cases with multiple databases or tables separated by commas) in the **Databases** or **Tables** column. Warning: entry only if the query uses the specified number of tables and databases.
- **Exception code** — enter a numerical error code (or multiple codes separated with a comma) in the field with an icon and press **Enter** or the **Process** tab.
- **Host** — enter a name of a host where queries were executed. Warning: for queries executed on cluster hosts when running a distributed query, filtering by an empty value in the **User** column is not supported.
- **Duration** — you can enter the minimum and maximum query execution time (in seconds).
- **Tables**, **Type** — you can select a value (query status or type) from the drop-down list.

The icon in the column header indicates that the column is used to filter queries.

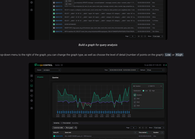
Table sorting is available in the following columns: **Exception** (queries ending in errors), **Exception code**, **Used memory**, **Time started**, **Duration**, **Read bytes**, **Read rows**, **Written bytes**, **Written rows**, **Query kind**, **Status**, **Host**, **Databases**, **Tables**, **User**, **Type**. To sort the data in the table by the selected column in ascending or descending order, click the column header. Click the column header again to change the sort direction. The **Exception** and the **Frequency** columns cannot be sorted.



Current conditions for sorting and filtering data

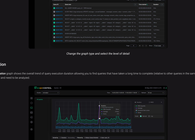
Query details

To view detailed information on a query, click its identifier in the **Query ID** field of the **Queries** table.



Selecting a query

The **Query details** page that opens contains the full text of the selected query, as well as the description of all error if it has occurred during the query processing. The page also displays query statistics for the current query in the **Common stats**.



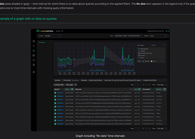
The Query details page

Graphs

On the **Queries** page, you can build graphs to analyze queries in your AQOM cluster: stats during a specified period of time, based on the following metrics:

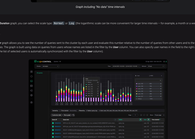
- **Duration** — total time of queries.
- **Used** — number of queries sent by particular users.
- **Kind** — number of queries of various types.
- **Host** — distribution of queries across cluster hosts (with denoting possibility accelerate by various metrics of load (total query execution).

To build a graph, click **Show graph** and select the appropriate tab from the drop-down list.



Build a graph for query analysis

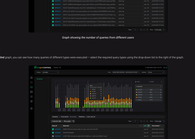
In the drop-down menu to the right of the graph, you can change the graph type, as well as choose the type of stacked chart of points on the graph: **Line** or **Area**.



Change the graph type and select the kind of area

Duration

The graph shows the overall trend of query execution duration allowing you to find queries that have taken a long time to complete (relative to other queries in the same time interval) and examine the average.



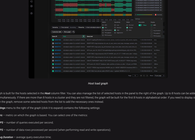
Graph of queries by execution duration

The graph displays the following information on the execution time of queries (and no on AQOM cluster) during a specified period of time:

- **95th Quantile** — 95th quantile of query execution time.
- **Average** — average query execution time.
- **Extreme zone** (points shown in red) The **Average** tab is above the **95th Quantile** tab — time interval where there are only a few queries (values in most cultures). For example, in the graph above, the **Average** tab is above the **95th Quantile** tab in the time interval from 12:00 to 12:30. Most queries in this time interval are executed in 1 s, but there are queries with a duration of 20 s. Such queries can be considered outliers and can be examined in more detail.

  Dark zone (zone between the **Average** and **95th Quantile** tabs — time interval when duration of queries are comparable. A peak in this zone means that all queries in the cluster have increased in duration and may require attention.
- **Event** — number of queries.

Hover the mouse over the graph. Then information for which there is no data about queries according to the applied filters. The **No data** item appears in the report only if the graph contains one or more time intervals with missing query information.

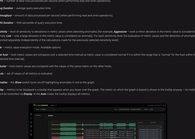


Graph including "No data" time intervals

For the **Duration** graph, you can select the tooltip type: **Normal** or **Log**. In logarithmic scale can be more convenient for larger time intervals — for example, a month or a week.

Used

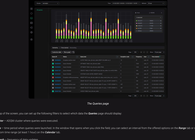
The **Used** graph allows you to see the number of queries sent to the cluster by each user and evaluate the comfort relative to the number of queries from other users and to the load of all queries. The graph is built using data on queries from users whose names are listed in the filter by the **Users** search. You can also specify users' names in the field to the right of the graph (the list of selected users is displayed in the field by the **Users** button).



Graph showing the number of queries of different types

Kind

On the **Kind** graph, you can see how many queries of different types were executed — select the required query types using the drop-down list to the right of the graph.



Graph showing the number of queries of different types

The graph is built according to the filters set in the table of queries. For example, the graph above shows the number of queries performed by the **admin** user on the selected time period.

Host

Using the **Host** graph, you can analyze the load on AQOM cluster hosts and find potential anomalies in their work or executing distance queries. In this context, an anomaly is a deviation in the value of a metric at a certain point in time from its value under the "usual" conditions (over a selected time period). For example, the output of the metric on the number of queries executed on the host per second can be used to detect the host on which the number of queries per second is significantly higher than on the other hosts in the cluster — then the behavior of the host can be considered anomalous. The task of detection of the graph is the search for anomalies in the behavior of individual hosts in the cluster.

Host load graph

The graph is built for the hosts selected in the **Host** system filter. If no hosts are selected in the filter, the graph will display data for the top 5 hosts with the highest load (in terms of the selected metric). If there are more than 5 hosts in a cluster and they are comparable in load, the graph will be built for the first 5 hosts in alphabetical order. If you need to display other hosts in the graph, select the desired hosts in the filter and click the **Update** button.

The **Settings** button to the right of the graph (click it to expand) is similar the following settings:

- **Metric** — metrics on which the graph is based. You can select one of the metrics:
  - **QPS** — number of queries executed per second.
  - **RPS** — number of queries executed per second (when performing read and write operations).
  - **Avg bandwidth** — average query execution time.
  - **Throughput** — number of queries executed per second (when performing read and write operations).
  - **P95 Duration** — 95th percentile of query execution time.
- **Granularity** — level of granularity for data points in metric values when detecting anomalies (for example, **Aggregate** — over a minute duration in the metric value) is considered an anomaly, **Leaf** — only a large deviation in the metric value is considered an anomaly, and the value of the metric values with the deviations of anomalies are averaged over the entire time interval for the graph (for example, over a minute).
- **Anomaly** — the level of anomalies in the graph. The more value on which the graph is based is shown in the monthly anomaly — its visibility (levels) are controlled by **Display** the **Auto** mode, the levels displays all metrics).



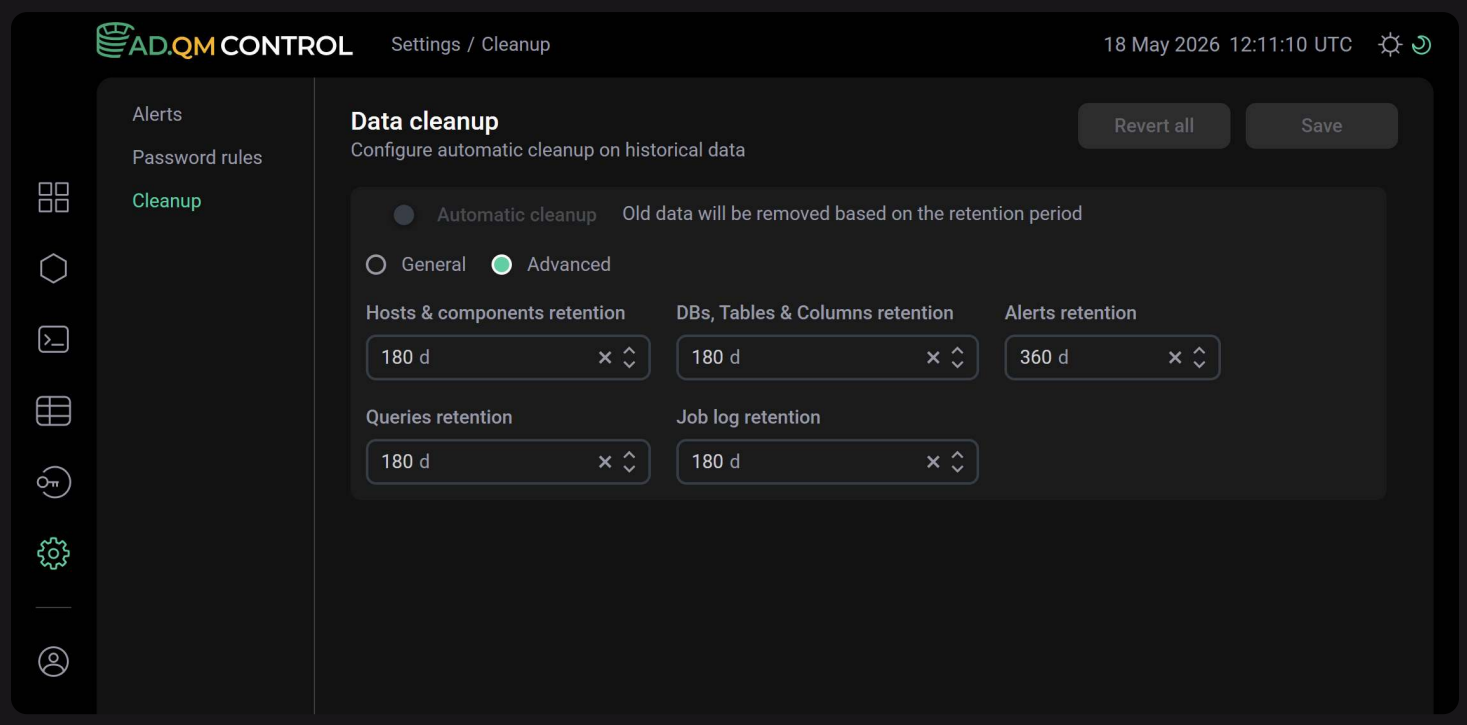
Metric values on the host graph

Data cleanup

Elena Dvoryadkina

To Table of Contents

On the ***Settings*** → ***Cleanup*** page in the ADQM Control web interface, a system owner (user with the `Owner` role) can enable automatic deletion of historical data (for example, outdated data stored for a long time) from ADQM Control in accordance with the specified retention period.



