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 1.2.0 Language: EN

Contents

1. Concepts
   - ADQM Control overview
2. 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
3. Access management
   - Manage users
   - SSL encryption
4. Cluster maintenance
   - Cluster actions
   - Service actions
   - Configure an external database
5. Web user interface
   - Alert settings
   - View a dashboard
   - Cluster metrics
   - Tables
   - Queries
6. References
   - Configuration parameters
   - Query exception codes
   - Monitoring metrics
7. 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.

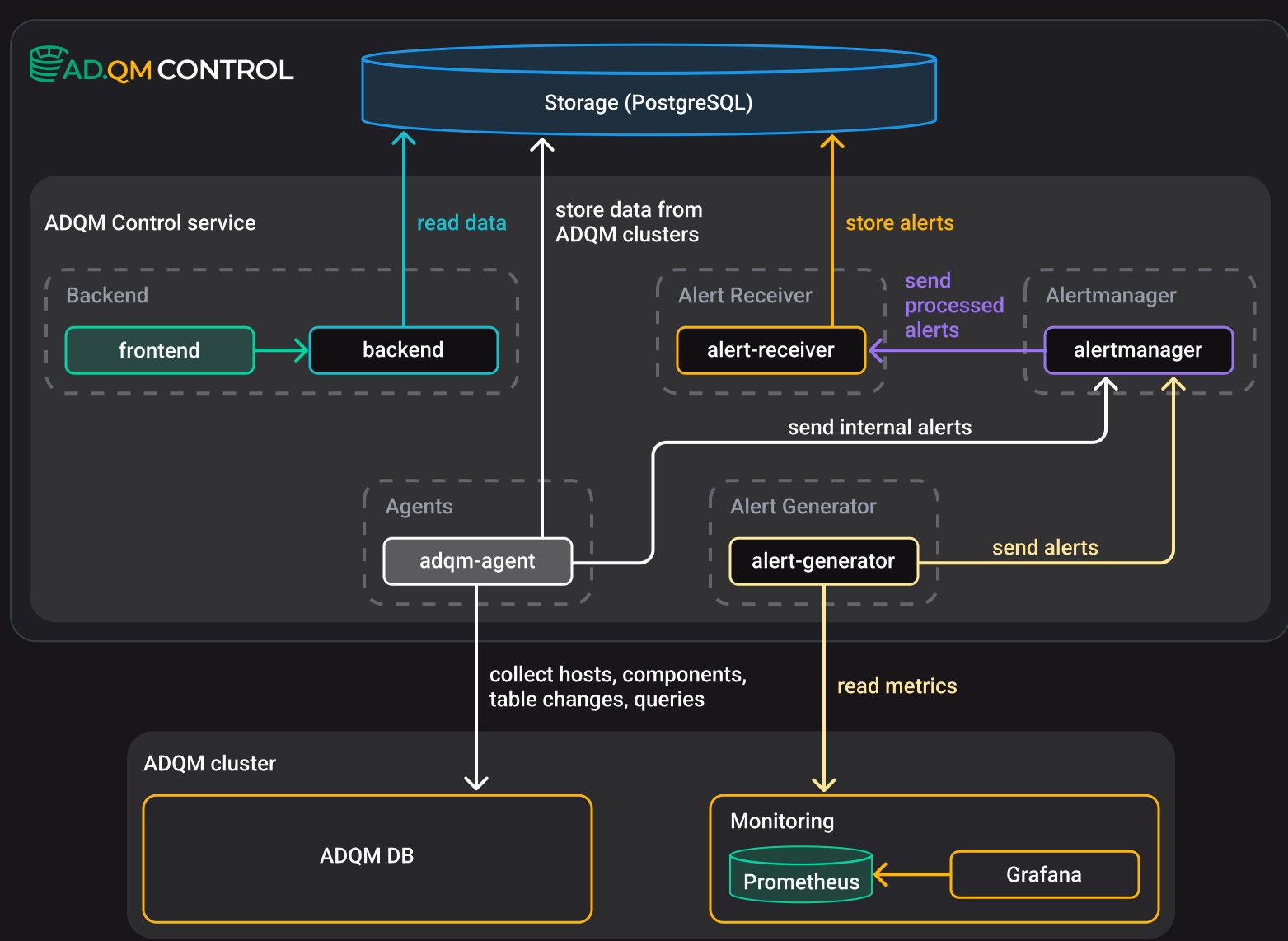
  In the ADQM Control user interface, you can specify criteria for generating alerts of different types (for example, configure thresholds, set timeouts) or disable tracking of some events.

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.

Installation

ADQM CONTROL

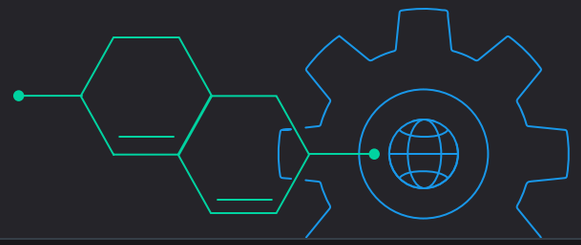
To Table of Contents

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

ADQM CONTROL

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

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.

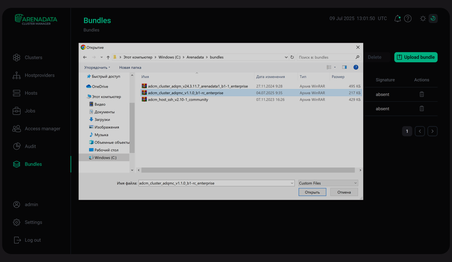
Step 1. Get the Enterprise version of a cluster bundle

Please get a cluster bundle from the Arenadata support team.

Step 2. Upload a cluster bundle to ADCM

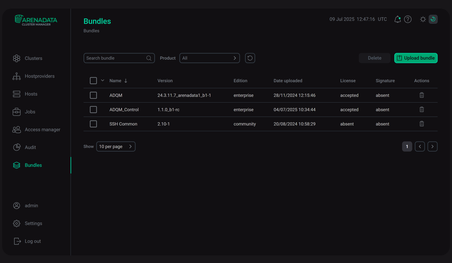
To upload a bundle to ADCM, follow the steps:

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

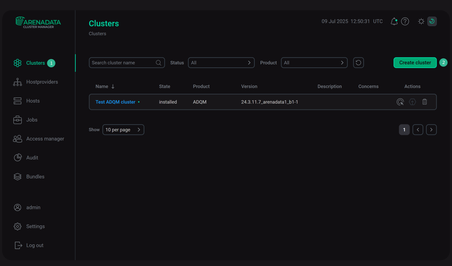
Step 3. Create a cluster based on the uploaded bundle

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.

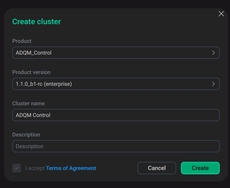
The steps for cluster creation are given below:

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



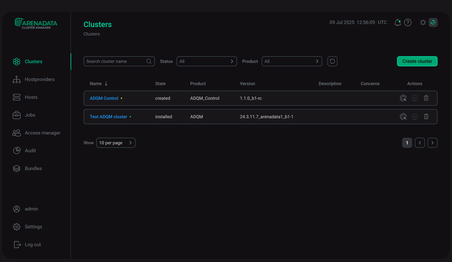
Switch to cluster creation

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

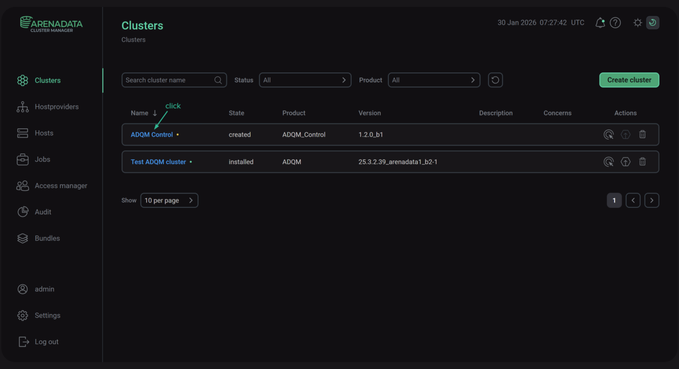
Add services

Elena Dvoryadkina, Daria Barysheva

To Table of Contents

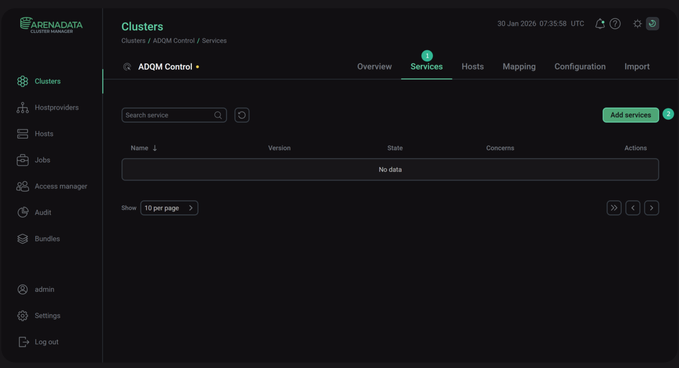
In ADCM, a **service** means a software that performs some function. The steps for adding services to a cluster are listed below:

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



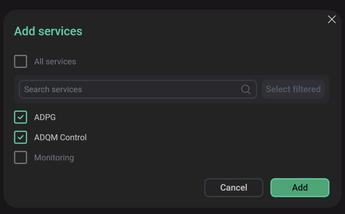
Select a cluster

2. Open the ***Services*** tab on the cluster page and click ***Add service***.



Switch to adding services

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



Select services

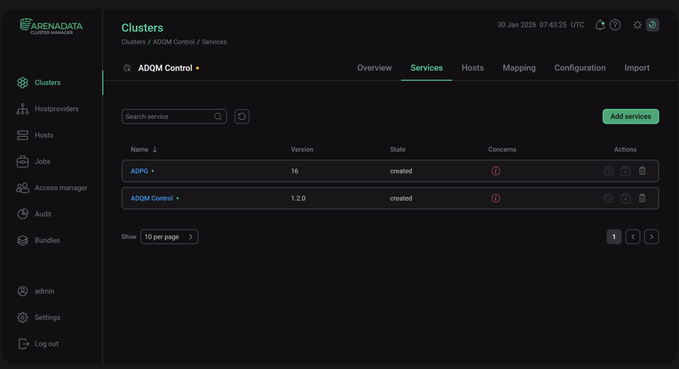
The brief description of available services is listed below.

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

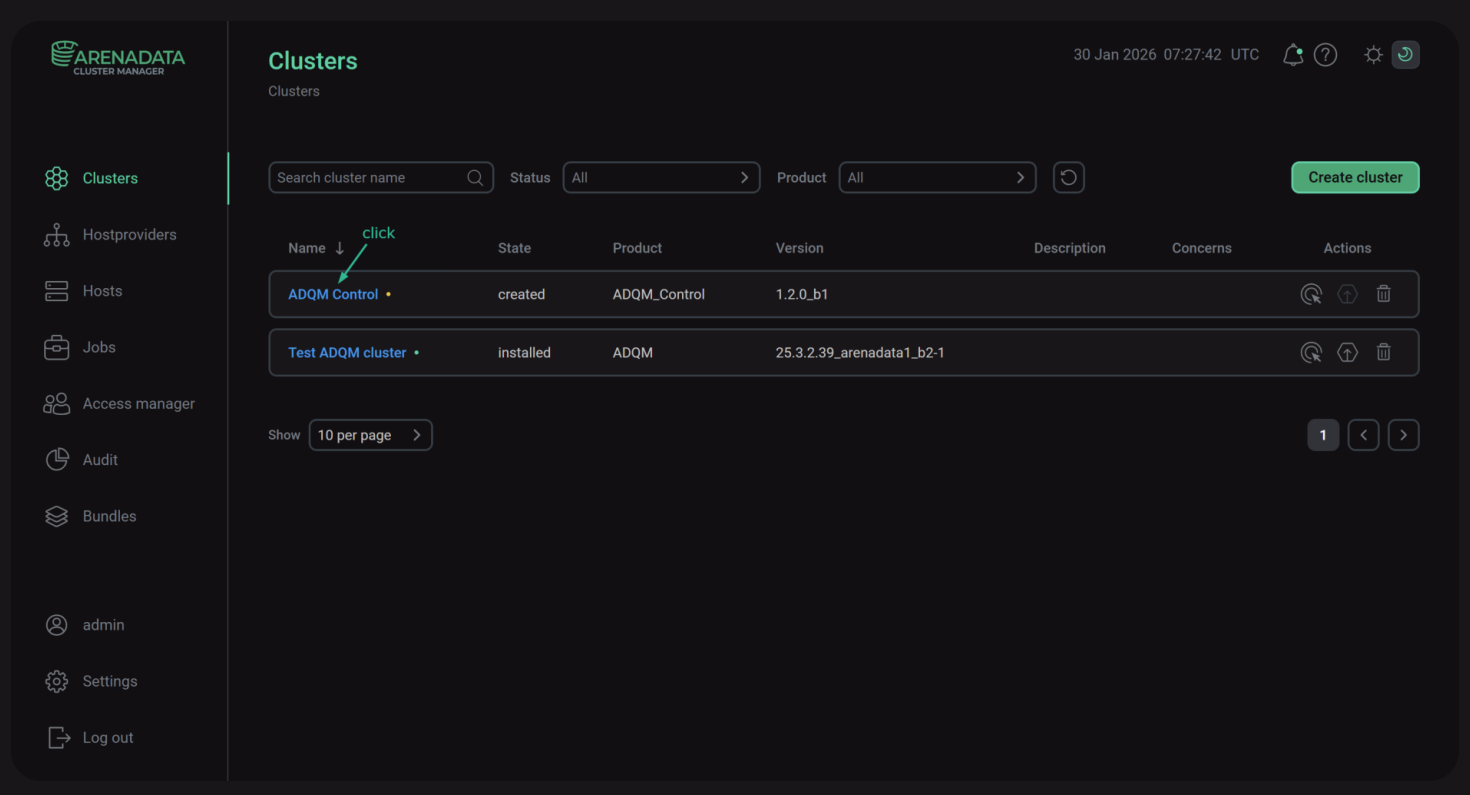
To Table of Contents

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.

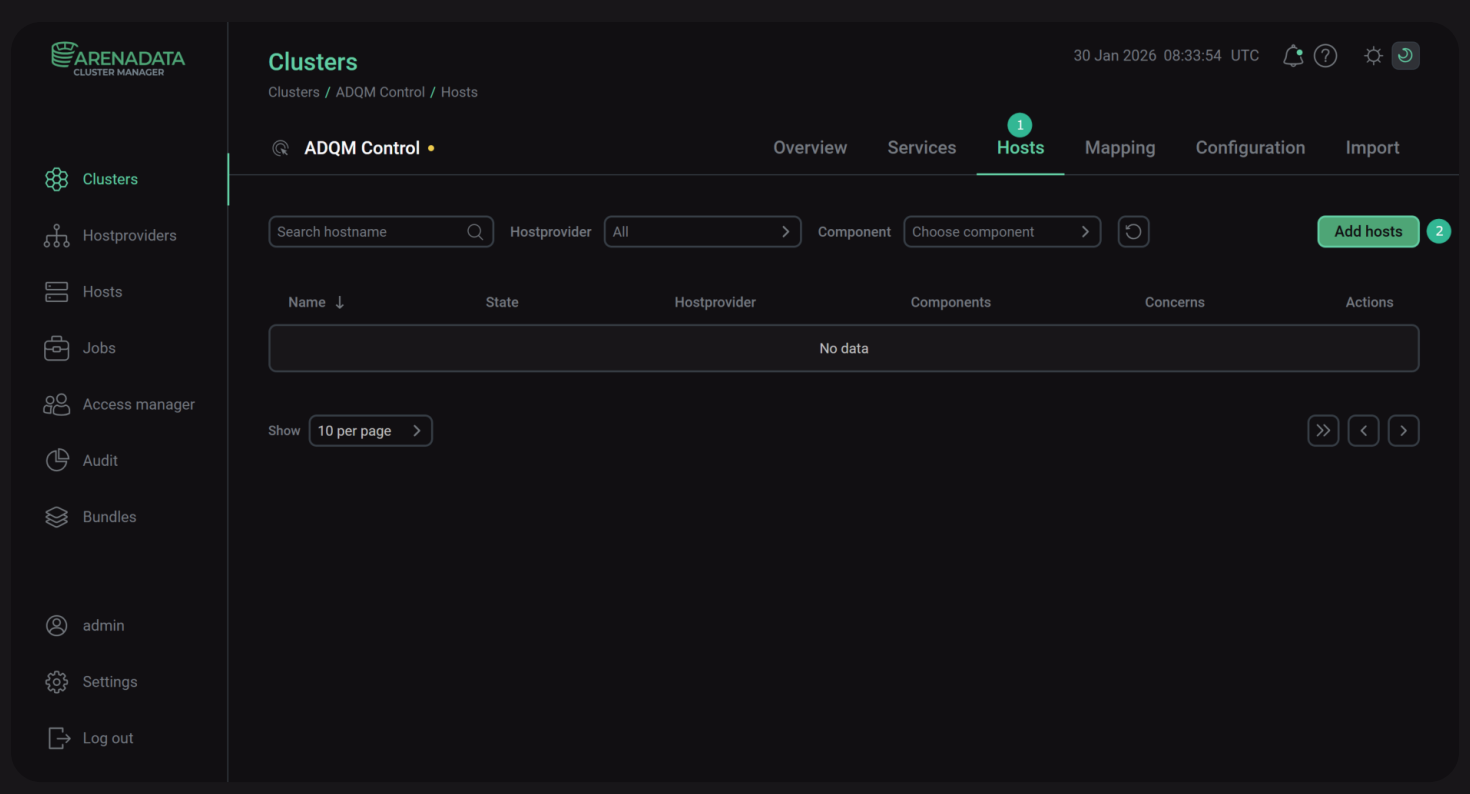
The steps for adding hosts to a cluster are listed below:

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



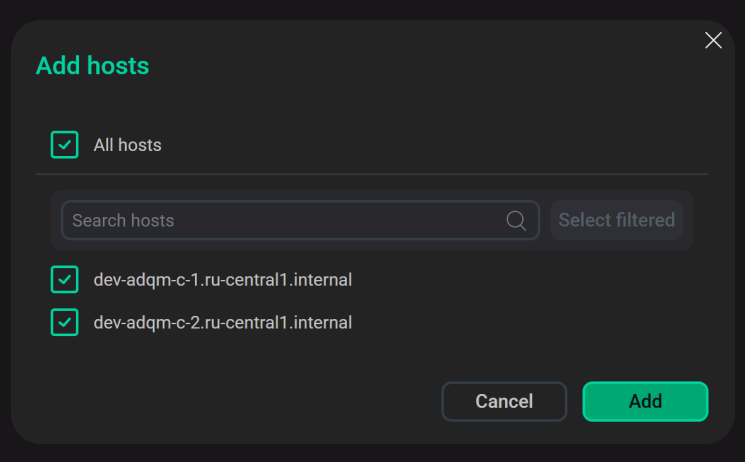
Select a cluster

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



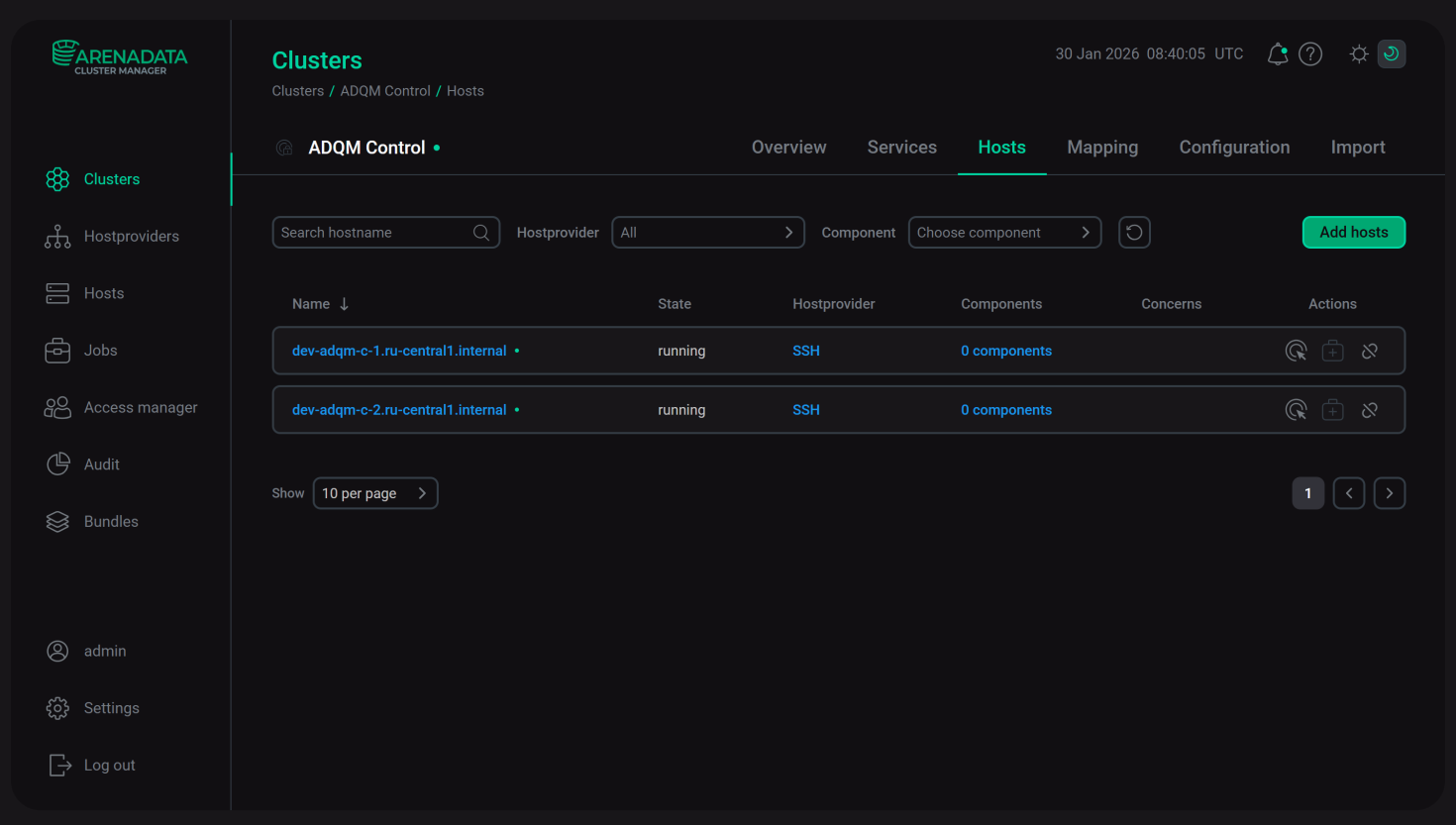
Switch to adding hosts

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 Baryshева

ADQM CONTROL

To Table of Contents

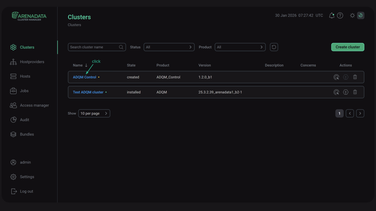
In ADCM, a **component** means a part of a service that should be deployed on one or several cluster hosts.

IMPORTANT

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

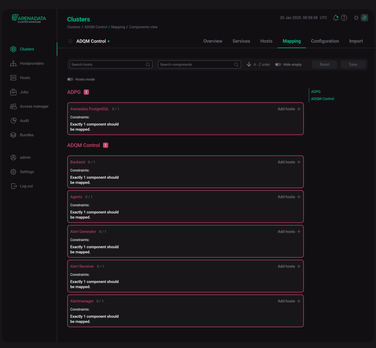
The steps for adding components to hosts are listed below:

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.

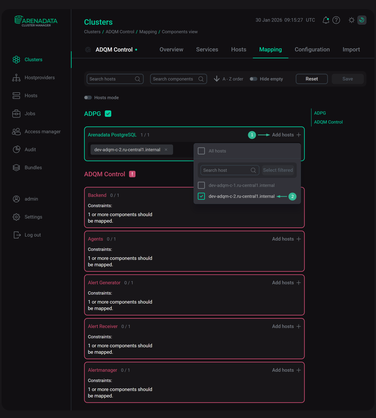
CAUTION

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.

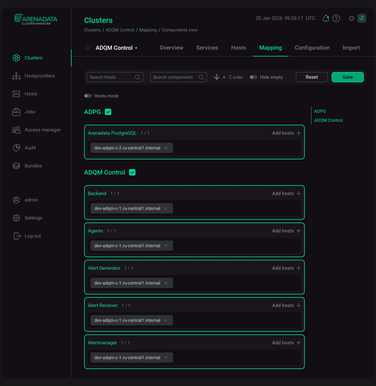


Select hosts for a component

TIP

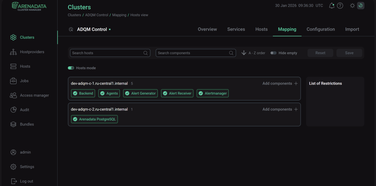
It is recommended to install the Arenadata PostgreSQL component on a dedicated host, separately from the components of the ADQM Control service.

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.



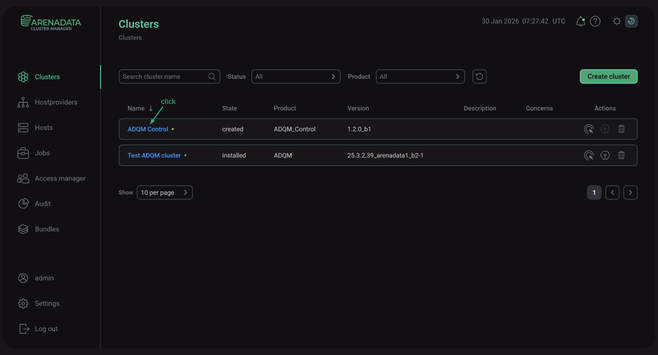
Use mapping by hosts

Configure services

Elena Dvoryadkina, Daria Barysheva

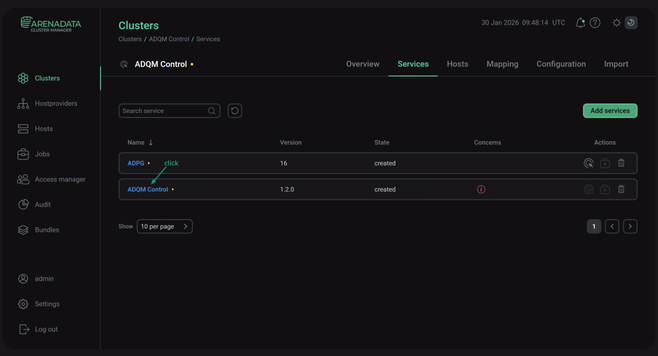
To Table of Contents

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



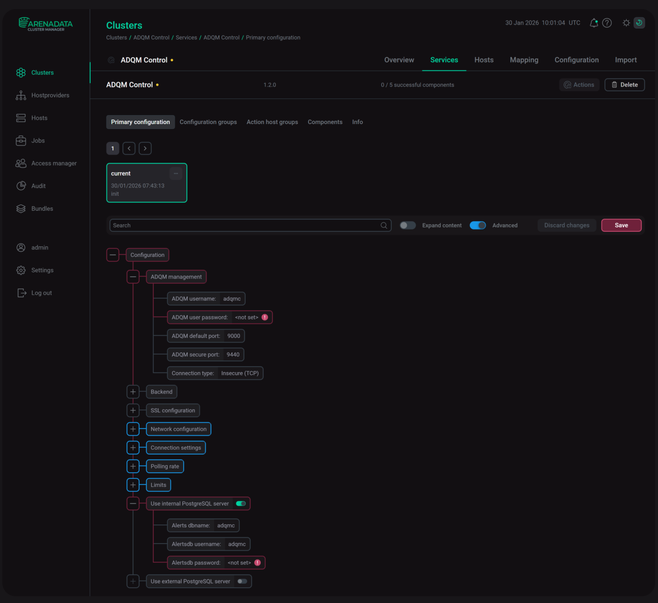
Select a cluster

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

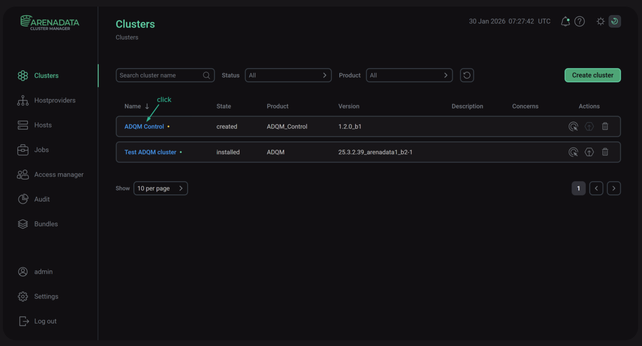
Ansible settings

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

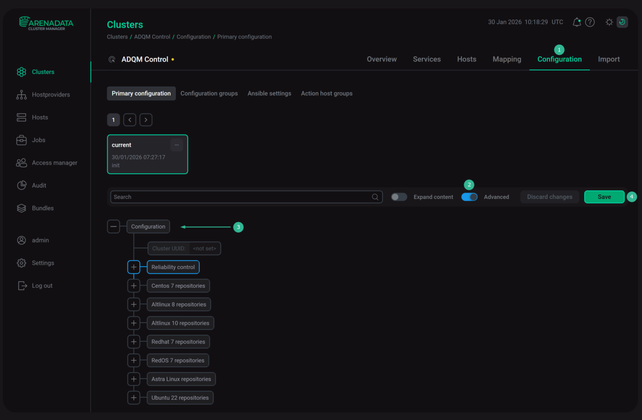
2. Open the ***Configuration*** tab on the cluster page. The ***Configuration*** tab includes the following sections: ***Primary configuration***, ***Configuration groups***, ***Ansible settings***.
3. Fill in all necessary parameters on the selected tab and click ***Save***.