Configure ADQM Control data cleanup

The ***Automatic cleanup*** switch enables/disables the data cleanup job.

A retention period, after which data will be automatically deleted from ADQM Control, can be configured in two ways:

- Enable the ***General*** option and set a retention period common to all data types (information on ADQM tables/queries/hosts, alerts, etc.) using the ***Common retention*** parameter.
- Enable the ***Advanced*** option and configure the storage time separately for each data type using the following parameters:
  - ***Hosts & components retention*** — information on ADQM cluster hosts and components;
  - ***Queries retention*** — information on SQL queries run in the ADQM cluster;
  - ***DBs, Tables & Columns retention*** — information on ADQM cluster databases and tables;
  - ***Job log retention*** — logs of system jobs performed in ADQM Control;
  - ***Alerts retention*** — alerts about problems detected in the ADQM cluster.

Parameter values that configure the data storage time are specified in days, in the range from 1 to 365, inclusive.

Once all parameters have been configured, click ***Save*** to save the settings. The ***Revert all*** button discards all changes made (not yet saved by clicking ***Save***).

Configuration parameters

Contents

ADQM Control

ADPG

Monitoring

To Table of Contents

This section describes parameters that are configured for ADQM Control service via ADCM. To see about the configuration process refer to the [Configuration](#) article in the section **Online installation of ADQM**.

ADCM

Some of the parameters become visible in the ADCM UI after the **Show advanced** flag has been set.

ADQM Control

ADQM management

Parameter	Description	Default value
ADQM username	Username to connect to ADQM	arenadata
ADQM password	Password for the **ADQM username** user to connect to ADQM. Should contain at least 8 characters, including at least one digit, one lowercase letter, one uppercase letter, and one special character	
ADQM port	Default port of the native interface of ADQM	9000
ADQM secure port	Default port for TSL secured communications with ADQM (it is used **ADQM username** and **ADQM password**)	9440
Connection type	Type of connection to ADQM. Available values: **Plaintext**, **TSS**, **Mutual TLS**, **Custom CERT validation**	Plaintext (TCP)

NOTE

If you set the **ADQM username**, the **ADQM port**, and **ADQM password** parameters during a run and ensure that a node in ADH for the ADQM cluster is available, then ADQM Control will create a user with the specified parameters on the ADQM cluster. If you do not set these parameters, then ADQM Control will create a user with the default parameters. Also, you can create a user manually on ADQM cluster nodes:

```
CREATE USER arenadata IDENTIFIED WITH sha256_password BY 'password';

GRANT ON CLUSTER default_cluster ALL WITH GRANT OPTION TO arenadata;
```

Zookeeper

Parameter	Description	Default value
Session timeout	ADQM Control side interface session timeout (in seconds)	1800
Enable CRT validation	Enables CRT validation	Enabled

Idle configuration

NOTE

For details on how to configure and manage ADQM Control, see the [Configuration](#) article.

Parameter	Description	Default value
[ADQM Control's Backend] – Enable SSL	Enables SSL for the Backend component	Disabled
[ADQM Control's ADCM Agent] – Enable SSL	Enables SSL for the Agents component	Disabled
[ADQM Control's Web Frontend] – Enable SSL	Enables SSL for the Web Frontend component	Disabled
[ADQM Control's Web Frontend] – Enable HSTS	Enables SSL for the Web Frontend component	Disabled
[ADQM Control] – Create SSL	Enables SSL for the Web Frontend and Backend components	Disabled
[ADQM Control Services] – Enable SSL for Backend	Path to the file with the private key of the server SSL certificate for the Backend component. Used for Backend's Agents client	/etc/adqmcluster/ssl/server.key
[ADQM Control Services] – Certificate for Backend	Path to the server SSL certificate for the Backend component	/etc/adqmcluster/ssl/server.crt
[ADQM Control] – Certificate for Agents	Path to the server SSL certificate for the Agents component	/etc/adqmcluster/ssl/server.crt
[ADQM Control] – Private key for Agents	Path to the file with the private key of the server SSL certificate for the Agents component. Used for Agents' client	/etc/adqmcluster/ssl/server.key
[ADQM Control] – Certificate authority for Agents	Path to the file with the SSL certificate authority for the ADCM Agents component	/etc/adqmcluster/ssl/ca.crt

Network configuration

Parameter	Description	Default value
Backend secret	Port to access the ADQM Control side interface	8000
Alert Manager port	Port for Alert Manager	9021
ADQM agent port	Port the Agents service listens on	8002
ADQM exporter port	Port the admin manager listens on	9005
Alert Manager ext port	Port the alert receiver listens on	10000
Prometheus port	Prometheus connection timeout (in seconds)	9090
Alert manager connection timeout	Alert manager connection timeout (in seconds)	30

Connection settings

Parameter	Description	Default value
Backend connection timeout	Maximum number of connections to PostgreSQL database of the Backend component	3
ADQM agent connection timeout	Maximum number of connections to PostgreSQL database of the Agents component	3
Statement timeout	Timeout for queries and transactions (in seconds)	120
Backend connection pool size	Maximum number of connections to the database pool that runs longer than the specified time	30000
Migration timeout	Database migration timeout (in seconds)	9000
Prometheus timeout	Prometheus connection timeout (in seconds)	30
Alert manager connection timeout	Alert manager connection timeout (in seconds)	30

Polling rate

Parameter	Description	Default value
Alerts manager interval	Frequency of updating information about alerts in the ADQM cluster (in seconds)	30
ADQM/ZK pings interval	Frequency of updating information about states in the ADQM cluster (in seconds)	30
Query update interval	Frequency of collecting queries (in seconds)	120
Query log interval	Frequency of updating queries (in seconds)	120
Check query log interval	Frequency of checking **Query log** (in seconds)	60

Limits

Parameter	Description	Default value
Maximum query size now queries per minute	Maximum number of queries executed per minute	10000
Collect queries limit	Maximum number of queries collected from the **system.query_log** table	10000
Query concepts threshold	Number of attempts to run a query	10
Counting tables rows threshold	Maximum number of rows in the table for which the number of rows is counted. Available values: **Always**, **Never**, **Only if system.tables.total_rows is known**, **Only if counting is faster than**	0
Tables size limit	Maximum number of tables for which the size is counted	21700
Table columns metadata retrieval	Maximum number of tables for which the columns are available (in seconds)	3600
Maximum session per table to cache	Maximum number of sessions per table to the cache	100
Queries cache size	Size of the query cache	5120
Queries cache TTL	Query cache TTL (in seconds)	3600
Capacity of working tables cache	Capacity of the working tables cache	50000000
Query queries cache number to evict	Amount of the working table cache to be evicted (in seconds)	10000
Maximum number of the working tables cache to be evicted	Maximum number of working tables cache for the working table cache to be evicted	1000000
Maximum of parallel workers	Maximum number of parallel workers	8

Control PostgreSQL server

NOTE

To access the configuration parameters, activate the **Use external PostgreSQL server** option.

Parameter	Description	Default value
server_ip_port	Host and port to access the PostgreSQL server in the format **<host>:<port>**	
db-is-discrete	Database name to be used by the ADCM Control service	arenadata
db-user-name	Name of a user with access to the database specified in the **db-is-discrete** parameter	arenadata
db-user-password	Password for the **db-user-name** user to access the database specified in the **db-is-discrete** parameter. Should contain at least 8 characters, including at least one digit, one lowercase letter, one uppercase letter, and one special character	
SSL to use by an external server	Specifies whether SSL is used by an external server	Disabled

ADPG

Parameter	Description	Default value
data directory	Directories used to store data on ADPG hosts	/pg_data/

ADPG configurations

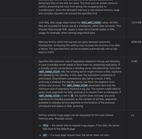
NOTE

You can edit the entire parameter values in archive files using the **Collect recommendations** and **Apply** buttons of the ADPG service.