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

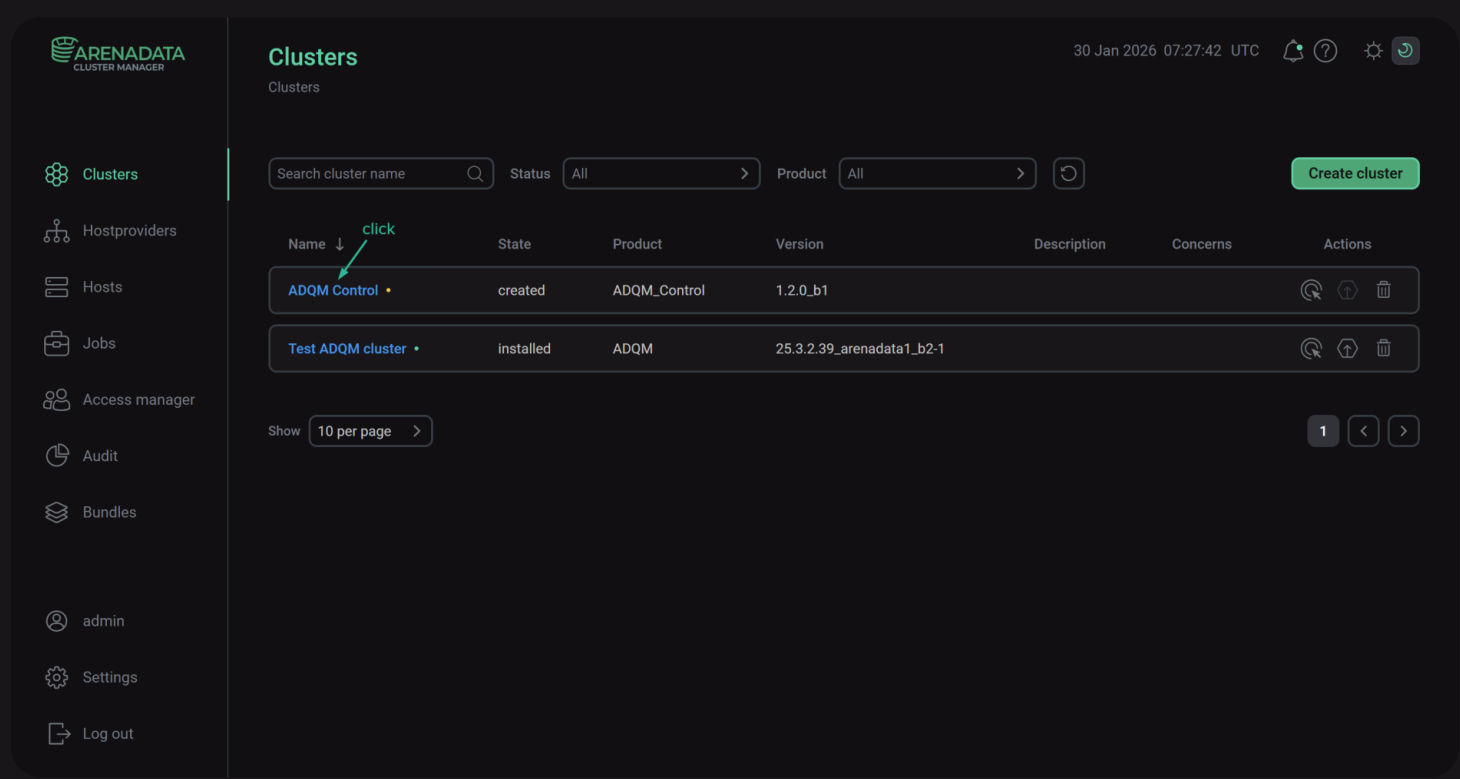
Integrate with an ADQM cluster

Elena Dvoryadkina

To Table of Contents

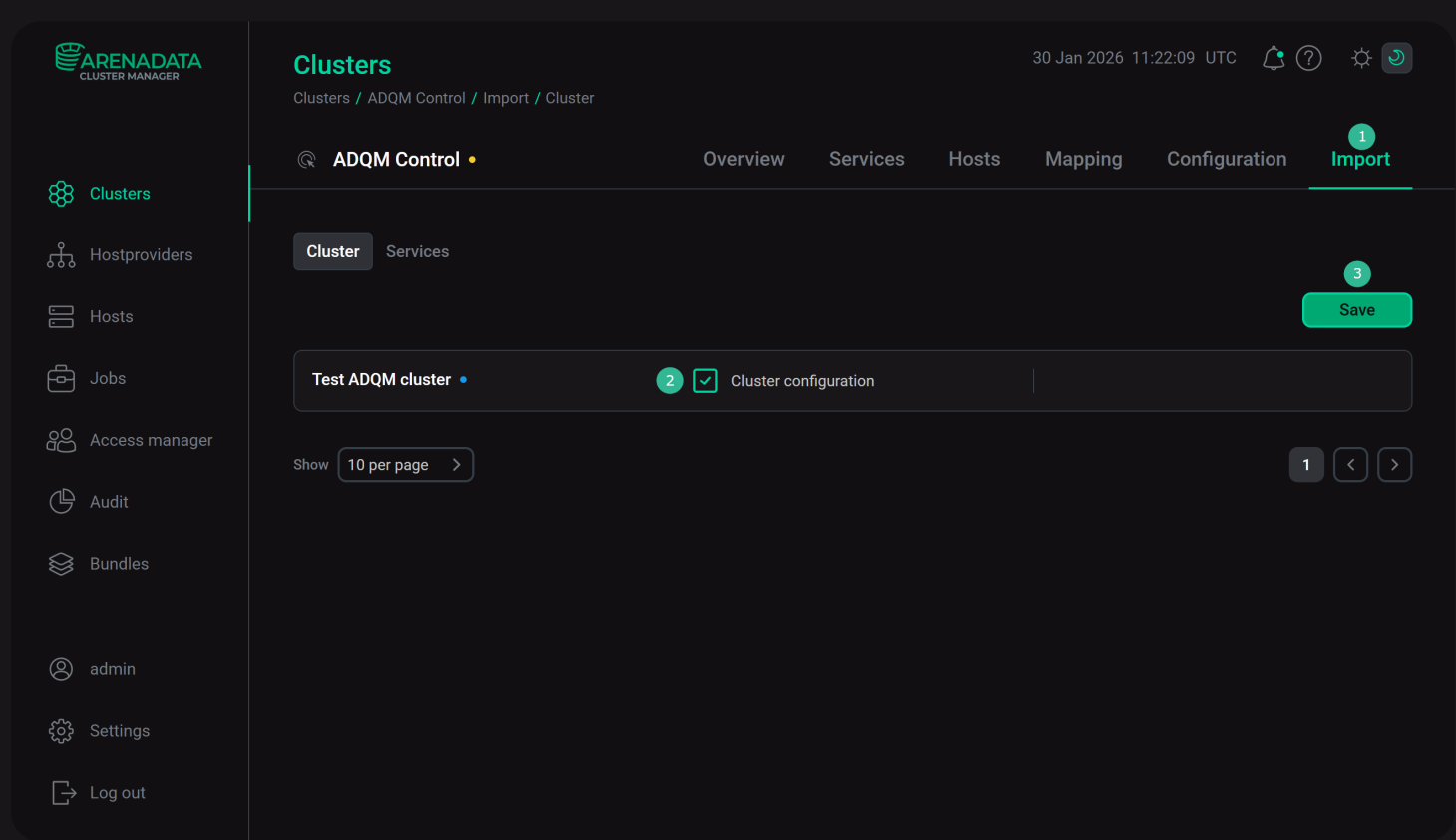
To set up an ADQM Control cluster, you should integrate it with at least one pre-installed ADQM cluster with the Monitoring service configured:

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

A. Elena Dorovydenka, Daria 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.

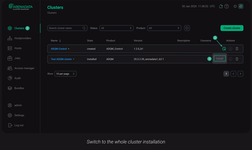
In both cases each service starts automatically after installation.

> **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

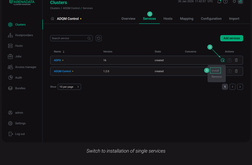
2. Wait until the job is completed.

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.



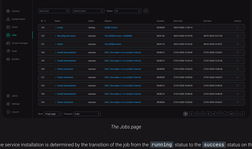
Switch to installation of single services

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

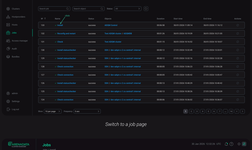
Step 2. View the results of installation

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.

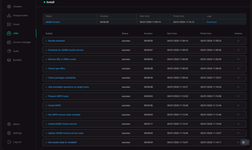


The Jobs page

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

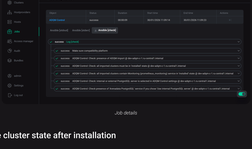


Install cluster job page

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.

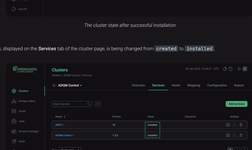


Job details

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`.



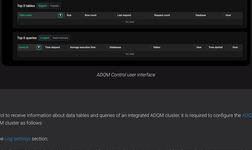
The cluster state after successful installation

- 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 `Owner` 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 ADQM DB 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 ADQM DB service to save configuration changes and restart the service.
>
> - If you have imported an ADQM cluster version up to and including 24.3.11.7.1.b1 into ADQM Control, and the Monitoring service was installed during the installation of the ADQM cluster (i.e. its state ***Install*** action at the cluster level, not by the service actions), 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

A Beta Installation

ADOM CONTROL

Contents

To Table of 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 ADOM Control to collect and store ADOM Control cluster monitoring metrics, and also supports the ability to use the Grafana web application for visualization and analysis of information. This article describes the steps required to add the service to an ADOM Control cluster.

Overview

After adding the monitoring service to ADOM Control, you can install the node exporter on all or selected hosts of your cluster. It is a monitoring agent that reads host system metrics and exports them in a format that can be read by the Prometheus server. The node exporter runs as a Linux service on the specified hosts and listens for service requests on port 9100 by default. You can configure Prometheus. 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-compatible monitoring system set up (for example, your own Prometheus server or VictoriaMetrics), you can use it to collect ADOM Control metrics. To do this, configure access to ADOM Control metrics in your monitoring system using ADOM Control's Prometheus parameters specified on the service configuration.
- You can also use the Installation mechanism to migrate all metrics from the Prometheus server deployed in ADOM Control to your Prometheus.

Step 1. Add the service

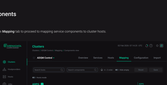
1. In the ADOM Control interface, open the **Clusters** page and click your ADOM Control cluster name. On the cluster page that opens, scroll to the **Services** tab and click **Add services**.

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

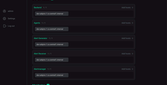
1. On the cluster page, open the **Mapping** tab to proceed to mapping service components to cluster hosts.

Switch to mapping service components

Monitoring service components

Component	Description
Prometheus Server	Stores and allows requesting all metrics for an ADOM Control cluster
Grafana	Visualizes ADOM 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

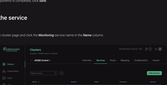
CAUTION

It is not recommended to install the Prometheus Server component on hosts with ADOM Control – use separate hosts for it. Otherwise, if an ADOM 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's configuration parameters.

Configure the monitoring service

Fields highlighted in red are required.

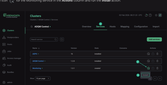
- **Prometheus settings** – *Password for Grafana connection* – a password of a Grafana user (**GRAFANA_WEBPASS**) that is used to connect to Prometheus. The password must contain a combination of numbers, symbols, lowercase and uppercase letters and have a minimum length of 8.
- **Grafana settings** – **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 required parameters, click **Save**.

Step 4. Install the service

1. On the **Services** tab, click the icon for the Monitoring service 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 **Installed** → **Installation**.

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 ADOM Control's services are collected from the cluster hosts – see ADOM Control Metrics in Grafana. To do this, view metrics in the Prometheus format in the browser or use the Prometheus or Grafana web interfaces.

Metrics in the Prometheus format

1. In the address bar of your browser, enter an IP address of a host where the Prometheus Server component is installed. Specify also a port number from the **web.listen-address** parameter in the **Source parameters** section of the **Prometheus settings** — settings for accessing to system metrics of the ADOM Control cluster, 127.0.0.1:9090 – for example, http://127.0.0.1:9090. To log into the Prometheus interface, use a username and password that are also specified in the **Prometheus settings** section – the **Prometheus.com to login/logout to Prometheus** setting.
2. In the **Expression** field, you can enter a metric name and click **Execute** – values of this metric on all hosts of the ADOM Control cluster will be shown in the interface.

Metrics in the Prometheus format

Prometheus web interface

1. In the address bar of your browser, enter an IP address of a host where the Prometheus Server component is installed. Specify also a port number from the **web.listen-address** parameter in the **Source parameters** section of the **Prometheus settings** — settings for accessing to system metrics of the ADOM Control cluster, 127.0.0.1:9090 – for example, http://127.0.0.1:9090. To log into the Prometheus interface, use a username and password that are also specified in the **Prometheus settings** section – the **Prometheus.com to login/logout to Prometheus** setting.
2. In the **Expression** field, you can enter a metric name and click **Execute** – values of this metric on all hosts of the ADOM 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 where the Grafana component is installed. Specify also a port number from the **web.listen-address** parameter in the **Grafana settings** section of the Monitoring service configuration (the default value is 127.0.0.1:3000). For example, http://127.0.0.1:3000. To log in, use **admin** as a username, and the **Grafana administrator's password** parameter value (that specified 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 **ADOM_Dashboard**/**ADOM_Control_Cluster_name** section. In this section, you can select one of the available dashboards by name and type a search metrics in the corresponding field in the search bar.

View ADOM 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

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.

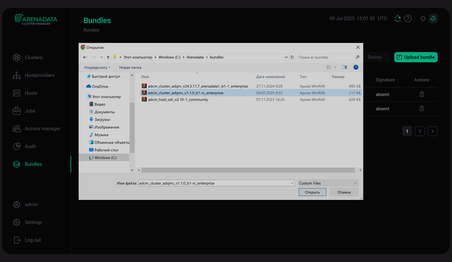
Step 1. Get the Enterprise version of a cluster bundle

Please get a cluster bundle from the Arenadata support team.

Step 2. Upload a cluster bundle to ADCM

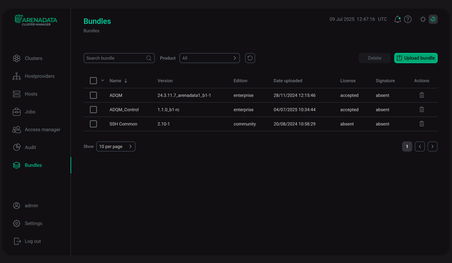
To upload a bundle to ADCM, follow the steps:

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

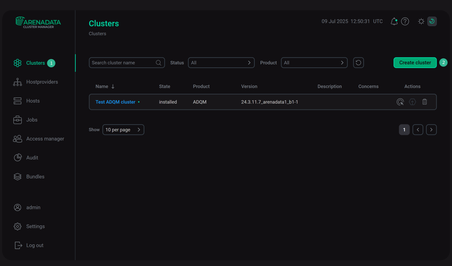
Step 3. Create a cluster based on the uploaded bundle

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.