Parameter	Description	Default value
listen_addresses	IP addresses on which the server listens for connections from client applications	*
port	TCP port the server listens on	5432
max_connections	Maximum number of concurrent connections to the database server	100
shared_buffers	Memory size for shared memory buffers used by the database server. Active if reasonable tuning values are not known for shared buffers	25MB
max_prepared_transactions	Maximum number of transactions that can be in the prepared state	0
work_mem	Maximum amount of memory to be used by a query operation (in kilobytes)	4MB
max_parallel_workers_per_gather	Maximum number of workers that can be started by a single Gather or Gather Merge node	0
max_parallel_maintenance_workers	Maximum number of parallel workers that can be started by a single utility command	2
effective_cache_size	Sets the ADPG query planner's assumption about the effective size of the disk cache that is available to a single query. The bigger the value, the more likely the index scans will be used, the smaller the value, the more likely the sequential scans will be used. Active if reasonable tuning values are not known for effective_cache_size	4 GB
maintenance_work_mem	Maximum size for maintenance operations (such as VACUUM, CREATE INDEX, and ALTER TABLE ADD FOREIGN KEY)	1024 MB
work_mem	Base maximum memory that used for internal operations when the server executes complex queries (such as sorts or hash tables) before writing to temporary disk files. Active if reasonable tuning values are not known for work_mem	32 MB
min_wal_size	Until WAL disk usage stays below **WAL MIN SIZE**, old WAL files are always recycled for future use at a checkpoint, rather than removed. This can be used to ensure that enough WAL space is reserved to handle spikes in WAL usage, for example when running large batch jobs. Active if reasonable tuning values are not known for min_wal_size	1024 MB
max_wal_size	Maximum WAL size that triggers a checkpoint. If this limit is exceeded when a checkpoint is begun, the system will request a checkpoint be performed at the earliest opportunity. The soft limit is not guaranteed to be enforced in all cases. Active if reasonable tuning values are not known for max_wal_size	4096 MB
wal_keep_size	Specifies the minimum size of past WAL files kept in the WAL directory, in case a standby server needs to fetch them for streaming replication. If a standby server connected to the sending server falls behind by more than **WAL KEEP SIZE** megabytes, the sending server might remove a WAL segment still needed by the standby server, in which case the replication connection will be terminated. Downstream connections will also eventually fail. Active if reasonable tuning values are not known for wal_keep_size. In the context of **WAL KEEP SIZE**, one unit is one megabyte. With the default value of 0, no WAL is reserved. The value -1 means that the maximum size is taken from the value of **max_wal_size**	0 MB
huge_pages	Allows ADPG to request large pages for the main shared memory area. Possible values: **off** – huge pages are not used; **on** – huge pages are used; **try** – the server will try to request huge pages, but start anyway if this fails	try
logging_collector	Enables the built-in logging collector. Active if reasonable tuning values are not known for logging_collector	on
log_directory	Determines the common log files. It can be specified as an absolute path or relative to the cluster data directory	log (the absolute path is /pg_data/adpg/log)
log_filename	Specifies the pattern of log file names. The system log time is used to compute the actual file name if the pattern includes strftime escapes	postgresql-%a.log
log_rotation_age	Maximum lifetime of a log file. After this time has elapsed, a new log file will be created. Set to 0 to disable time-based log file rotation	1d
log_rotation_size	Maximum size of a log file. When a log file reaches the specified size, a new log file will be created. Set to 0 to disable size-based log file rotation	10MB
log_min_messages	Minimum severity of log messages that are written to the server log. Possible values: **DEBUG5**, **DEBUG4**, **DEBUG3**, **DEBUG2**, **DEBUG1**, **INFO**, **NOTICE**, **WARNING**, **ERROR**, **LOG**, **FATAL**, **PANIC**. Active if reasonable tuning values are not known for log_min_messages	warning
log_min_error_statement	Minimum severity of error messages that are written to the server log. Possible values: **DEBUG5**, **DEBUG4**, **DEBUG3**, **DEBUG2**, **DEBUG1**, **INFO**, **NOTICE**, **WARNING**, **ERROR**, **LOG**, **FATAL**, **PANIC**. Active if reasonable tuning values are not known for log_min_error_statement	error

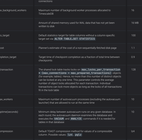
Custom postgres.conf

In the **Custom postgres.conf** section, you can set additional parameters to be written to the [postgres.conf](#) file. To add a parameter, click **Add property**.



Custom postgres.conf section

In the window that appears, enter a parameter key and value and click **Save**.

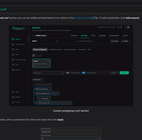


Window to add a parameter

You can edit multiple parameters.

Pool of Connect Blocks

Parameters of the connection pool (**Connect Blocks**) are configured in the **Pool of Connect Blocks** section. The **Connect limit**, **Connect timeout**, and **Connect retry interval** parameters are set by default.



The "Pool of Connect Blocks" section

Control log files conf

The listed above you will need to edit the [log_control](#) file that contains client authentication.

SSL configuration

NOTE

For more information on SSL encryption support in ADQM, see the [SSL](#) article.

Parameter	Description	Default value
Enable SSL	Enables SSL encryption of incoming traffic for the ADPG server	Disabled
Certificate file	Path to the server SSL certificate file	/etc/adpg/ssl/server.crt
Private key file	Path to the file with the private key of the server SSL certificate	/etc/adpg/ssl/server.key
Certificate authority file	Path to the CA certificate file	/etc/adpg/ssl/ca.crt

Monitoring

Prometheus settings

Parameter	Description	Default value
Scrape interval	Time how frequently to scrape targets	15s
Scrape timeout	Specifies how long to wait until a scrape request times out	10s
Password for pushing metrics	Password of a shadow user to connect to Prometheus	-
Prometheus is used to monitor Arenadata Postgres	User credentials for logging into the Prometheus web interface	-
Service parameters	Prometheus parameters	-

Grafana settings

Parameter	Description	Default value
Grafana administrator password	Password of a Grafana administrator user	-
Grafana server port	Port to access the Grafana web interface	11731

Node Exporter settings

Parameter	Description	Default value
Listen port	Port to listen for a host's system metrics in the Prometheus format	11100
Metrics to expand	Endpoint to receive system metrics	/metrics

SSL configuration

Parameter	Description	Default value
[Prometheus] – Create SSL	Enables SSL encryption of incoming traffic for the Prometheus Server component	Disabled
[Prometheus] – Certificate file	Path to the server SSL certificate file for the Prometheus Server component	/etc/adh-prometheus/prometheus.server.crt
[Prometheus] – Certificate authority file	Path to the file with the SSL certificate authority for the Prometheus Server component	/etc/adh-prometheus/prometheus.ca.crt
[Grafana] – Certificate authority file	Path to the file with the SSL certificate authority for the Grafana component	/etc/adh-grafana/grafana.ca.crt
[Grafana] – Create SSL	Enables SSL encryption of incoming traffic for the Grafana component	Disabled
[Grafana] – Private key file	Path to the file with the SSL certificate for the Grafana component	/etc/adh-grafana/grafana.server.key
[Grafana] – Certificate authority file	Path to the file with the SSL certificate authority for the Grafana component	/etc/adh-grafana/grafana.ca.crt
[Node exporter] – Create SSL	Enables SSL encryption of incoming traffic for the Node Exporter component	Disabled
[Node exporter] – Certificate file	Path to the server SSL certificate file for the Node Exporter component	/etc/node-exporter/node_exporter.crt
[Node exporter] – Private key file	Path to the file with the private key of the server SSL certificate for the Node Exporter component	/etc/node-exporter/node_exporter.key

Query navigation codes

Contents

Query Processing
Symbols
Resource Governing
Physical Operations
Data Stealth
Data Shaping
Networking
Distributed Execution
Core Storage
System Governance
Integrations
Access Control
Features

To Table of Contents



The Process Icon will open this list for ease.

Syntax Processing

Control how your query is interpreted and executed.

Syntax and grammar

Control code	Purpose
CMD0	SIMPLE_QUERY
CMD1	COMPLEX_QUERY
CMD2	SCRIPT_QUERY
CMD3	SCRIPT_LOGIC
CMD4	CONTROL_FLOW_ONLY
CMD5	CONTROL_FLOW_PRO

API versions and limits

Control code	Purpose
CMD6	ADVANCED_SCRIPT_ONLY
CMD7	FULL_API_OFF
CMD8	FULL_API_ON
CMD9	SCRIPT_FULL_APPS

Specific query parsing

Control code	Purpose
CMD10	SIMPLE_INSERT_SYS
CMD11	SIMPLE_INSERT_SYS_PRO
CMD12	SIMPLE_INSERT_SCRIPT
CMD13	SIMPLE_INSERT_SCRIPT_PRO
CMD14	SIMPLE_INSERT_SYSTEM

Symbols

Control how symbols are used.

- Symbols are the names of objects in the query, such as the column name or table.
- Symbols can be overloaded, which means that the same symbol can refer to different objects in different contexts. For example, the symbol "name" can refer to a column in a table, or it can refer to a variable in a script.
- Symbols can be used to create aliases, which are alternative names for objects. Aliases can be used to make queries more readable or to avoid conflicts between symbols.