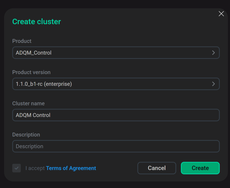
The steps for cluster creation are given below:

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



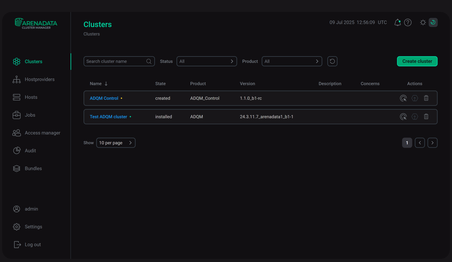
Switch to cluster creation

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

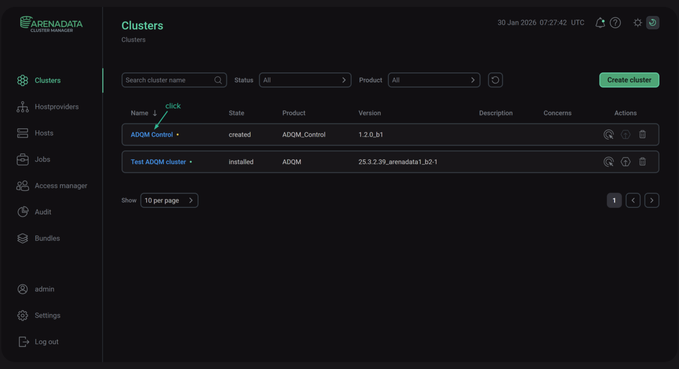
Add services

Elena Dvoryadkina, Daria Barysheva

To Table of Contents

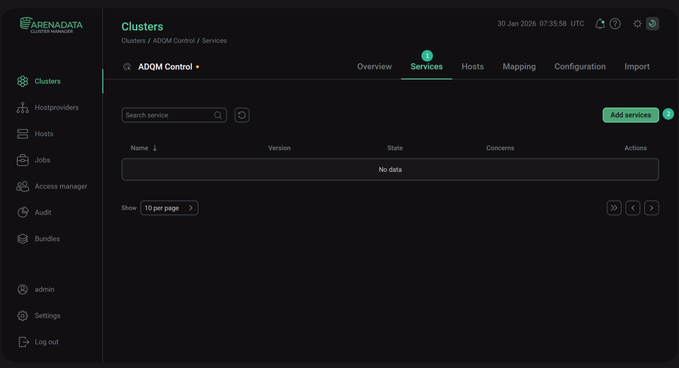
In ADCM, a **service** means a software that performs some function. The steps for adding services to a cluster are listed below:

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



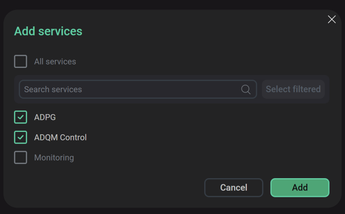
Select a cluster

2. Open the ***Services*** tab on the cluster page and click ***Add service***.



Switch to adding services

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



Select services

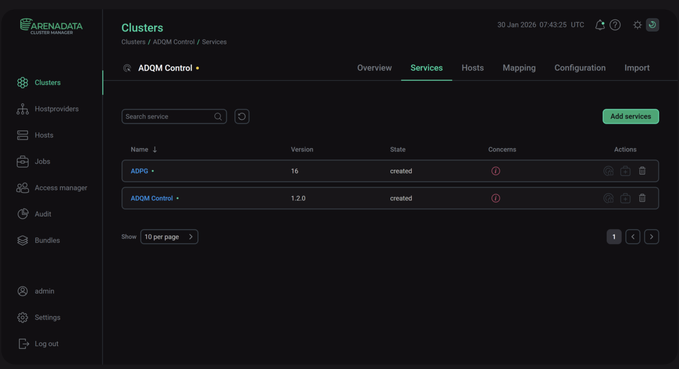
The brief description of available services is listed below.

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

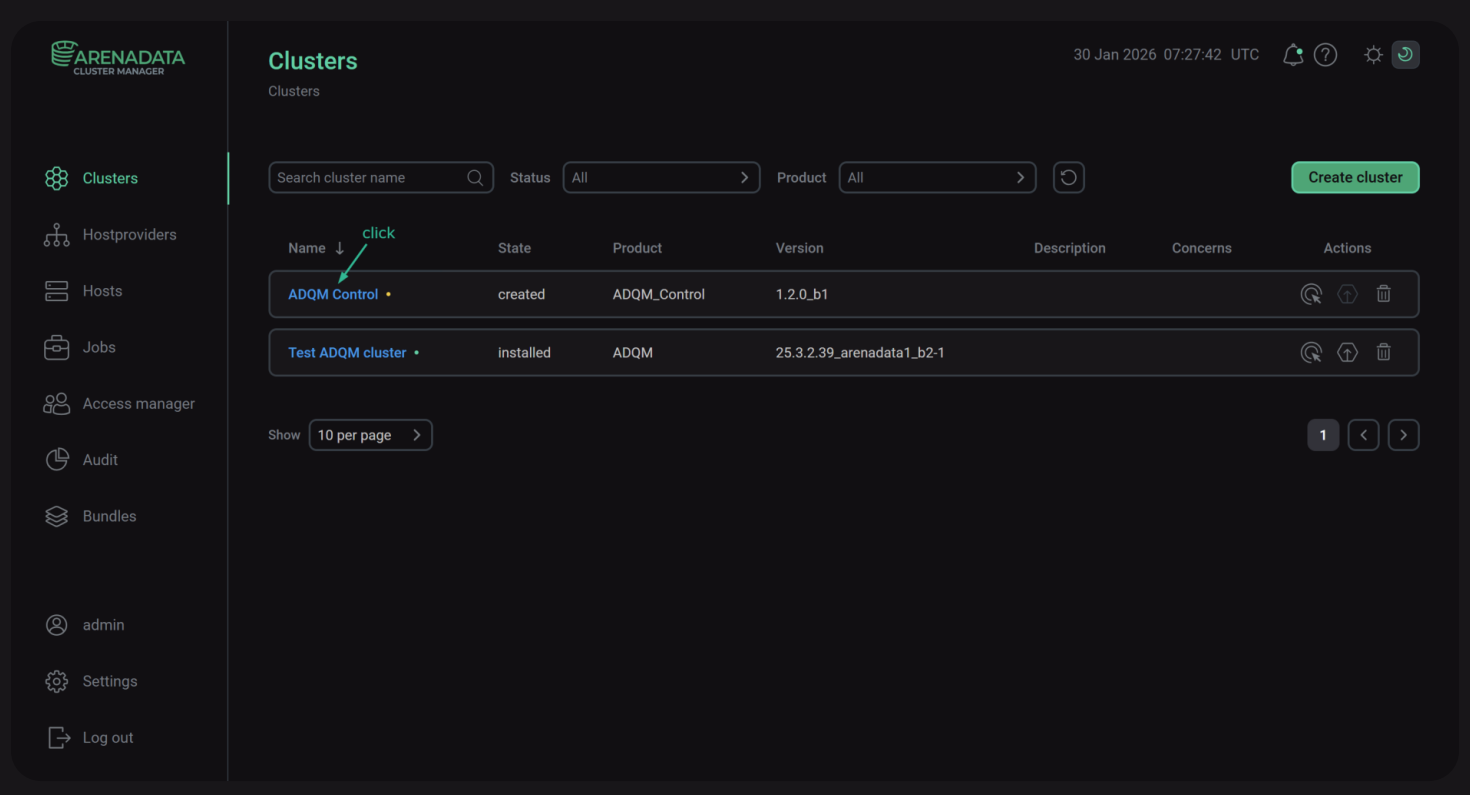
To Table of Contents

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.

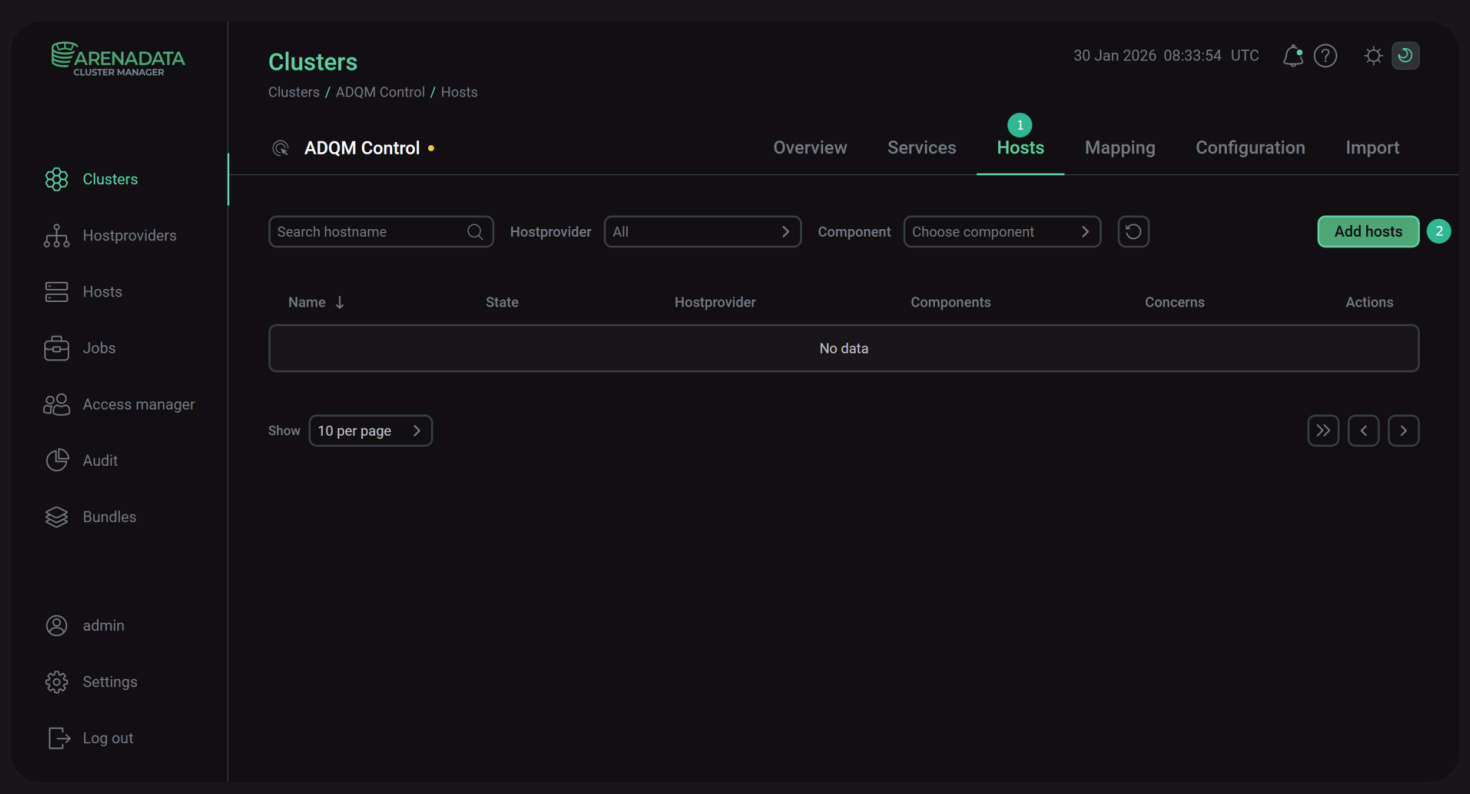
The steps for adding hosts to a cluster are listed below:

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



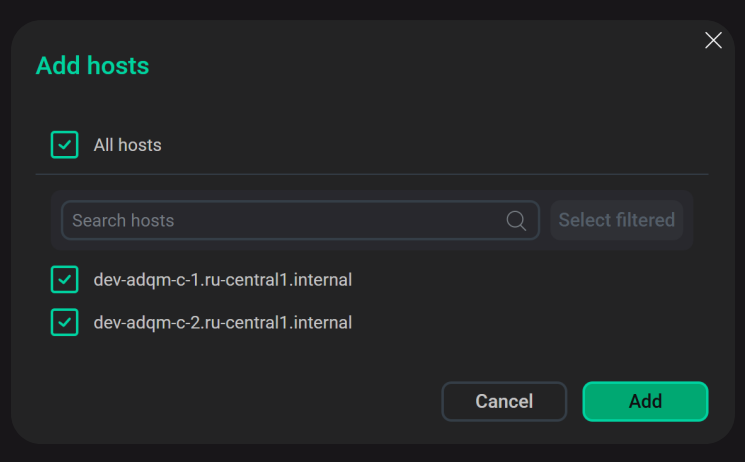
Select a cluster

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



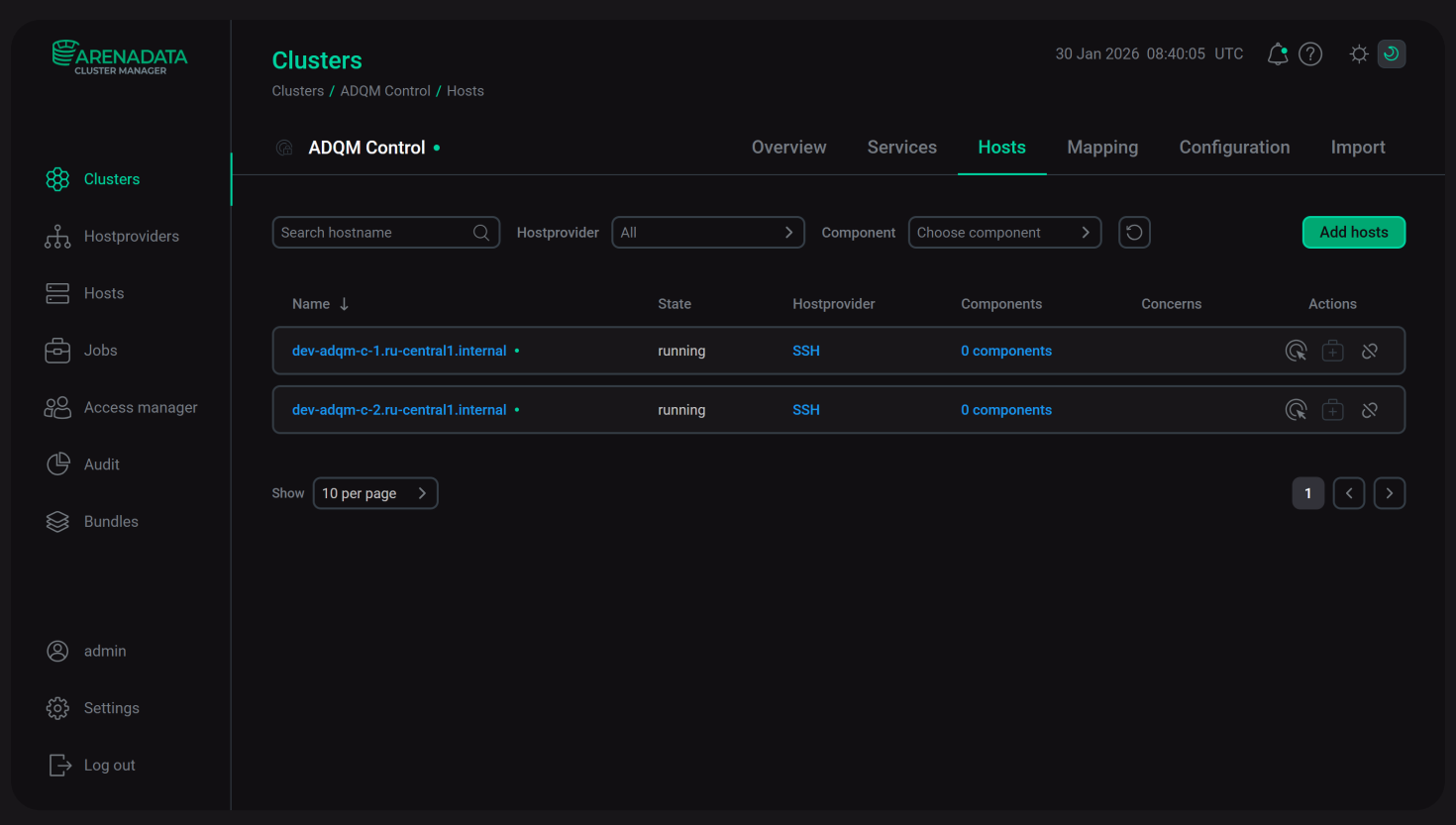
Switch to adding hosts

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 Baryshева

ADQM CONTROL

To Table of Contents

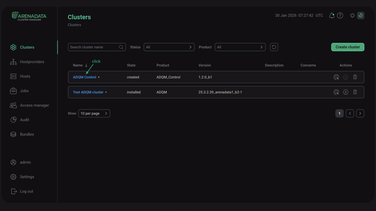
In ADCM, a **component** means a part of a service that should be deployed on one or several cluster hosts.

IMPORTANT

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

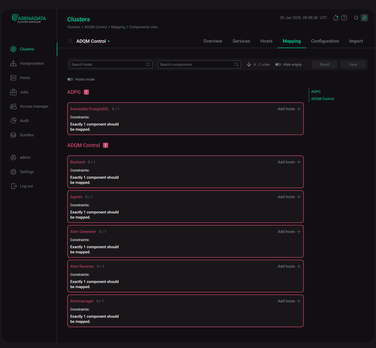
The steps for adding components to hosts are listed below:

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.