Identifier and naming

Control code	Purpose
CMD15	IN_DOUBT_DEFAULTS_TEXT
CMD16	UNIVERSAL_QUICKLY
CMD17	UNIVERSAL_SLOW
CMD18	UNIVERSAL_EFFICIENT
CMD19	UNIVERSAL_BEST_FIT
CMD20	AMBIGUOUS_QUESTION
CMD21	AMBIGUOUS_CLARIFICATION

Functions and operators

Control code	Purpose
CMD22	FULL_SUPPORT
CMD23	ENABLED_BY_ADVANCED_CLIENT_SAFETY
CMD24	ADVANCED_FUNCTIONS
CMD25	UNIVERSAL_ADVANCED_FUNCTIONS
CMD26	ADVANCED_FUNCTIONS_PRO
CMD27	ENCRYPTED_FUNCTIONS_PROTECTED

Type system and type hints

Control code	Purpose
CMD28	UNIVERSAL_LOW
CMD29	UNIVERSAL_MEDIUM
CMD30	UNIVERSAL_HIGH_TYPE
CMD31	UNIVERSAL_FULL_AI_ASSIST
CMD32	UNIVERSAL_API
CMD33	UNIVERSAL_API_PRO

Alias, abbreviation and autocomplete

Control code	Purpose
CMD34	UNIVERSAL_ONLINE
CMD35	FULL_AUTO_COMPLETE
CMD36	FULL_AUTO_COMPLETE_PRO
CMD37	UNIVERSAL_API
CMD38	UNIVERSAL_API_PRO

Runtime Operations

Control how your query is executed, including its resource consumption.

- Runtime operations are the actions that are performed when a query is executed.
- Runtime operations can be used to control the flow of execution, to access data, and to perform calculations.
- Runtime operations can be affected by a variety of factors, including the size of the data set, the complexity of the query, and the available resources.

Arithmetic and logic

Control code	Purpose
CMD39	SIMPLE_MATH
CMD40	FULL_MATH_LOGIC
CMD41	ADVANCED_FUNCTION_PARSE

Runtime operations

Control code	Purpose
CMD42	SIMPLE_QUERY_ALL_AUTOMATIC_CHECK
CMD43	SIMPLE_QUERY_ONLY
CMD44	SIMPLE_QUERY_ONLY_PRO
CMD45	API_CONTROL_BLOCKED

Resource Governing

Ensure queries run predictably and don't overwhelm shared resources.

- Resource governing is the process of controlling the amount of resources that a query can consume.
- Resource governing can be used to prevent queries from consuming too much memory, CPU time, or other resources.
- Resource governing can be used to improve the performance of queries by ensuring that they have the resources they need to run efficiently.

Memory and disk space

Control code	Purpose
CMD46	SIMPLE_QUERY_MEMORY
CMD47	SIMPLE_QUERY_DISK
CMD48	SIMPLE_QUERY_MEM_DISK

Query lifecycle and timeouts

Control code	Purpose
CMD49	SIMPLE_QUERY_TIMEOUT
CMD50	FULL_CORE

Error codes and exceptions

Control code	Purpose
CMD51	FULL_ERROR_CODE
CMD52	FULL_QUERY_INFO
CMD53	FULL_QUERY_INFO_PRO

Concurrency, queuing, and prioritization

Control code	Purpose
CMD54	QUEUE_MANAGEMENT
CMD55	QUEUE_STATUS
CMD56	FULL_QUERY_PRIORITY_QUEUE
CMD57	FULL_TIMEOUT_QUEUE
CMD58	QUEUE_CLEANUP
CMD59	QUEUE_STATUS_FULL

Query rate and IO suppression

Control code	Purpose
CMD60	SIMPLE_QUERY_IO
CMD61	FULL_QUERY_IO_SPEED
CMD62	QUERY_IO_FULL
CMD63	FULL_QUERY_IO_SPEED_CHECK

Physical Operations

Control how your query is executed, including its resource consumption.

- Physical operations are the actions that are performed when a query is executed.
- Physical operations can be used to control the flow of execution, to access data, and to perform calculations.
- Physical operations can be affected by a variety of factors, including the size of the data set, the complexity of the query, and the available resources.

Scan/Read/Seek operations

Control code	Purpose
CMD64	SIMPLE_QUERY_FULL_SCAN
CMD65	SPECIAL_FLAGGED_TO_SCAN
CMD66	FULL_SCAN_ONLY
CMD67	SIMPLE_QUERY_SEEK
CMD68	SIMPLE_QUERY_SEEK_PRO

Join algorithms and management

Control code	Purpose
CMD69	SIMPLE_SPECIAL
CMD70	SIMPLE_QUERY_LEFT_OUTER_JOIN
CMD71	FULL_HASH_JOIN
CMD72	SIMPLE_QUERY_MERGE_JOIN
CMD73	FULL_LOOP_JOIN
CMD74	SIMPLE_QUERY_FULL_OUTER

Sorting, aggregation, and materialization

Control code	Purpose
CMD75	SIMPLE_SORT
CMD76	FULL_SORT_AND_GROUP
CMD77	FULL_SORT_AND_GROUP_PRO

Data modification operations

Control code	Purpose
CMD78	SIMPLE_INSERT
CMD79	SIMPLE_UPDATE
CMD80	SIMPLE_DELETE
CMD81	SIMPLE_MERGE

Parallelism and distribution

Control code	Purpose
CMD82	SIMPLE_PARALLEL
CMD83	SIMPLE_PARALLEL_PRO
CMD84	SIMPLE_DISTRIBUTED
CMD85	SIMPLE_DISTRIBUTED_PRO

Data Stealth

Control how your query is executed, including its resource consumption.

- Data stealth is the process of hiding data from unauthorized users.
- Data stealth can be used to protect sensitive data, such as personal information or financial data.
- Data stealth can be used to comply with regulations, such as the General Data Protection Regulation (GDPR).

Encryption and decryption

Control code	Purpose
CMD86	ENCRYPTION_ON
CMD87	ENCRYPTION_OFF
CMD88	ENCRYPTION_LEVEL_SET

Column, partition & row security

Control code	Purpose
CMD89	SECURITY_CHANGE_TO_COLUMN
CMD90	ADD_AT_COLUMN_SECURITY
CMD91	DROP_AT_COLUMN_SECURITY
CMD92	SECURITY_CHANGE_TO_ROW
CMD93	ADD_AT_ROW_COLUMN_SECURITY
CMD94	DROP_AT_ROW_COLUMN_SECURITY

Masking and redaction

Control code	Purpose
CMD95	MASKING_ON
CMD96	MASKING_OFF
CMD97	MASKING_LEVEL_SET

Tokenization

Control code	Purpose
CMD98	TOKENIZATION_ON
CMD99	TOKENIZATION_OFF
CMD100	TOKENIZATION_LEVEL_SET

Data Shaping

Control how your query is shaped for your needs.

Compression algorithms

Control code	Purpose
CMD101	UNIVERSAL_COMPRESSION_ART_LEVEL
CMD102	UNIVERSAL_COMPRESS
CMD103	UNIVERSAL_DECOMPRESS
CMD104	UNIVERSAL_DECOMPRESS_PRO
CMD105	UNIVERSAL_COMPRESS_PRO
CMD106	UNIVERSAL_COMPRESSION_ART_LEVEL_PRO
CMD107	UNIVERSAL_COMPRESSION_ART_LEVEL_PRO_PRO

Specific formats

Control code	Purpose
CMD108	SIMPLE_PARSE_PROTOCOL_BUFFERS
CMD109	SIMPLE_PARSE_AVRO
CMD110	SIMPLE_PARSE_APACHE_PARQUET
CMD111	SIMPLE_PARSE_ORC
CMD112	SIMPLE_PARSE_THRIFT
CMD113	SIMPLE_PARSE_MSGPACK

Networking

Control how your query is executed, including its resource consumption.

- Networking is the process of connecting computers and other devices together.
- Networking can be used to share data, resources, and applications.
- Networking can be used to improve the performance of queries by reducing the amount of data that needs to be transferred.

Connection and session

Control code	Purpose
CMD114	SIMPLE_SESSIONS_TO_QUERY
CMD115	API_CHECK
CMD116	QUERY_TIME
CMD117	API_CHECK_PRO
CMD118	SIMPLE_SESSIONS_TIME_PRO

Protocol and transfer

Control code	Purpose
CMD119	QUERY_TIME_PRO
CMD120	QUERY_TIME_PRO_PRO

Distributed Execution

Control how your query is executed, including its resource consumption.

Coordination

Control code	Purpose
CMD121	API_CHECK_PRO
CMD122	QUERY_TIME_PRO_PRO_PRO
CMD123	QUERY_TIME_PRO_PRO_PRO_PRO
CMD124	API_CHECK
CMD125	QUERY_TIME

Replication and management

Control code	Purpose
CMD126	SIMPLE_SESSIONS_TO_QUERY
CMD127	API_CHECK_PRO
CMD128	SIMPLE_SESSIONS_TIME_PRO
CMD129	API_CHECK_PRO_PRO
CMD130	QUERY_TIME_PRO_PRO_PRO

Network and partitioning

Control code	Purpose
CMD131	QUERY_TIME_PRO_PRO_PRO_PRO

Core Storage

Control how your query is executed, including its resource consumption.

- Core storage is the process of storing data on disk or in memory.
- Core storage can be used to improve the performance of queries by reducing the amount of data that needs to be read from disk or memory.
- Core storage can be used to protect data from loss or corruption.

Data and management

Control code	Purpose
CMD132	FULL_ELEMENTARY_TO_ELEVATED_API
CMD133	FULL_ELEVATED_API
CMD134	ELEVATED_API_ONLY

Metadata and structure

Control code	Purpose
CMD135	SIMPLE_SPECIAL_TO_ALL
CMD136	SIMPLE_SPECIAL_TO_ALL_PRO
CMD137	FULL_ALL_SPECIAL
CMD138	FULL_ALL_SPECIAL_PRO

Caching and background operations

Control code	Purpose
CMD139	SIMPLE_QUERY_CACHE
CMD140	SIMPLE_QUERY_CACHE_PRO
CMD141	SIMPLE_QUERY_CACHE_PRO_PRO

System Governance

Control how your query is executed, including its resource consumption.

- System governance is the process of managing the resources that are available to a system.
- System governance can be used to ensure that resources are used efficiently and that they are available when they are needed.
- System governance can be used to protect data from loss or corruption.

API, applications

Control code	Purpose
CMD142	FULL_QUERY_ACCESS
CMD143	SIMPLE_QUERY_ACCESS
CMD144	SIMPLE_QUERY_ACCESS_PRO
CMD145	SIMPLE_QUERY_ACCESS_PRO_PRO
CMD146	FULL_QUERY_ACCESS_PRO_PRO
CMD147	FULL_QUERY_ACCESS_PRO_PRO_PRO

Configuration and schema

Control code	Purpose
CMD148	UNIVERSAL_ART_FULL
CMD149	UNIVERSAL_ART_ALL
CMD150	UNIVERSAL_ART_ALL_PRO
CMD151	UNIVERSAL_ART_ALL_PRO_PRO

Permissions

Control code	Purpose
CMD152	UNIVERSAL_ART_ALL_PRO_PRO_PRO
CMD153	ULTIMATE_ACCESS_ALL
CMD154	ULTIMATE_ACCESS_ALL_PRO

Access Control

Control how your query is executed, including its resource consumption.

- Access control is the process of controlling who can access a system or data.
- Access control can be used to protect data from unauthorized access, modification, or destruction.
- Access control can be used to comply with regulations, such as the General Data Protection Regulation (GDPR).

Authentication

Control code	Purpose
CMD155	UNIVERSAL_LOW
CMD156	MEDIUM_MEDIUM
CMD157	FULL_MEDIUM_HIGH
CMD158	ULTIMATE_FULL_AUTH
CMD159	LAST_RESORT
CMD160	ESCALATE_QUERY
CMD161	ESCALATE_QUERY_PRO

Authorization

Control code	Purpose
CMD162	ULTIMATE_ACCESS_QUERY
CMD163	ULTIMATE_ACCESS_QUERY_PRO
CMD164	FULL_ACCESS_QUERY
CMD165	ULTIMATE_ACCESS_QUERY_PRO_PRO

Encryption

Control code	Purpose
CMD166	UNIVERSAL_ADMIN
CMD167	ADVANCED_ENCRYPTION_ON
CMD168	LOCK_DOWN

Integrations

Control how your query is executed, including its resource consumption.

Client and driver support

Control code	Purpose
CMD169	API_ACCESS
CMD170	ADVANCED_CONTROL_APPLY
CMD171	API_APPLY
CMD172	API_ADMIN
CMD173	ADVANCED_API_ADMIN

Extensions and protocols

Control code	Purpose
CMD174	ADVANCED_API_ACCESS
CMD175	ADVANCED_CONNECTION_ONLINE
CMD176	ADVANCED_CONNECTION_ONLINE_PRO
CMD177	ADVANCED_ADMIN
CMD178	ADVANCED_ADMIN_PRO

Storage system and I/O

Control code	Purpose
CMD179	SIMPLE_QUERY_ACCESS
CMD180	SIMPLE_QUERY_ACCESS_PRO

Features

Control how your query is executed, including its resource consumption.

Working and research

Control code	Purpose
CMD181	ADVANCED_ADMIN_ACCESS
CMD182	ADVANCED_ADMIN_ACCESS_PRO
CMD183	ADVANCED_ADMIN_ACCESS_PRO_PRO

Analytics

Control code	Purpose
CMD184	ADVANCED_ADMIN_ACCESS_PRO_PRO_PRO
CMD185	FULL_ADMIN_ACCESS
CMD186	FULL_ADMIN_ACCESS_PRO
CMD187	FULL_ADMIN_ACCESS_PRO_PRO
CMD188	FULL_ADMIN_ACCESS_PRO_PRO_PRO

Transaction

Control code	Purpose
CMD189	SIMPLE_TRANSACTION
CMD190	ADVANCED_SIMPLE_TRANSACTION

Machine learning

Control code	Purpose
CMD191	SIMPLE_QUERY_ACCESS_PRO_PRO_PRO

System Stress

Control how your query is executed, including its resource consumption.

Limits and soft/hard stops

Control code	Purpose
CMD192	ADVANCED_API_ACCESS
CMD193	API_ADMIN_PRO
CMD194	SIMPLE_QUERY
CMD195	SIMPLE_QUERY_PRO

API calls

Control code	Purpose
CMD196	SIMPLE_QUERY_PRO
CMD197	SIMPLE_QUERY_PRO_PRO
CMD198	QUERY_TIMEOUT
CMD199	SYSTEM_QUERY
CMD200	SYSTEM_QUERY_PRO
CMD201	QUERY_ABORT
CMD202	QUERY_ABORT_PRO

Session disconnect

Control code	Purpose
CMD203	SIMPLE_QUERY_ACCESS
CMD204	SIMPLE_QUERY_ACCESS_PRO
CMD205	SIMPLE_QUERY_ACCESS_PRO_PRO
CMD206	SIMPLE_QUERY_ACCESS_PRO_PRO_PRO
CMD207	SIMPLE_QUERY_ACCESS_PRO_PRO_PRO_PRO
CMD208	SIMPLE_QUERY_ACCESS_PRO_PRO_PRO_PRO_PRO

ADQM Control monitoring metrics

Elena Dvoryadkina

Contents

To Table of Contents

System metrics

ADQM Control service metrics

The Monitoring service installed in an ADQM Control cluster collects two groups of metrics: system metrics from hosts and ADQM Control service metrics.

You can view metrics in your browser in the Prometheus format (ports and endpoints for access to metrics are described below), as well as use the Prometheus and Grafana web interfaces.

System metrics

System metrics indicate general characteristics of cluster hosts, usually related to resource consumption: for example, CPU utilization, disk capacity, memory usage, I/O performance, and other parameters.

To view system metrics of ADQM Control hosts in the Prometheus format, enter in the address bar of your browser: *http://<adqmc_host_ip>:11203/metrics*, where `11203` and `/metrics` are a port number and endpoint to listen for system metrics, which are set up in the ***Node Exporter settings*** section of the Monitoring service configuration.

In the Grafana interface, system metrics for ADQM Control hosts are displayed in the ***System*** dashboard and in the ***System*** row of the ***General*** dashboard.

The ***General*** dashboard also contains the ***Database*** section with the ***Postgres requests duration*** and ***Postgres requests duration per service*** graphs based on the `postgres_client_duration_seconds_bucket` metric — duration of queries to the PostgreSQL database.

The General dashboard in Grafana



The General dashboard in Grafana UI

ADQM Control service metrics

ADQM Control service metrics allow you to monitor and analyze various parameters of ADQM Control operation and performance: for example, availability of service components, types and frequency of errors, response time for user requests, frequency of alert generation and alert types, and other metrics.

Use the addresses listed in the table below to explore monitoring metrics of the ADQM Control service components in the Prometheus format. Provided addresses contain default ports for accessing metrics of service components. You can change these ports in the ***Network configuration*** section of the ADQM Control service configuration.

ADQM Control service component	Address to access component metrics
Agents	http://<host_ip>:5002/api/v1/metrics
Alert Generator	http://<host_ip>:5001/api/v1/metrics
Alert Receiver	http://<host_ip>:12322/api/v1/metrics
Alertmanager	http://<host_ip>:9093/metrics
Backend	http://<host_ip>:5555/api/v1/metrics

The tables below describe the ADQM Control service metrics, grouped as they are presented on dashboards in the Grafana interface.

Adqm_agent

Metrics of the Agents component show which information about ADQM clusters is collected and stored by ADQM Control.