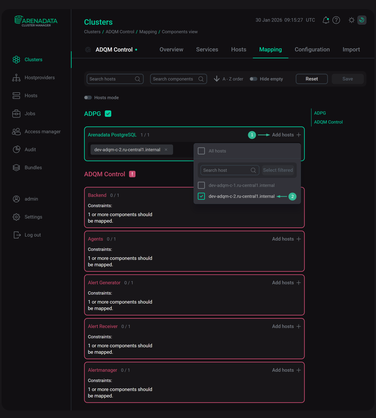
CAUTION

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.

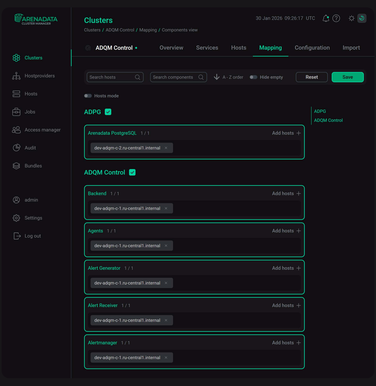


Select hosts for a component

TIP

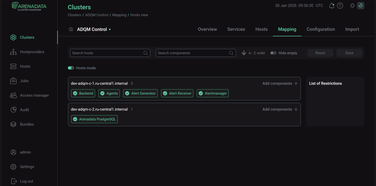
It is recommended to install the Arenadata PostgreSQL component on a dedicated host, separately from the components of the ADQM Control service.

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.



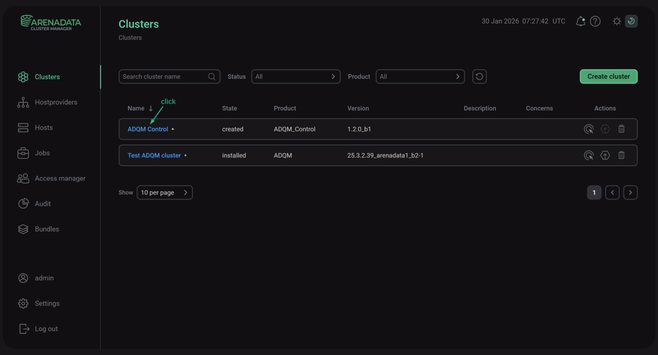
Use mapping by hosts

Configure services

Elena Dvoryadkina, Daria Barysheva

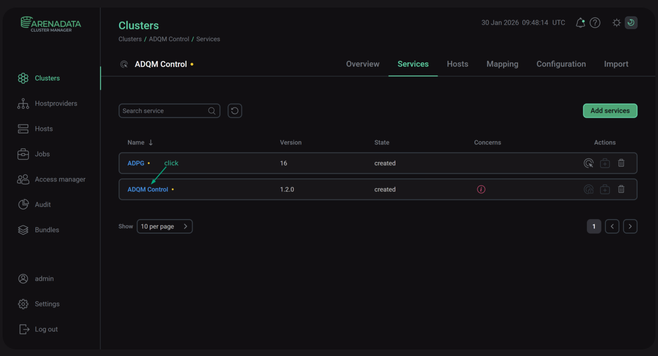
To Table of Contents

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



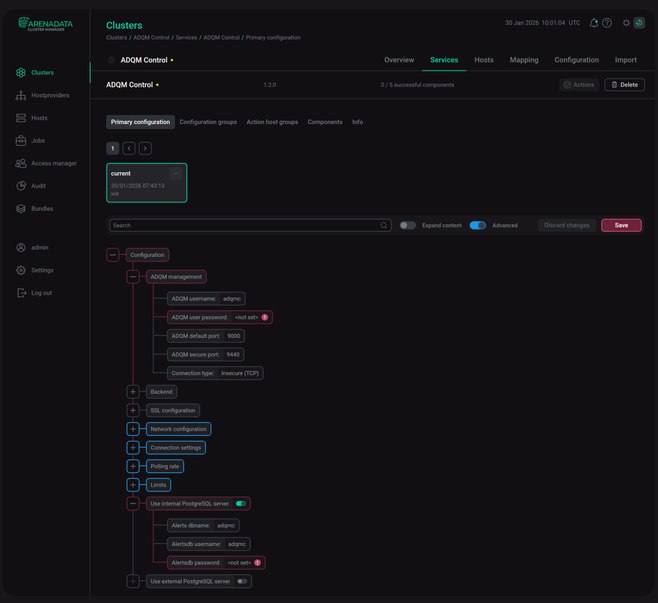
Select a cluster

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

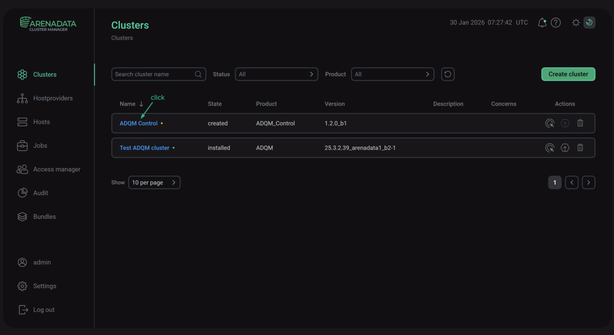
Ansible settings

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

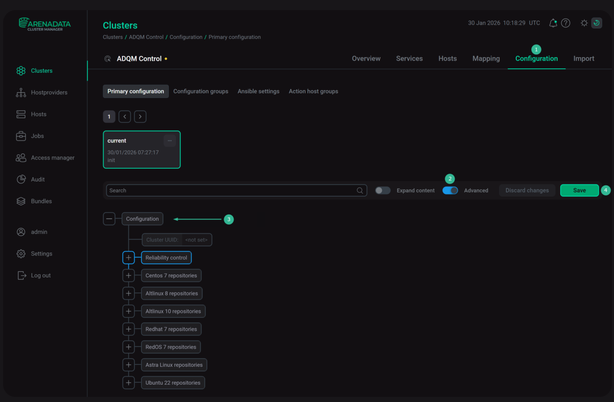
2. Open the ***Configuration*** tab on the cluster page. The ***Configuration*** tab includes the following sections: ***Primary configuration***, ***Configuration groups***, ***Ansible settings***.
3. Fill in all necessary parameters on the selected tab and click ***Save***.

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.

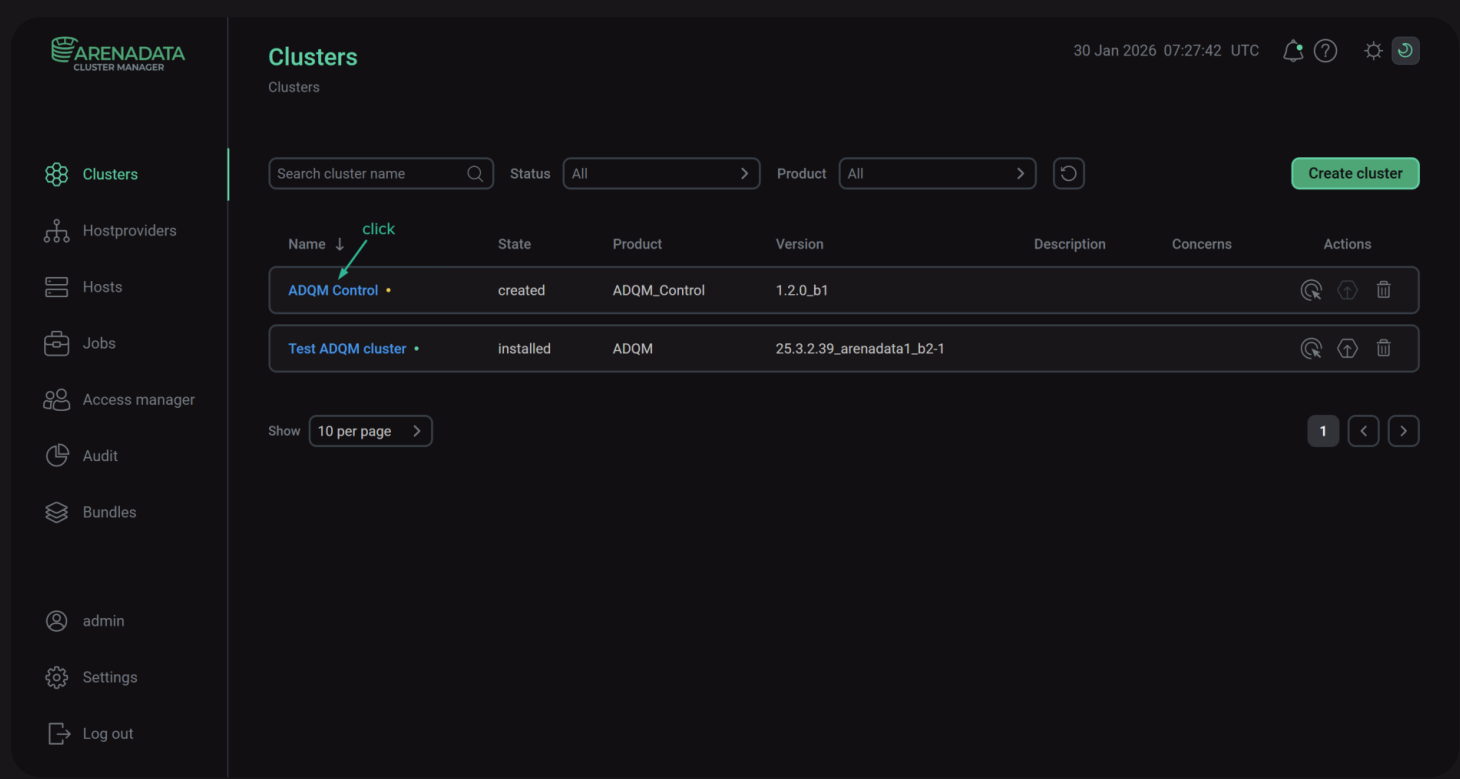
Integrate with an ADQM cluster

Elena Dvoryadkina

To Table of Contents

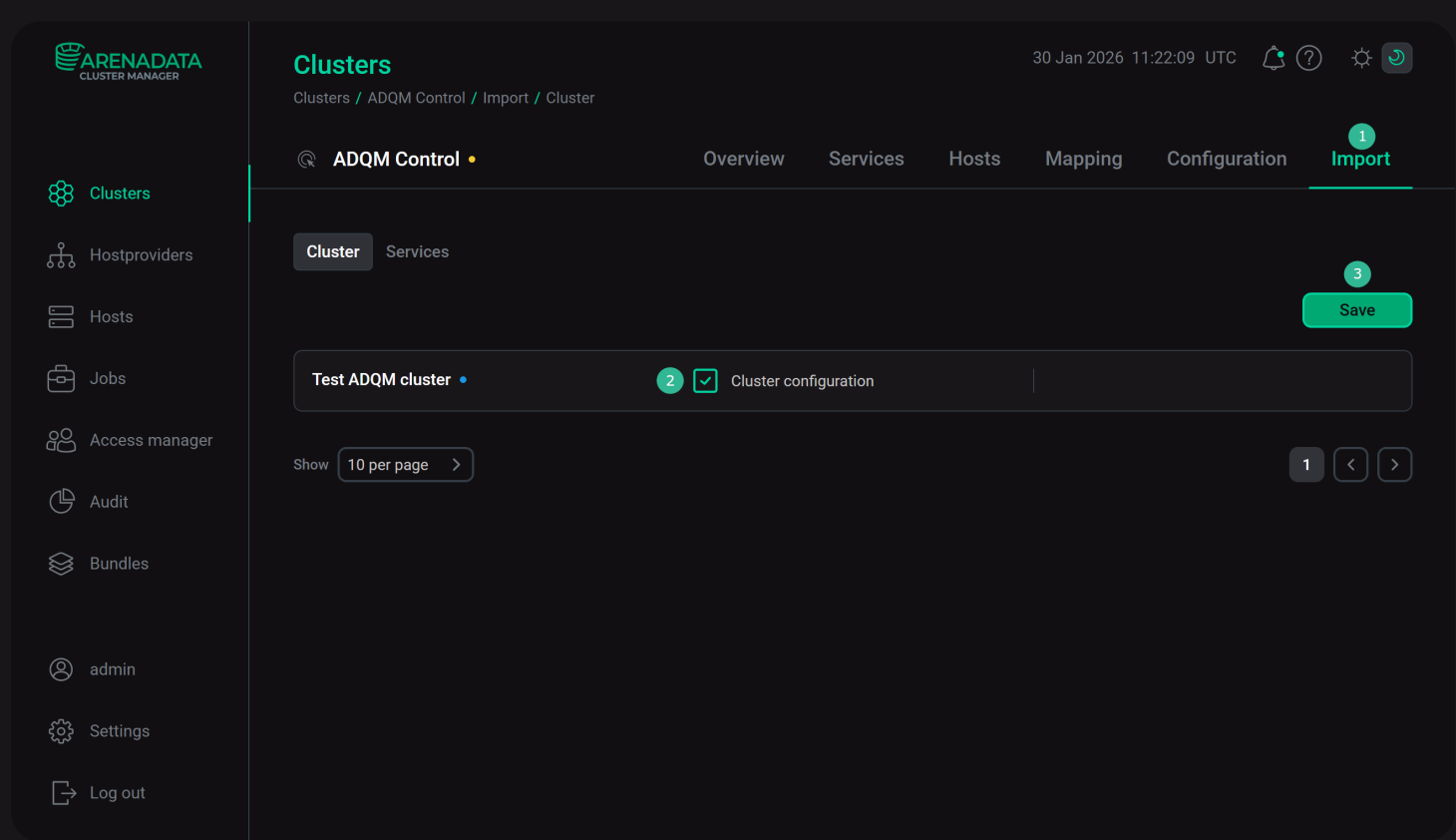
To set up an ADQM Control cluster, you should integrate it with at least one pre-installed ADQM cluster with the Monitoring service configured:

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

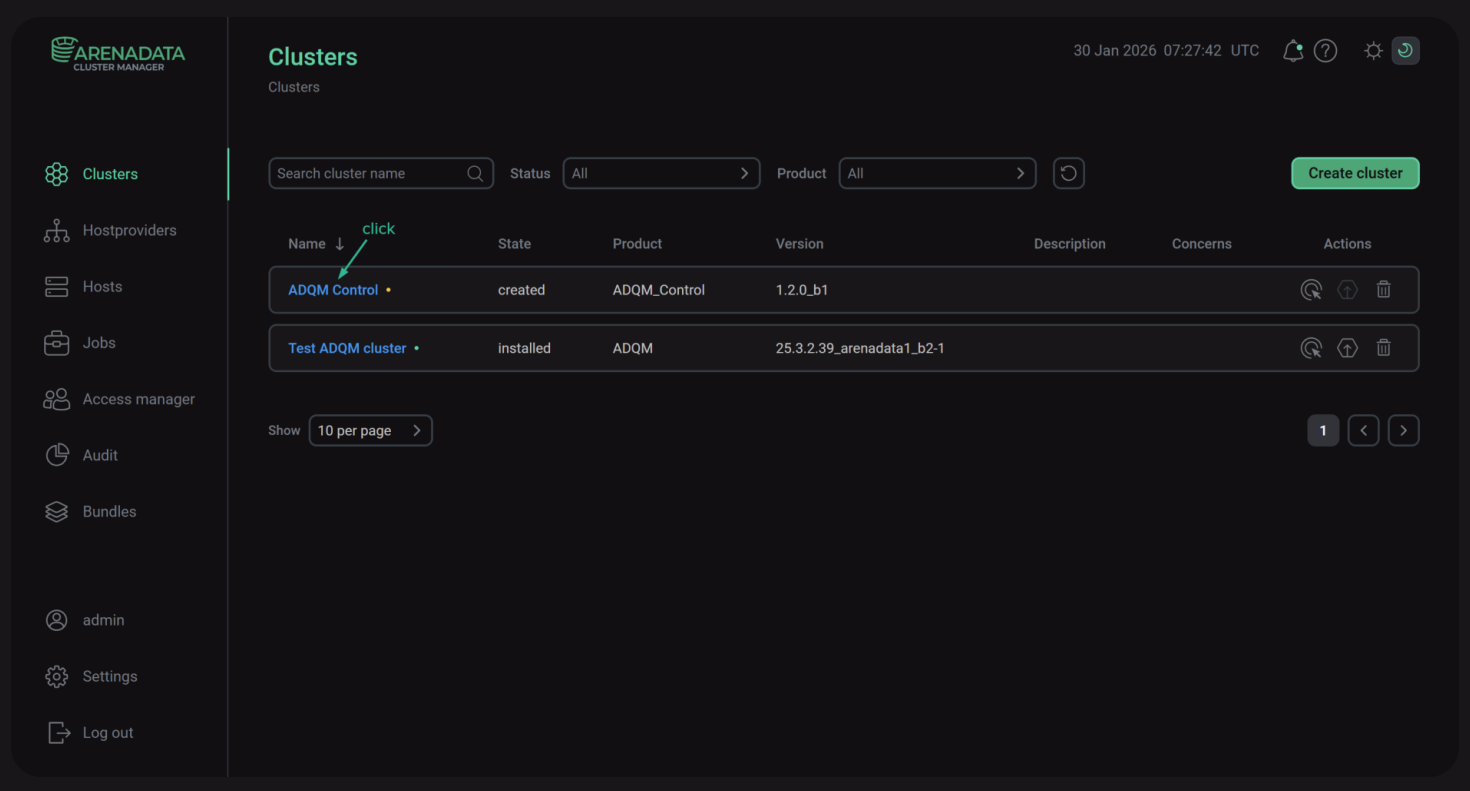
To Table of Contents

IMPORTANT

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

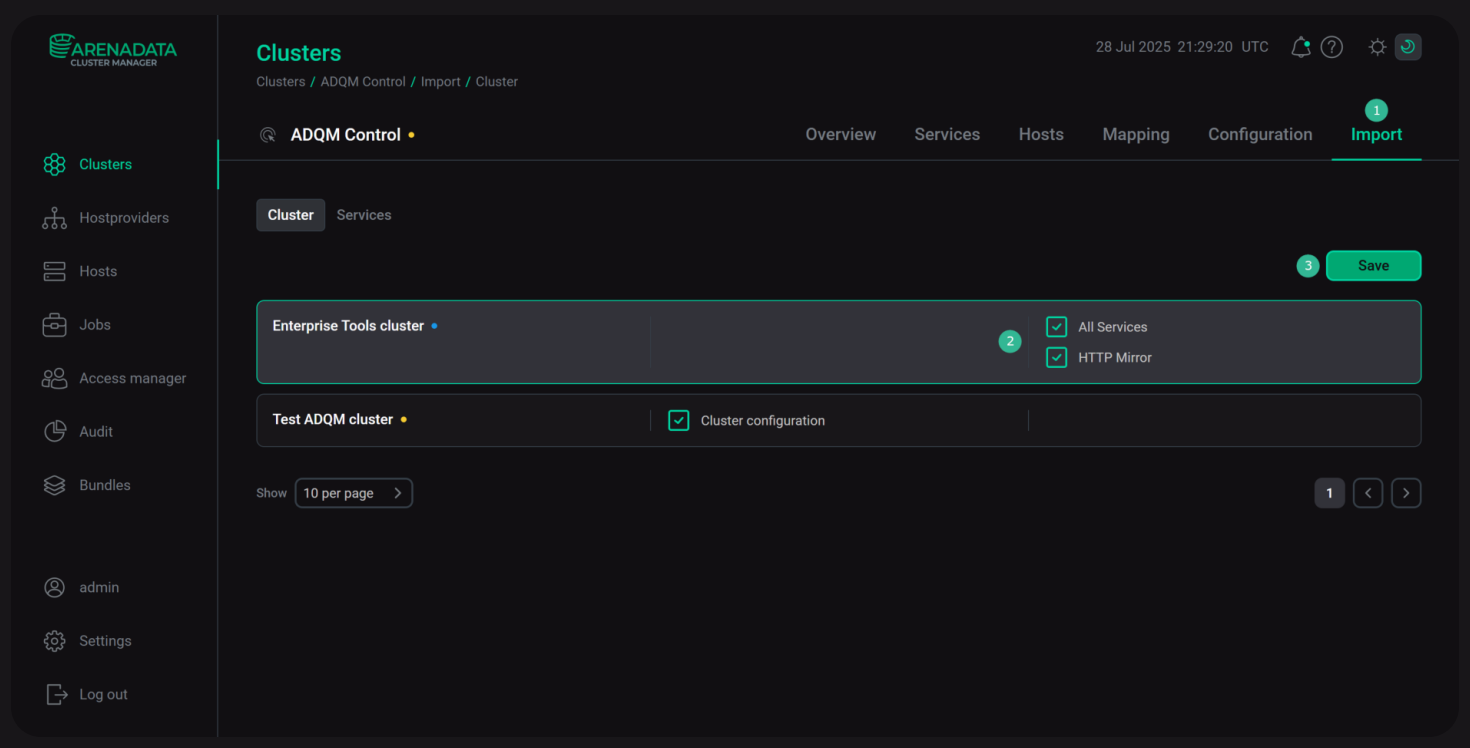
In order to determine that a new cluster should be installed offline, it is necessary to integrate it with the previously created Enterprise Tools (ET) cluster. Follow the steps:

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

A. Elena Dorovydenka, Daria 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.

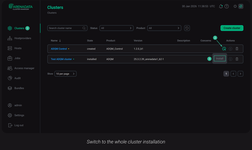
In both cases each service starts automatically after installation.

> **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

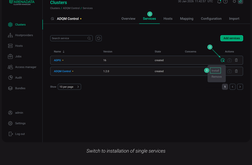
2. Wait until the job is completed.

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.



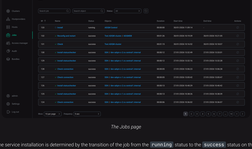
Switch to installation of single services

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

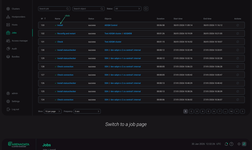
Step 2. View the results of installation

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.

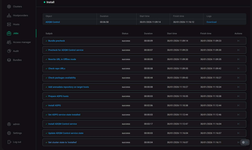


The Jobs page

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

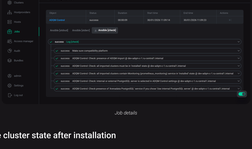


Install cluster job page

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.

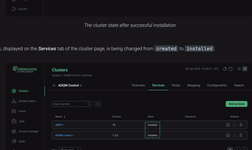


Job details

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`.



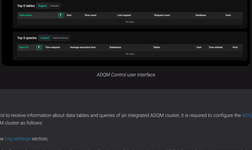
The cluster state after successful installation

- 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 `Owner` 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 ADQM DB 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 ADQM DB service to save configuration changes and restart the service.
>
> - If you have imported an ADQM cluster version up to and including 24.3.11.7.1.b1 into ADQM Control, and the Monitoring service was installed during the installation of the ADQM cluster (i.e. its state ***Install*** action at the cluster level, not by the service actions), 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

A Beta Installation

ADOM CONTROL

Contents

To Table of 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 ADOM Control to collect and store ADOM Control cluster monitoring metrics, and also supports the ability to use the Grafana web application for visualization and analysis of information. This article describes the steps required to add the service to an ADOM Control cluster.

Overview

After adding the monitoring service to ADOM Control, you can install the node exporter on all or selected hosts of your cluster. It is a monitoring agent that reads host system metrics and exports them in a format that can be read by the Prometheus server. The node exporter runs as a Linux service on the specified hosts and listens for service requests on port 9100 by default. You can configure Prometheus. 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-compatible monitoring system set up (for example, your own Prometheus server or VictoriaMetrics), you can use it to collect ADOM Control metrics. To do this, configure access to ADOM Control metrics in your monitoring system using ADOM Control's Prometheus parameters specified on the service configuration.
- You can also use the Installation mechanism to migrate all metrics from the Prometheus server deployed in ADOM Control to your Prometheus.

Step 1. Add the service

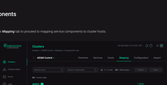
1. In the ADOM Control interface, open the **Clusters** page and click your ADOM Control cluster name. On the cluster page that opens, scroll to the **Services** tab and click **Add services**.

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

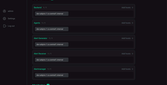
1. On the cluster page, open the **Mapping** tab to proceed to mapping service components to cluster hosts.

Switch to mapping service components

Monitoring service components

Component	Description
Prometheus Server	Stores and allows requesting all metrics for an ADOM Control cluster
Grafana	Visualizes ADOM 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

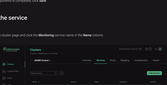
CAUTION

It is not recommended to install the Prometheus Server component on hosts with ADOM Control – use separate hosts for it. Otherwise, if an ADOM 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's configuration parameters.

Configure the monitoring service

Fields highlighted in red are required.

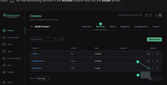
- **Prometheus settings** – *Password for Grafana connection* – a password of a Grafana user (**GRAFANA_WEBPASS**) that is used to connect to Prometheus. The password must contain a combination of numbers, symbols, lowercase and uppercase letters and have a minimum length of 8.
- **Grafana settings** – **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 required parameters, click **Save**.

Step 4. Install the service

1. On the **Services** tab, click the icon for the Monitoring service 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 **Installed** → **Installation**.

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 ADOM Control's services are collected from the cluster hosts – see ADOM Control Metrics in Grafana. To do this, view metrics in the Prometheus format in the browser or use the Prometheus or Grafana web interfaces.

Metrics in the Prometheus format

1. In the address bar of your browser, enter an IP address of a host where the Prometheus Server component is installed. Specify also a port number from the **web.listen-address** parameter in the **Source parameters** section of the **Prometheus settings** — settings for accessing to system metrics of the ADOM Control cluster, 127.0.0.1:9090 – for example, http://127.0.0.1:9090. To log into the Prometheus interface, use a username and password that are also specified in the **Prometheus settings** section – the **Prometheus.com to login/logout to Prometheus** setting.
2. In the **Expression** field, you can enter a metric name and click **Execute** – values of this metric on all hosts of the ADOM Control cluster will be shown in the interface.

Metrics in the Prometheus format

Prometheus web interface

1. In the address bar of your browser, enter an IP address of a host where the Prometheus Server component is installed. Specify also a port number from the **web.listen-address** parameter in the **Source parameters** section of the **Prometheus settings** — settings for accessing to system metrics of the ADOM Control cluster, 127.0.0.1:9090 – for example, http://127.0.0.1:9090. To log into the Prometheus interface, use a username and password that are also specified in the **Prometheus settings** section – the **Prometheus.com to login/logout to Prometheus** setting.
2. In the **Expression** field, you can enter a metric name and click **Execute** – values of this metric on all hosts of the ADOM 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 where the Grafana component is installed. Specify also a port number from the **web.listen-address** parameter in the **Grafana settings** section of the Monitoring service configuration (the default value is 127.0.0.1:3000). For example, http://127.0.0.1:3000. To log in, use **admin** as a username, and the **Grafana administrator's password** parameter value (that specified 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 **ADOM_Dashboard**/**ADOM_Control_Cluster_name** section. In this section, you can select one of the available dashboards by name and type a search metrics in the corresponding field in the search bar.

View ADOM Control metrics in Grafana

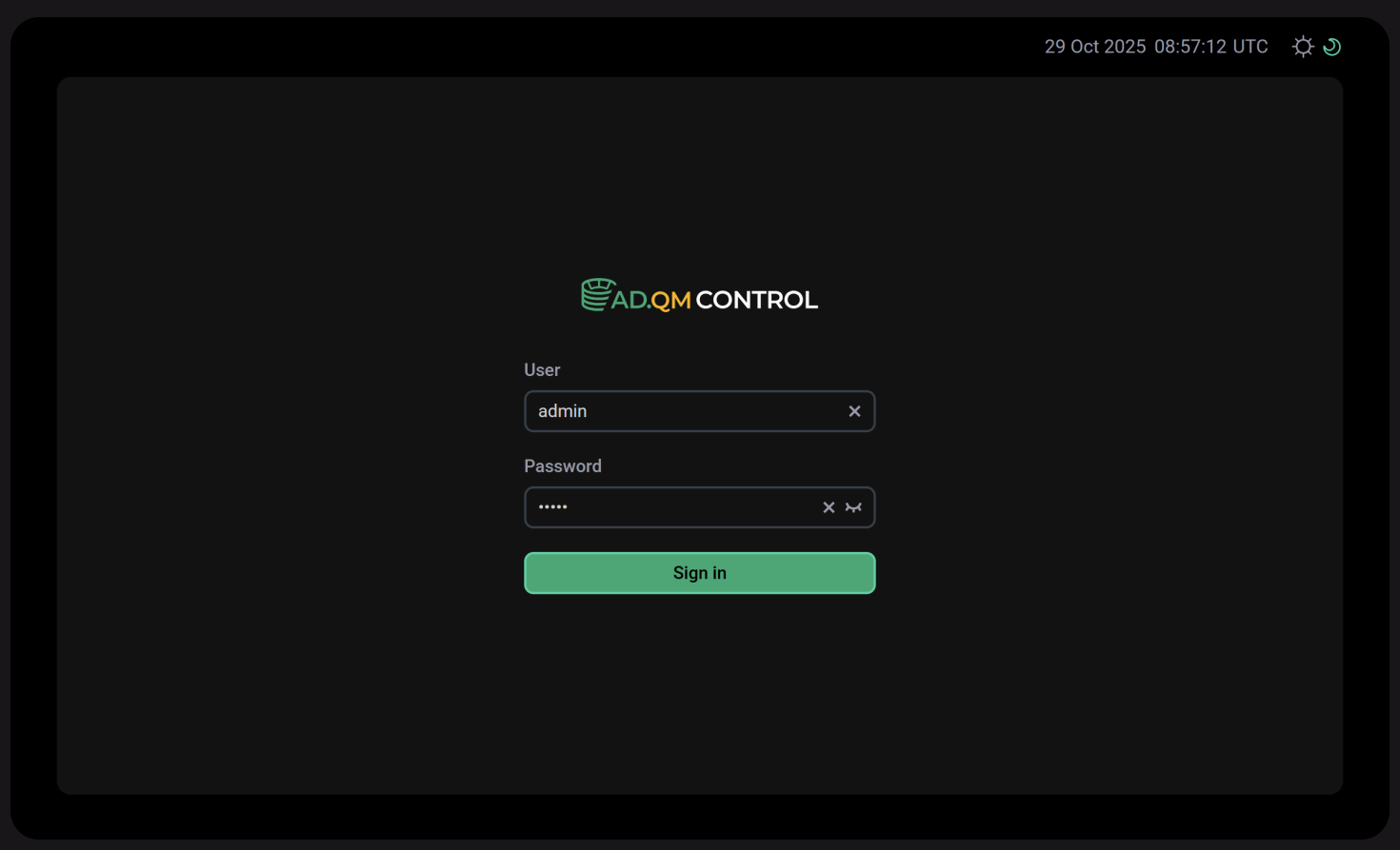
Connect to ADQM Control

Elena Dvoryadkina

To Table of Contents

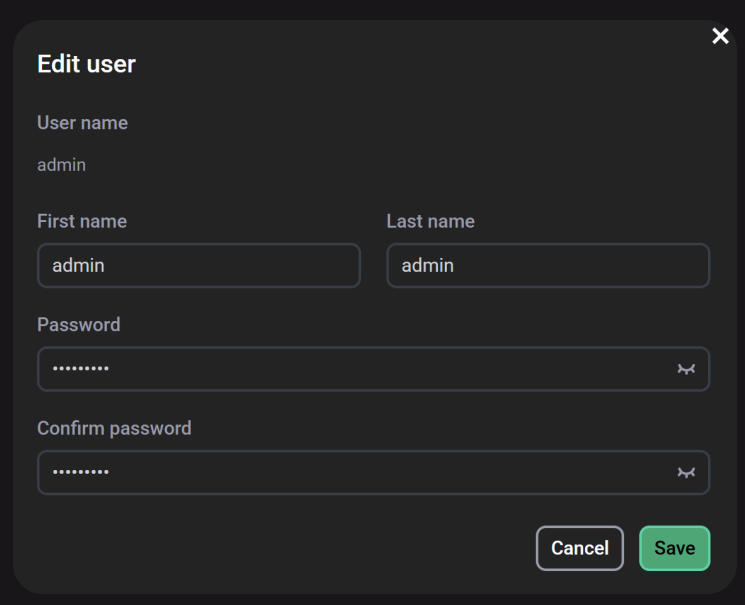
To connect to the ADQM Control web interface, follow the steps:

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

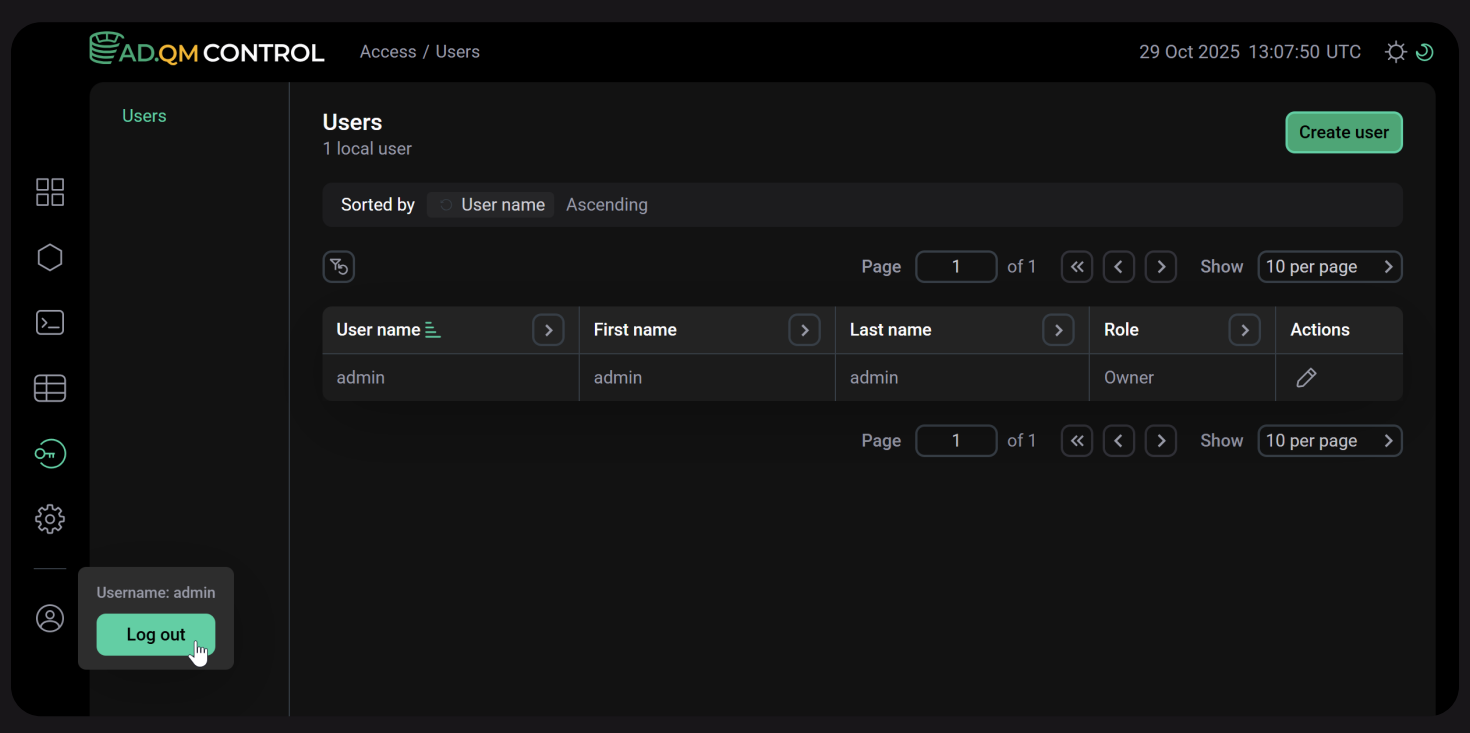
3. It is recommended to assign a new password to the `admin` user instead of the default one. To do this, go to the ***Access*** → ***Users*** page, open the edit window for the `admin` user, set a new password in the ***Password*** and ***Confirm password*** fields, and click ***Save***.



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

AD.QM CONTROL

Contents

To Table of Contents

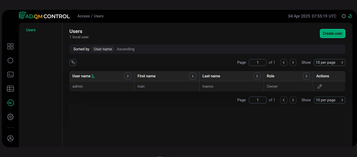
View a list of users
Add a user
Edit a user
Set password requirements

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

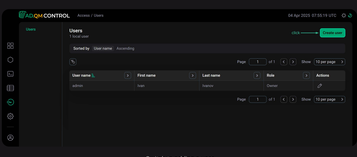
In column headers of the table with a list of users, there are filters that you can use to select specific data. To open a filter, click the > icon.

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:

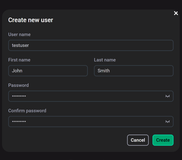
1. On the ***Access*** → ***Users*** page, click ***Create user***.



Switch to adding a user

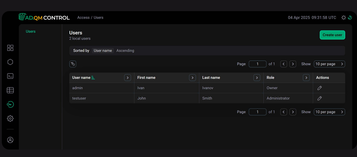
2. In the window that opens, fill in the following fields:
   - ***User name*** — user's login;
   - ***First name*** — user's name;
   - ***Last name*** — user's last name;
   - ***Password*** — user's password.

> **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.

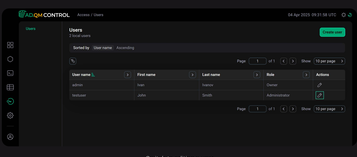


User is added

Edit a user

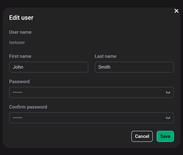
To edit user data, follow the steps:

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.



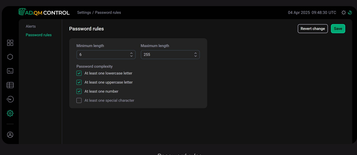
Edit user data

> **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

Contents

Create SSL certificates

Configure ADOM Control to enable SSL encryption

- Manage SSL clusters
- SSL parameters in service configurations
- Example

Test a connection

Disable SSL

To Table of Contents

Create SSL certificates

To create client-server SSL certificates, use the OpenSSL utility, or the OpenSSL-compatible CA, to create a CA certificate and a server certificate.

For each node of the ADOM Control cluster, create a server certificate with the following parameters:

```
$ openssl req -newkey rsa:2048 -nodes \
-keyout <node_name>.key -out <node_name>.csr \
-subj "/C=CA/ST=Ontario/L=Ottawa/O=Fortinet/OU=FortiAnalyzer/CN=<node_name>/emailAddress=<email_address>"
$ openssl x509 -req -days 365 -in <node_name>.csr \
-CA ca.crt -CAkey ca.key -CAcreateserial \
-out <node_name>.crt
$ openssl rsa -in <node_name>.key -out <node_name>.key.nopass
$ openssl pkcs12 -export -inkey <node_name>.key.nopass \
-in <node_name>.crt -certfile ca.crt \
-out <node_name>.p12 -name "<node_name>"
$ openssl rsa -check -in <node_name>.key.nopass
$ openssl x509 -in <node_name>.crt -noout -text
$ openssl pkcs12 -info -in <node_name>.p12 -nodes -nomac
```

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

```
$ openssl x509 -req \
-in <node_name>.csr \
-CA ca.crt -CAkey ca.key \
-CAserial ca.srl -extfile v4.ext \
-out <node_name>.crt -days 365 -sha256 \
-copy_extensions copyall
```

For maximum security, the server certificate in this example uses an expiry. For some enrollment tools such as the certificate in the example of configuring SSL for a ADOM Control cluster, the server certificate does not expire. Consult your enrollment tool documentation for information on how to create a server certificate with an unlimited or long expiry.

Copy the CA server certificate, server certificate, and the CA certificate for the client-server connection to a location that is accessible by the FortiAnalyzer. Typically, the CA certificate and the server certificate are created on the primary node of the ADOM Control cluster. The CA certificate and the server certificate are then copied to the secondary nodes. The CA certificate and the server certificate are then copied to the FortiAnalyzer. Depending on your environment, the CA certificate and the server certificate may be created on the FortiAnalyzer. In this case, the CA certificate and the server certificate are then copied to the ADOM Control cluster.

In the Linux environment, the SSL certificates are stored in the /var/lib/ssl directory. Depending on your environment, the certificates are stored in the /var/lib/ssl/certs/ or /var/lib/ssl/private/ directories. Consult your Linux administrator for information on the location of the SSL certificates.

- For the Backend System, Admin Services, and Alert Receiver components of the ADOM Control service, copy files to the root user's CA and private key trust store:
  - /etc/pki/ca-trust/source/anchors/
  - /etc/pki/ca-trust/extracted/pem/tls-ca-bundle.pem
  - /etc/pki/ca-trust/extracted/openssl/ca-bundle.trust.crt
- For the Alert Manager component of the ADOM Control service, copy files to the root user's CA and private key trust store:
  - /usr/share/ssl/certs/ca-bundle.crt
  - /usr/share/ssl/certs/ca-bundle.trust.crt
  - /etc/pki/tls/certs/ca-bundle.crt
  - /etc/pki/tls/certs/ca-bundle.trust.crt
- For the Alert Manager component of the ADOM Control service, copy files to the root user's CA and private key trust store:
  - /usr/share/ssl/certs/ca-bundle.crt
  - /usr/share/ssl/certs/ca-bundle.trust.crt
  - /etc/pki/tls/certs/ca-bundle.crt
  - /etc/pki/tls/certs/ca-bundle.trust.crt

For instructions, consult your Linux administrator documentation.

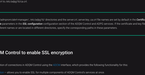
Configure ADOM Control to enable SSL encryption

You can enable SSL encryption of connections in ADOM Control using the FGT interface, which provides the following functionality for the:

- Manage SSL cluster, which allows you to enable SSL encryption of ADOM Control cluster services at one
- SSL parameters in service configurations, which allows you to configure SSL for each service independently, so you can change the default ports, or certificates and keys.

Manage SSL cluster action

1. Select the **Cluster** page. Select the cluster. In the **Cluster Action** tab, in the **Cluster Action** section, select **Manage SSL**.



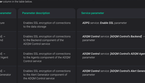
Launch the "Manage SSL" action

ADOM will display a dialog window. When you set the **Enable SSL** option, it prompts options 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 in the **Manage SSL** action configuration window:

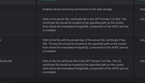
Parameter	Description
Enable SSL	Enable SSL encryption of connections between ADOM Control cluster members.
Certificate File	Select the SSL certificate file in the FGT format (.crt file). The certificate file should be located in the specified path on the FGT. The certificate file should be the same for all cluster members. The certificate file should be a trusted certificate.
Private Key File	Select the SSL private key file in the FGT format (.key file). The private key file should be located in the specified path on the FGT. The private key file should be the same for all cluster members. The private key file should be a trusted certificate.
Certificate Authority File	Select the SSL certificate file in the FGT format (.crt file). The CA certificate file should be located in the specified path on the FGT. The CA certificate file should be the same for all cluster members. The CA certificate file should be a trusted certificate.



ADOM Control configuration parameters in the SSL certificate of connections

3. Fill in the **Configuration** section of the ADOM Control configuration. The following parameters are available:

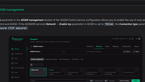
Parameter	Description	Default value
Enable SSL	Enable secure connecting connections to the data storage	Disabled
Certificate File	Select the SSL certificate file in the FGT format (.crt file). The certificate file should be located in the specified path on the FGT. The certificate file should be the same for all cluster members. The certificate file should be a trusted certificate.	/etc/pki/tls/certs/v4.crt
Private Key File	Select the SSL private key file in the FGT format (.key file). The private key file should be located in the specified path on the FGT. The private key file should be the same for all cluster members. The private key file should be a trusted certificate.	/etc/pki/tls/certs/v4.key
Certificate Authority File	Select the SSL certificate file in the FGT format (.crt file). The CA certificate file should be located in the specified path on the FGT. The CA certificate file should be the same for all cluster members. The CA certificate file should be a trusted certificate.	/etc/pki/tls/certs/v4.crt



ADOM Control configuration parameters in the SSL certificate of connections

ADOM Control SSL configuration

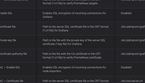
Parameter	Description	Default value
ADOM Control's Backend - Enable SSL	Enable SSL encryption of incoming connections for the Backend Component	Disabled
ADOM Control's ADOM Agent - Enable SSL	Enable SSL encryption of incoming connections for the Agents Component	Disabled
ADOM Control's Alert Receiver - Enable SSL	Enable SSL encryption of incoming connections for the Alert Receiver Component	Disabled
ADOM Control's Alert Receiver - Enable SSL	Enable SSL encryption of incoming connections for the Alert Receiver Component	Disabled
ADOM Control Service's - Certificate File	Path to the file with the private key of the server SSL certificate ("key" file in the Linux directory). This key file should be located in the specified path on the FGT. The key file should be the same for all cluster members. The key file should be a trusted certificate.	/etc/pki/tls/certs/v4.crt
ADOM Control Service's - Private Key File	Path to the file with the private key of the server SSL certificate ("key" file in the Linux directory). This key file should be located in the specified path on the FGT. The key file should be the same for all cluster members. The key file should be a trusted certificate.	/etc/pki/tls/certs/v4.key
ADOM Control Service's - Certificate Authority File	Path to the file with the CA certificate in the FGT format (.crt file) for the Certificate Authority (CA) that issues the SSL certificates. This CA certificate file should be located in the specified path on the FGT. The CA certificate file should be the same for all cluster members. The CA certificate file should be a trusted certificate.	/etc/pki/tls/certs/v4.crt
ADOM Control Service's - Certificate Authority Bundle File	Path to the file with the CA certificate in the FGT format (.crt file) for the Certificate Authority (CA) that issues the SSL certificates. This CA certificate file should be located in the specified path on the FGT. The CA certificate file should be the same for all cluster members. The CA certificate file should be a trusted certificate.	/usr/share/ssl/certs/ca-bundle.crt
Manager - Certificate File	Path to the file with the CA certificate in the FGT format (.crt file) for the Manager component. This CA certificate file should be located in the specified path on the FGT. The CA certificate file should be the same for all cluster members. The CA certificate file should be a trusted certificate.	/usr/share/ssl/certs/ca-bundle.crt
Manager - Private Key File	Path to the file with the private key of the Manager component. This key file should be located in the specified path on the FGT. The key file should be the same for all cluster members. The key file should be a trusted certificate.	/usr/share/ssl/private/ca-bundle.key
Manager - Certificate Authority File	Path to the file with the CA certificate in the FGT format (.crt file) for the Certificate Authority (CA) that issues the SSL certificates. This CA certificate file should be located in the specified path on the FGT. The CA certificate file should be the same for all cluster members. The CA certificate file should be a trusted certificate.	/usr/share/ssl/certs/ca-bundle.crt



Configured parameters of the ADOM Control service in Basic SSL certificate of connections

ADOM Control, ADOM Agent configuration

The **Connection type** parameter in the **ADOM management** section of the ADOM Control service configuration, allows you to enable the use of only secure TCP connections between ADOM Control and ADOM Agent. **Secure** and **Secure Preferred** options are available in the **Connection type** parameter. If ADOM Control **Connection type** parameter is set to **Secure** (TCP secure), ADOM Agent **Connection type** parameter, if ADOM Control is set to **Secure** (TCP secure).