Metric name	Description
adqm_agent_hosts	Number of ADQM cluster hosts
adqm_agent_databases	Number of ADQM databases
adqm_agent_tables	Number of ADQM tables
adqm_agent_columns	Number of columns in ADQM tables
adqm_agent_queries	Number of queries run in ADQM
adqm_agent_queries_normalized	Number of normalized (linked to tables) queries
jobs_total	Number of jobs performed by a service. Job statuses: `success`, `failed`. Job types: `alerts_cleanup`, `check_query_log`, `hosts_collection`, `job_log_cleanup`, `queries_collection`, `queries_duration_aggregation`, `queries_duration_collection`, `queries_normalization`, `tables_collection`
adqm_agent_queries_unnormalizable	Number of queries that cannot be normalized
adqm_agent_queries_unnormalized	Number of unnormalized queries
adqm_agent_queries_queue_size	Size of the current queue of queries for normalization
adqm_agent_normalization_cache_hit_ratio	Coefficient of hitting normalization tables in the cache
adqmc_database_corruption	Indicates issues with the consistency of the ADQM Control database (the ***Missing**** panels at the bottom of the ***Adqm_agent*** dashboard)



The Adqm_agent dashboard in Grafana UI

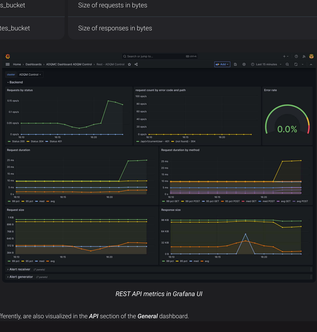
Alerts

Metric name	Description
alertgenerator_alerts_lifetime_bucket	Alert lifetime (the period between the time an alert is sent and the time the alert is no longer considered valid)
alertgenerator_alerts_total	Number of generated alerts
alertgenerator_fired_alerts_total	Number of alerts sent to Alert Manager
alertreceiver_alerts_received_total	Number of alerts received by Alert Receiver
alertgenerator_alerts_resend_total	Number of alerts resent to Alert Manager

Rest

The ***Rest*** dashboard contains the ***Backend***, ***Alert receiver***, and ***Alert generator*** rows, where panels display the following metrics for the corresponding service components.

Metric name	Description
http_request_count_total	Number of requests
http_request_duration_seconds_bucket	Duration of requests in seconds
http_request_size_bytes_bucket	Size of requests in bytes
http_response_size_bytes_bucket	Size of responses in bytes



REST API metrics in Grafana UI

REST API metrics, grouped differently, are also visualized in the ***API*** section of the ***General*** dashboard.

ADQM Control releases

Elena Dvoryadkina

Contents

To Table of Contents

1.3.1
1.3.0
1.2.1
1.2.0
1.1.1
1.1.0
1.0.0

1.3.1

1.3.1.b2

Date: 01.09.2026

Bug fixes

- Fixed the definition of `adpg_datadir` (the current value from ADPG is now used instead of the default one of `adpg_pg_data`)

1.3.1.b1

Date: 17.07.2026

Bug fixes

- Fixed an issue with 6-hour buckets on query charts
- Fixed an issue where an alert about the ADQMDB service unavailability could be triggered on a host without ADQMDB installed

1.3.0

Date: 07.07.2026

New features

- Introduced the Host graph on the ***Queries*** page — distribution of queries across cluster hosts with anomaly detection
- Introduced table groups (aggregated tables)
- Added alerts about host and ADQMDB service unavailability
- Added data cleanup/retention settings
- Added the aggregated view to the Top 10 tables dashboard section
- Added a modal window with graphs for top 10 tables on the ***Dashboard*** page
- Added information about the lifetime of tables
- Added graphs related to request count for tables
- Added support for RHEL 9
- Added support for RED OS 8.0 Certified edition

Improvements

- Added the ability to select the level of detail (`Low`/`High`) for graphs by tables and queries
- Moved alert cleanup settings to the ***Settings*** → ***Cleanup*** page
- Added support for non-UUID query IDs

Bug fixes

- Corrected the alert cooldown period check
- Long-running migration now receives a stop signal on cancellation instead of leaking resources
- Fixed an error related to stopping the collection of ADQMDB metrics when Prometheus was unavailable while the host was still in the cluster

Misc/Internal

- Query monitoring now tracks rows and bytes processed per query and per aggregated bucket
- Added basic authentication support for adqmc-alert-receiver
- Bundle contract version is now 2.1
- ADQM cluster topologies can now be determined via ZooKeeper as an alternative to Prometheus
- Implemented the `pull/push` option for the `job-log` command of ADQM Control Maintenance CLI
- Removed C++ services
- Performed internal refactoring
- Minimum version of ADQMDB is now 25.3.2.39

1.2.1

1.2.1.b2

Date: 18.03.2026

New features

- Added support for Astra Linux 1.8.4 Voronezh
- Added support for Astra Linux 1.8.4 Orel

Bug fixes

- Fixed an issue related to Monitoring when ADQM Control was not working with enabled SSL

1.2.1.b1

Date: 02.03.2026

Improvements

- Added the ability to access the ADQM Notebook web interface from the Dashboard and Tables pages
- Large numbers (such as the number of data rows in a table and the number of queries sent to a table) are now displayed in a human readable format

1.2.0

Date: 04.02.2026

Improvements

- Updated permissions in the role-based access model of ADQM Control
- Added support for filtering queries by multiple exception codes
- Added support for filtering queries by exception code presets
- Implemented various backend and frontend optimizations

Misc/Internal

- Introduced ADQM Control Maintenance CLI
- Removed the Zookeeper service

1.1.1

Date: 10.09.2025

New features

- Added support for Alt Linux 10.2
- Added support for Astra Linux 1.7.6 Voronezh

Misc/Internal

- Updated submodules
- Increased the minimum version of ADCM to 2.7.1

1.1.0

Date: 24.07.2025

New features

- Introduced the ***Duration***, ***Kind***, and ***User*** graphs for queries
- Introduced incidents
- Added support for SSL in all services of ADQM Control
- Added the ***Comment*** field for ADQM table columns on the Table details page
- Implemented a new alert on missing the `system.query_log` table
- Added the threshold label to ADQM Control alerts
- Implemented acknowledgement for alerts
- Added multi-user, query duration, query text, and exception filters on the ***Queries*** page
- Added search filters on the Users page
- Extended ***Heat map*** with statistics information for each host
- Refactored paginator on the ***Queries*** page by removing the page counter

1.0.0

Date: 17.12.2024

New features

- Introduced ADQM Control services, including Agents, Alert generator, Alert manager, Alert receiver, and Backend
- Introduced basic functionality for managing users within ADQM Control
- Introduced the Dashboard page providing an overview of the physical cluster state, alerts, top 10 tables, and top 10 queries
- Introduced the Tables page for displaying table information over a specified time window
- Introduced functionality for viewing detailed information about columns within tables over a specified time window
- Introduced charts for tracking table metrics over a defined time period
- Introduced charts for tracking column-level metrics over a defined time period
- Introduced the Queries page that displays queries providing sorting and filtering options
- Introduced the Alerts history page that displays a full list of alerts providing sorting and filtering options
- Introduced the Heat map view for alerts — a visual representation of alert distribution over time
- Introduced the Settings page for configuring ADQM Control parameters, such as password complexity and rules for generating alerts
- Introduced functionality for using the ADPG service or an external PostgreSQL instance as a database for ADQM Control
- Introduced functionality for using either the built-in Zookeeper service or an external ZooKeeper instance as a coordination service

Contents

To Table of Contents

ADPG

ADPG

Parameter	Description	Default value
Data directory	Directories used to store data on ADPG hosts	/pg_data1

ADPG configurations

TIP

You can obtain the optimal parameter values to achieve the best ADPG performance using the ***Collect recommendations*** action of the ADPG service.

Parameter	Description	Default value
listen_addresses	TCP/IP addresses on which the server listens for connections from client applications	*
port	TCP port the server listens on	5432
max_connections	Maximum number of concurrent connections to the database server	100
shared_buffers	Amount of memory the database server uses for shared memory buffer. A reasonable starting value is 25% of the memory in your system	2048 MB
max_worker_processes	Maximum number of background processes that the system can support	23
max_parallel_workers	Maximum number of workers that the system can support for parallel operations	4
max_parallel_workers_per_gather	Maximum number of workers that can be started by a single Gather or Gather Merge node	2
max_parallel_maintenance_workers	Maximum number of parallel workers that can be started by a single utility command	2
effective_cache_size	Sets the ADPG query planner's assumption about the effective size of the disk cache available to a single query. This is factored into estimates of the cost of using an index. The higher this value, the more likely the index scan will be applied. The lower this value, the more likely the sequential scan will be selected	6144 MB
maintenance_work_mem	Memory limit for maintenance operations. Increasing the value for this parameter can improve the performance of operations for cleaning and restoring the database from a copy	1024 MB
work_mem	Non-shared memory limit used for internal operations when the server executes queries (for example, queries to temporary tables) before temporary files on disk are used. This limit acts as simple resource control, preventing the host from going into swapping due to overallocation. Since the allocation memory is non-shared memory, large and complex requests can exceed the specified limit	5242 kB
min_wal_size	Until WAL disk usage stays below the `min_wal_size` value, old WAL files are recycled for future use at a checkpoint, rather than removed. This ensures that enough WAL space is reserved to handle spikes in WAL usage, for example, when running large batch jobs	1024 MB
max_wal_size	Memory limit to which the log size can grow between automatic checkpoints. Increasing this setting may increase the recovery time after a failure. The specified limit can be exceeded automatically with a high load on ADPG	4096 MB
wal_keep_size	Specifies the minimum size of segments retained in the *pg_wal* directory, in case a standby server needs to fetch them for streaming replication. If a standby server connected to a sending server falls behind by more than `wal_keep_size` MB, the sending server might remove a WAL segment still needed by the standby. In this case, the replication connection is terminated. Downstream connections also fail as a result. If WAL archiving is enabled, the standby server can fetch the segment from archive and recover. The `wal_keep_size` parameter sets only the minimum size of segments retained in *pg_wal*. The system might need to retain more segments for WAL archival or to recover from a checkpoint. If `wal_keep_size` is set to `0`, the system does not keep any extra segments for standby purposes, so the number of old WAL segments available to standby servers depends on the location of the previous checkpoint and status of WAL archiving	0 MB
huge_pages	Defines whether huge pages can be requested for the main shared memory area. Possible values: • `try` — the server tries to request huge pages. If this fails, the server falls back to the default page • `on` — if a huge page request fails, the server does not start. • `off` — huge pages are not used.	try
superuser_reserved_connections	Number of connection "slots" that are reserved for PostgreSQL superuser connections	3
timescaledb.max_background_workers	Maximum number of background worker processes allocated to TimescaleDB	16
wal_buffers	Amount of shared memory used for WAL data that has not yet been written to disk	16 MB
default_statistics_target	Default statistics target for table columns without a column-specific target set via `ALTER TABLE_SET STATISTICS`	100
random_page_cost	Planner's estimate of the cost of a non-sequentially-fetched disk page	1.1
checkpoint_completion_target	Target time of checkpoint completion as a fraction of total time between checkpoints	0.9
max_locks_per_transaction	The shared lock table tracks locks on `max_locks_per_transaction * (max_connections + max_prepared_transactions)` objects (for example, tables). Hence, no more than this number of distinct objects can be locked at any one time. This parameter controls the average number of object locks allocated for each transaction. Individual transactions can lock more objects as long as the locks of all transactions fit in the lock table	128
autovacuum_max_workers	Maximum number of autovacuum processes (excluding the autovacuum launcher) that are allowed to run at the same time	10
autovacuum_naptime(seconds)	Minimum delay between autovacuum runs on any given database. In each round, the autovacuum daemon examines the database and executes the `VACUUM` and `ANALYZE` commands if it is needed for tables in that database	10
default_toast_compression	Default TOAST compression method for values of a compressible column. Possible values: `lz4`, `pglz`	lz4
effective_io_concurrency	Number of concurrent disk I/O operations that can be executed simultaneously. Raising this value will increase the number of I/O operations that a PostgreSQL session attempts to initiate in parallel	256
jit	Specifies whether PostgreSQL may use JIT compilation, if it is available	off
logging_collector	Enables the logging collector — a background process that captures log messages sent to *stderr* and redirects them into log files	Enabled
log_directory	Directory that contains log files. It can be specified as an absolute path, or relative to the ADPG cluster data directory	log (the absolute path is /pg_data1/adpg14/log)
log_filename	Log file name pattern. A value can include *strftime* %-escapes to define time-varying file names. If you specify a file name pattern without escapes, use a log rotation utility to save disk space	postgresql-%a.log
log_rotation_age	Maximum period of time to use a log file, after which a new log file is created. If this value is specified without units, it is taken as minutes. Set `log_rotation_age` to `0` to disable time-based log file creation	1d
log_rotation_size	Maximum size of a log file. After a log file reaches the specified size, a new log file is created. If the value is set without units, it is taken as kilobytes. Set `log_rotation_size` to `0` to disable size-based log file creation	0
log_min_messages	Minimum severity level of messages to be written to a log file. Possible values are `debug5`, `debug4`, `debug3`, `debug2`, `debug1`, `info`, `notice`, `warning`, `error`, `log`, `fatal`, and `panic` (see the *Severity levels* table). Messages with the specified severity or higher are included in the log file. For example, if you set `log_min_messages` to `warning`, the log file will include the `warning`, `error`, `log`, `fatal`, and `panic` messages	warning
log_min_error_statement	Specifies which SQL statements that cause errors are logged. Possible values are `debug5`, `debug4`, `debug3`, `debug2`, `debug1`, `info`, `notice`, `warning`, `error`, `log`, `fatal`, and `panic` (see the *Severity levels* table). The log file includes SQL statements for messages of the specified severity or higher. To disable error statement logging, set `log_min_error_statement` to `panic`	error

Custom postgresql.conf

In the ***Custom postgresql.conf*** section, you can set additional parameters to be written to the *postgresql.conf* file. To add a parameter, click ***Add property***.



Custom postgresql.conf section

In the window that appears, enter a parameter (its name and value) and click ***Apply***.

Window to add parameter

You can add multiple parameters.

Shared preload libraries

Section for setting the `shared_preload_libraries` parameter of the *postgresql.conf* configuration. ADGM Control requires the `timescaledb` and `timescaledb_toolkit` extensions. If you need to add other extensions (for example, `pg_stat_statements`), use the icon + .

The "Shared preload libraries" section

Custom pg_hba.conf

This field allows you to add records to the *pg_hba.conf* file that controls client authentication.

SSL configuration

NOTE

For more information on SSL encryption support in ADQM Control, see the *SSL encryption* article.

Parameter	Description	Default value
Enable SSL	Enables SSL encryption of incoming traffic for the ADPG service	Disabled
Certificate file	Path to the server SSL certificate file	/etc/adpg16/server.crt
Private key file	Path to the file with the private key of the server SSL certificate	/etc/adpg16/server.key
Certificate authority file	Path to the CA certificate file	/etc/adpg16/ca.crt

Contents

To Table of Contents

ADQM Control

ADQM Control

ADQM management

Parameter	Description	Default value
ADQM username	Username to connect to ADQM	adqmc
ADQM user password	Password for the `ADQM username` user to connect to ADQM (should have a minimum length of 8 characters and should not contain `#`, `|`, `{`, `}`, `%`, `$`, `^`, `/`, `<`, `>`, `;`, `?`, `"`, or `\`)	–
ADQM default port	Default port if it is not exported from an ADQM cluster	9000
ADQM secure port	Default port for SSL-secured communication with ADQM if it is not exported from an ADQM cluster	9440
Connection type	Type of connection (insecure or secure) between ADQM Control and ADQM. Possible values: `Insecure (TCP)`, `Secure (TCP secure)`	Insecure (TCP)

NOTE

A name and password passed to the `ADQM username` and `ADQM user password` parameters should belong to a user account that exists in ADQM. For a new cluster, ADQM provides the `default` user account without a password by default – you can use this account in ADQM Control, but first assign a password to it. However, it is recommended to create a separate user account in ADQM under which ADQM Control will connect to ADQM, for example:

```
CREATE USER adqmc ON CLUSTER default_cluster IDENTIFIED WITH sha256_password BY 'qwerty123';


GRANT ON CLUSTER default_cluster SELECT ON *.* TO adqmc;
```

Backend

Parameter	Description	Default value
Session timeout	ADQM Control web interface session timeout (in seconds)	1800
Enable CSRF validation	Enables CSRF validation	Enabled

SSL configuration

NOTE

For details on how to configure and enable SSL for components of ADQM Control's services, see the SSL encryption article.

Parameter	Description	Default value
[ADQM Control's Backend] → Enable SSL	Enables SSL for the Backend component	Disabled
[ADQM Control's ADQM Agent] → Enable SSL	Enables SSL for the Agents component	Disabled
[ADQM Control's Alert Generator] → Enable SSL	Enables SSL for the Alert Generator component	Disabled
[ADQM Control's Alert Receiver] → Enable SSL	Enables SSL for the Alert Receiver component	Disabled
[Alertmanager] → Enable SSL	Enables SSL for the Alertmanager component	Disabled
[ADQM Control Services] → Certificate file	Path to the server SSL certificate file for the Backend, Agents, Alert Generator, Alert Receiver components	/etc/adqmc/server.crt
[ADQM Control Services] → Private key file	Path to the file with the private key of the server SSL certificate for the Backend, Agents, Alert Generator, Alert Receiver components	/etc/adqmc/server.key
[ADQM Control Services] → Certificate authority file	Path to the file with the CA certificate for the Backend, Agents, Alert Generator, Alert Receiver components	/etc/adqmc/ca.crt
[Alertmanager] → Certificate file	Path to the server SSL certificate file for the Alertmanager component	/etc/admprom/alert-manager/server.crt
[Alertmanager] → Private key file	Path to the file with the private key of the server SSL certificate for the Alertmanager component	/etc/admprom/alert-manager/server.key
[Alertmanager] → Certificate authority file	Path to the file with the CA certificate for the Alertmanager component	/etc/admprom/alert-manager/ca.crt

Network configuration

Parameter	Description	Default value
Backend port	Port to access the ADQM Control web interface	5555
Alert Generator port	Port the alert generator listens on	5001
ADQM agent port	Port the Agents service listens on	5002
Alertmanager port	Port the alert manager listens on	9093
Alert Receiver port	Port the alert receiver listens on	12322

Connection settings

Parameter	Description	Default value
Backend connection pool size	Maximum number of connections to a PostgreSQL database of the Backend service	8
ADQM agent connection pool size	Maximum number of connections to a PostgreSQL database of the Agents service	8
Statement timeout	Timeout for queries and transactions (in seconds)	120
Cleanup timeout	Server timeout (in seconds) abort any database query that runs longer than the specified time	86400
Migration timeout	Database migration timeout (in seconds)	3600
Prometheus timeout	Prometheus connection timeout (in seconds)	60
Alertmanager timeout	Alertmanager connection timeout (in seconds)	60

Polling rate

Parameter	Description	Default value
Hosts scrape interval	Frequency of updating information about the ADQM cluster topology (in seconds)	80
ADQMDB scrape interval	Frequency of updating information about tables in the ADQM cluster (in seconds)	80
Query update interval	Frequency of collecting queries (in seconds)	105
Query normalize interval	Frequency of updating queries (in seconds)	120
Check query_log interval	Frequency of checking query_log (in seconds)	60

Limits

Parameter	Description	Default value
Normalize query raw max queries per iteration	Maximum number of queries normalized per iteration	50000
Collect queries limit	Number of queries collected from the query_log per iteration	10000
Query normalize threshold	Number of attempts to normalize a query	10
Existing tables tries	Number of attempts to get metrics of an ADQM table. If ADQM Control has failed to retrieve the metrics for a table after the expiration of the time `ADQMDB scrape interval * Existing tables tries`, it considers this table deleted	3
Tables cache size	Size of the table cache	32768
Table cache invalidate interval	Time interval of the table cache invalidation (in seconds)	28800
Maximum revision per table in cache	Maximum number of revisions per table in the cache	100
Queries cache size	Size of the query cache	8192
Batch size of queries normalization	Query batch size for normalization	1024
Capacity of existing tables cache	Capacity of the existing tables cache	5000000000
Existing tables cache invalidate period	Period of the existing table cache invalidation (in seconds)	604800
Table's existing period for existing tables cache calculation	Period of a table's existence (in seconds) for the existing table cache calculation	7776000
Maximum of parallel workers	Maximum number of parallel workers	0

Internal PostgreSQL server

NOTE

To access the configuration parameters, activate the ***Use internal PostgreSQL server*** option.

Parameter	Description	Default value
Alerts dbname	Name of a database to be created for use by ADQM Control services	adqmc
Alertsdb username	Name of a user with access to the `Alerts dbname` database	adqmc
Alertsdb password	Password for the `Alertsdb username` user to access the `Alerts dbname` database (should have a minimum length of 8 characters and should not contain `^`, `#`, `|`, or `\`)	–

External PostgreSQL server

NOTE

To access the configuration parameters, activate the ***Use external PostgreSQL server*** option.

Parameter	Description	Default value
server_ip_port	Click ***Add property*** to specify an IP address (as property name) and port (as property value) to access an external PostgreSQL server	–
Alerts dbname	Database name to be used by ADQM Control services	adqmc
Alertsdb username	Name of a user with access to the `Alerts dbname` database	adqmc
Alertsdb password	Password for the `Alertsdb username` user to access the `Alerts dbname` database (should have a minimum length of 8 characters and should not contain `^`, `#`, `|`, or `\`)	–
SSL is used by an external server	Specifies whether SSL is used by an external server	Disabled

Contents

To Table of Contents

Monitoring

Monitoring

Prometheus settings

Parameter	Description	Default value
scrape_interval	Specifies how frequently to scrape targets	1m
scrape_timeout	Specifies how long to wait until a scrape request times out	10s
Password for grafana connection	Password of a Grafana user to connect to Prometheus	—
Prometheus users to login/logout to Prometheus	User credentials for logging into the Prometheus web interface	—
Service parameters	Prometheus parameters	—

Grafana settings

Parameter	Description	Default value
Grafana administrator's password	Password of a Grafana administrator user	—
Grafana listen port	Port to access the Grafana web interface	11210

Node Exporter settings

Parameter	Description	Default value
Listen port	Port to listen for a host's system metrics in the Prometheus format	11203
Metrics endpoint	Endpoint to retrieve system metrics	/metrics

SSL configuration

Parameter	Description	Default value
[Prometheus] → Enable SSL	Enables SSL encryption of incoming traffic for the Prometheus Server component	Disabled
[Prometheus] → Certificate file	Path to the server SSL certificate file for the Prometheus Server component	/etc/admprom/prometheus/server.crt
[Prometheus] → Private key file	Path to the file with the private key of the server SSL certificate for the Prometheus Server component	/etc/admprom/prometheus/server.key
[Prometheus] → Certificate authority file	Path to the file with the CA certificate to verify Prometheus targets	/etc/admprom/prometheus/ca.crt
[Grafana] → Enable SSL	Enables SSL encryption of incoming traffic for the Grafana component	Disabled
[Grafana] → Certificate file	Path to the server SSL certificate file for the Grafana component	/etc/admprom/grafana/server.crt
[Grafana] → Private key file	Path to the file with the private key of the server SSL certificate for the Grafana component	/etc/admprom/grafana/server.key
[Grafana] → Certificate authority file	Path to the file with the CA certificate to verify Prometheus certificate	/etc/admprom/grafana/ca.crt
[Node-exporter] → Enable SSL	Enables SSL encryption of incoming traffic for the Node Exporter component	Disabled
[Node-exporter] → Certificate file	Path to the server SSL certificate file for the Node Exporter component	/etc/ssl/server.crt
[Node-exporter] → Private key file	Path to the file with the private key of the server SSL certificate for the Node Exporter component	/etc/ssl/server.key

Contents

To Table of Contents

Zookeeper

Zookeeper

Main

Parameter	Description	Default value
connect	ZooKeeper connection string used by other services or clusters. It is generated automatically	—
dataDir	Location where ZooKeeper stores the in-memory database snapshots and, unless specified otherwise, the transaction log of updates to the database. It is specified as the `dataDir` parameter in the *zoo.cfg* configuration file	/var/lib/zookeeper

zoo.cfg

Parameter	Description	Default value
clientPort	Port to listen for client connections, that is the port that clients attempt to connect to	2181
admin.serverPort	Port that the embedded Jetty server listens on	5181
admin.enableServer	Enables Admin server — an embedded Jetty server that provides an HTTP interface to the four-letter-word commands	Disabled
tickTime	Length of a single tick (in milliseconds) which is the basic time unit used by ZooKeeper to regulate heartbeats and timeouts	2000
initLimit	Amount of time (in ticks) to allow followers to connect and sync with a leader. Increase this value as needed if the amount of data managed by ZooKeeper is large	5
syncLimit	Amount of time (in ticks) to allow followers to sync with ZooKeeper. If followers fall too far behind a leader, they will be dropped	2
maxClientCnxns	Limits the number of active connections from a single client, identified by IP address, to a single member of the ZooKeeper ensemble	0
autopurge.snapRetainCount	When enabled, ZooKeeper auto-purge feature retains the specified number of the most recent snapshots and the corresponding transaction logs in *dataDir* and *dataLogDir* respectively, and deletes the rest. The minimum value is `3`	3
autopurge.purgeInterval	Time interval (in hours), for which the purge task has to be triggered. Set a positive integer to enable the auto purging	24
Add key,value	In this section, you can define values for custom parameters that are not displayed in ADCM UI, but are allowed in the *zoo.cfg* configuration file	—

SSL configuration

Parameter	Description	Default value
Enable SSL	Enables secure incoming connections in ZooKeeper	Disabled
sslQuorum	Enables encrypted quorum communication	Disabled
serverCnxnFactory	Specifies `ServerCnxnFactory` implementation. To use TLS-based server communication, set it to `NettyServerCnxnFactory`	org.apache.zookeeper.server.NettyServerCnxnFactory
ssl.quorum.keyStore.location	Fully-qualified path to the server keystore file	/etc/zookeeper/server.jks
ssl.quorum.keyStore.password	Password for the keystore specified in `ssl.quorum.keyStore.location`	—
ssl.quorum.trustStore.location	Fully-qualified path to the server truststore file	/etc/zookeeper/ca.jks
ssl.quorum.trustStore.password	Password for the truststore specified in `ssl.quorum.trustStore.location`	—
ssl.keyStore.location	Fully-qualified path to the server keystore file to be used for client TLS connections	/etc/zookeeper/server.jks
ssl.keyStore.password	Password for the keystore specified in `ssl.keyStore.location`	—
ssl.trustStore.location	Fully-qualified path to the server truststore file to be used for client TLS connections	/etc/zookeeper/ca.jks
ssl.trustStore.password	Password for the truststore specified in `ssl.trustStore.location`	—
ssl.protocol	Protocol to be used in client TLS negotiation	TLSv1.2
ssl.quorum.protocol	Protocol to be used in quorum TLS negotiation	TLSv1.2
secureClientPort	Port to be used for secure client connections	2281

Metrics

Parameter	Description	Default value
Listen port	Port to listen for Zookeeper service metrics	9020

zookeeper-env.sh

Parameter	Description	Default value
JAVA	Path to Java	$JAVA_HOME/bin/java
ZOOPIDFILE	Directory to store the ZooKeeper process ID	/var/run/zookeeper/zookeeper_server.pid
ZOO_LOG_DIR	Directory to store logs	/var/log/zookeeper
SERVER_JVMFLAGS	Used to set different JVM parameters (for example, parameters related to garbage collection)	-Xmx1024m