Enable the use of only secure TCP connections in ADOM

Monitoring SSL configuration

In the Monitoring section of the **Configuration** section of the Monitoring service configuration allows you to enable SSL encrypted connections to service components. Certificates keys files should be located in the specified path, or all cluster nodes where the corresponding components of monitoring are installed.

Parameter	Description	Default value
[Keystore/ssl] - Enable SSL	Enable SSL encryption of incoming connections for the Keystore component	Disabled
[Keystore/ssl] - Certificate file	Path to the file with the SSL certificate in the FGT format (.crt file) for the Keystore component	/etc/pki/tls/private/keystore.crt
[Keystore/ssl] - Private key file	Path to the file with the private key of the server SSL certificate ("key" file in the Linux directory)	/etc/pki/tls/private/keystore.key
[Keystore/ssl] - Certificate authority file	Path to the file with the CA certificate in the FGT format (.crt file) for the Keystore component	/etc/pki/tls/private/keystore.crt
[Gateway] - Enable SSL	Enable SSL encryption of incoming connections for gateway	Disabled
[Gateway] - Certificate file	Path to the file with the SSL certificate in the FGT format (.crt file) for the Gateway component	/etc/pki/tls/private/gateway.crt
[Gateway] - Private key file	Path to the file with the private key of the server SSL certificate ("key" file in the Linux directory)	/etc/pki/tls/private/gateway.key
[Gateway] - Certificate authority file	Path to the file with the CA certificate in the FGT format (.crt file) for the Gateway component	/etc/pki/tls/private/gateway.crt
[kcaa-exporter] - Enable SSL	Enable SSL encryption of incoming connections for kcaa-exporter	Disabled
[kcaa-exporter] - Certificate file	Path to the file with the SSL certificate in the FGT format (.crt file) for the kcaa-exporter component	/etc/pki/tls/private/kcaa-exporter.crt
[kcaa-exporter] - Private key file	Path to the file with the private key of the server SSL certificate ("key" file in the Linux directory)	/etc/pki/tls/private/kcaa-exporter.key

Configuration parameters of the Monitoring service for SSL connections

3. After specifying all the necessary parameters for configuring SSL connections, click **Save** and then **Revert/Save and Install** in the upper-right menu bar.

Example

1. In the **ADOM management** section of the ADOM Control service configuration, set the **Connection type** parameter to **Secure** (TCP secure) and then click **Save**. ADOM Control will restarts its ADOM Agent and TCP connections will be secure. **(ADOM secure port)**.

Enable the use of only secure TCP connections in ADOM

Test **Save**

2. In **Manage SSL** action for the cluster, in the configuration window that appears, uncheck **Enable SSL** and select one the following options to encrypt traffic between ADOM:
   - **[ADFS] - Secure HTTS**
   - **[ADOM Control] - Secure Backend**
   - **[ADOM Control] - Secure ADOM Agent**
   - **[ADOM Control] - Secure Alert Receiver**
   - **[ADOM Control] - Secure Alert Manager**
   - **[ADOM Control] - Secure Alert Receiver**
   - **Certificate authority file** - Path to the file with the CA certificate in the FGT format (.crt file).

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

Disable SSL with the "Manage SSL" action

Test a connection

To test that ADOM Control works in secure mode on HTTPS, you can open a web interface at https://10.0.1.200:443 in the browser or the IP address of the FortiGate that the Management protocol is connected. Ensure that your web browser trusts the CA certificate.

Disable SSL

To disable SSL, select the **Manage SSL** action and the **Enable SSL** option in the configuration of a cluster.

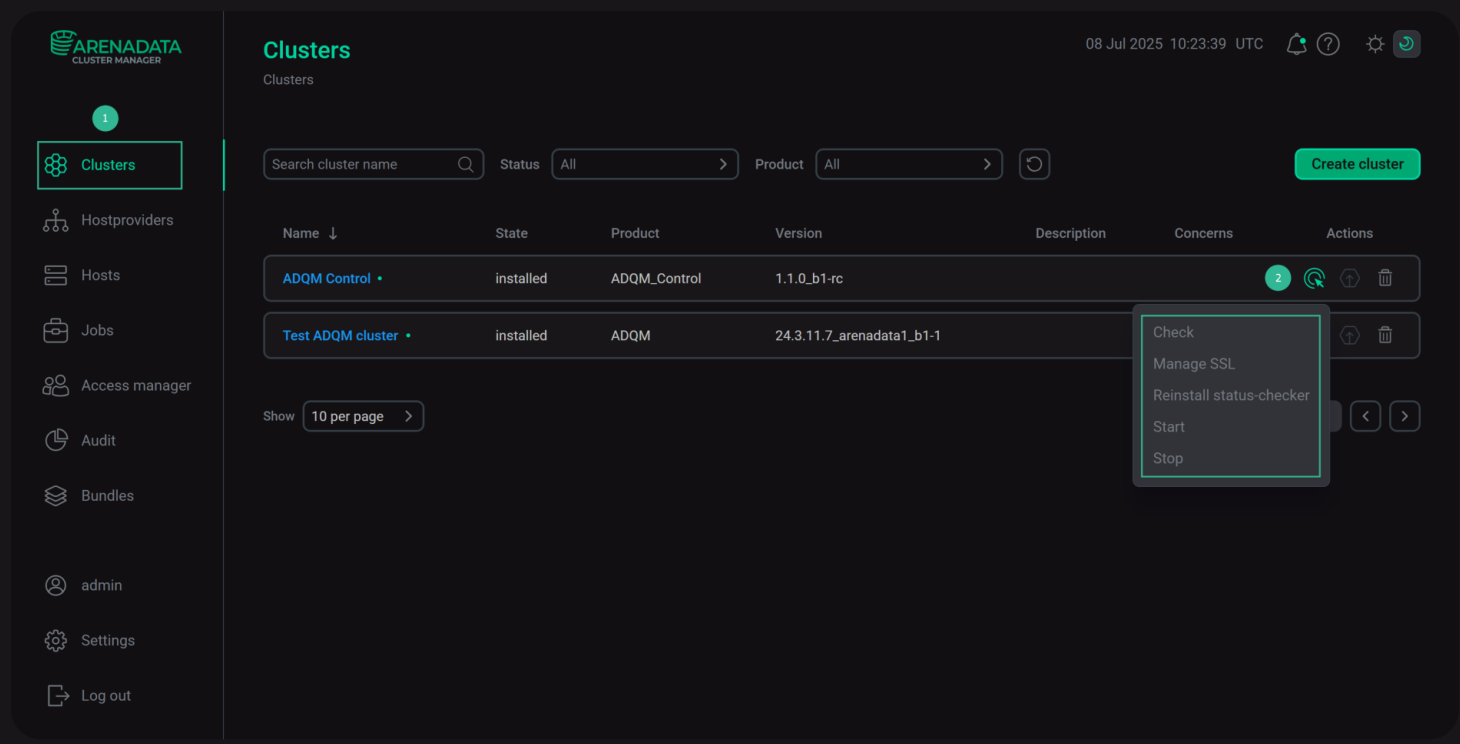
Disable SSL with the "Manage SSL" action

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.

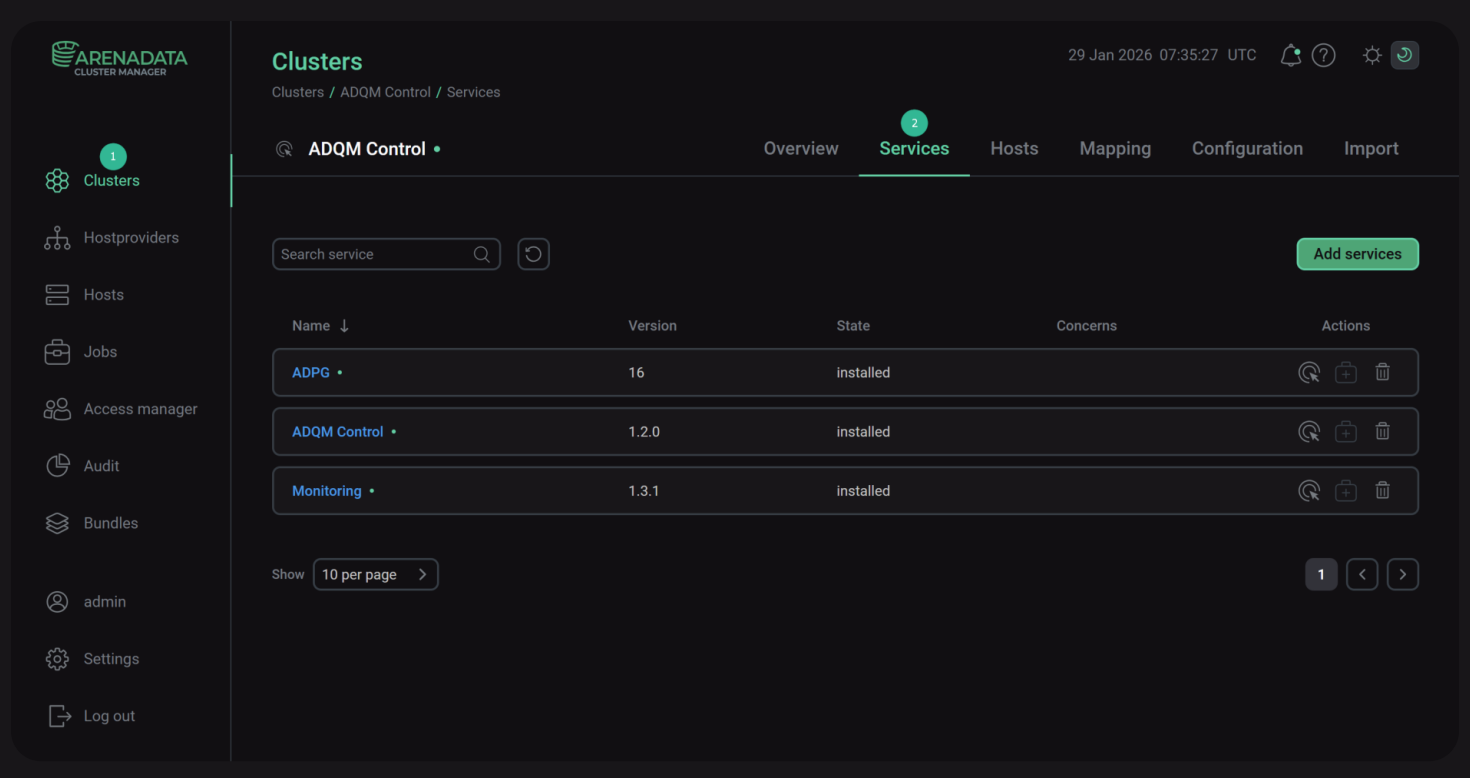
A set of actions available for an ADQM Control cluster depends on the cluster state:

- 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

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:

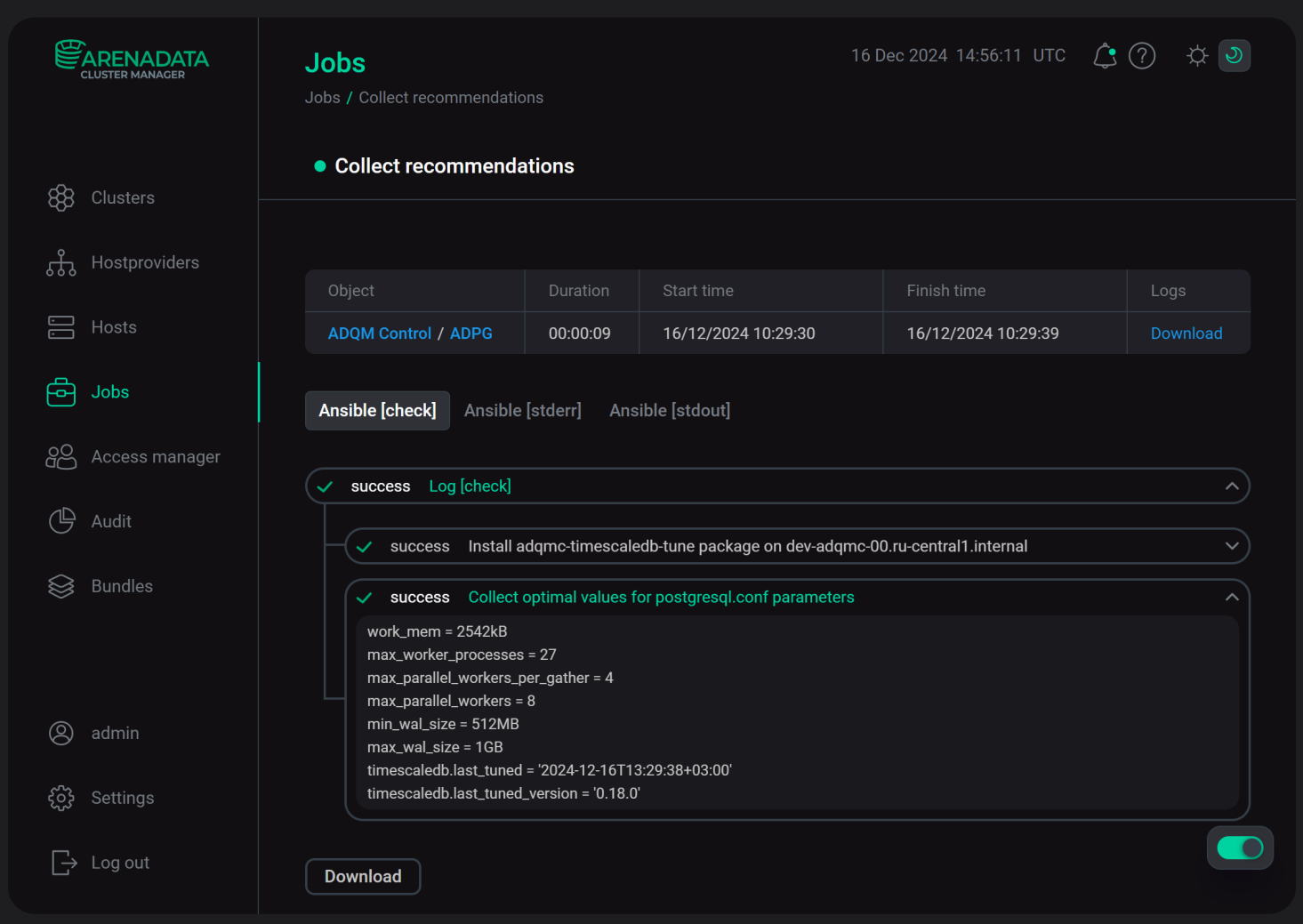
ADQM Control

- ***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

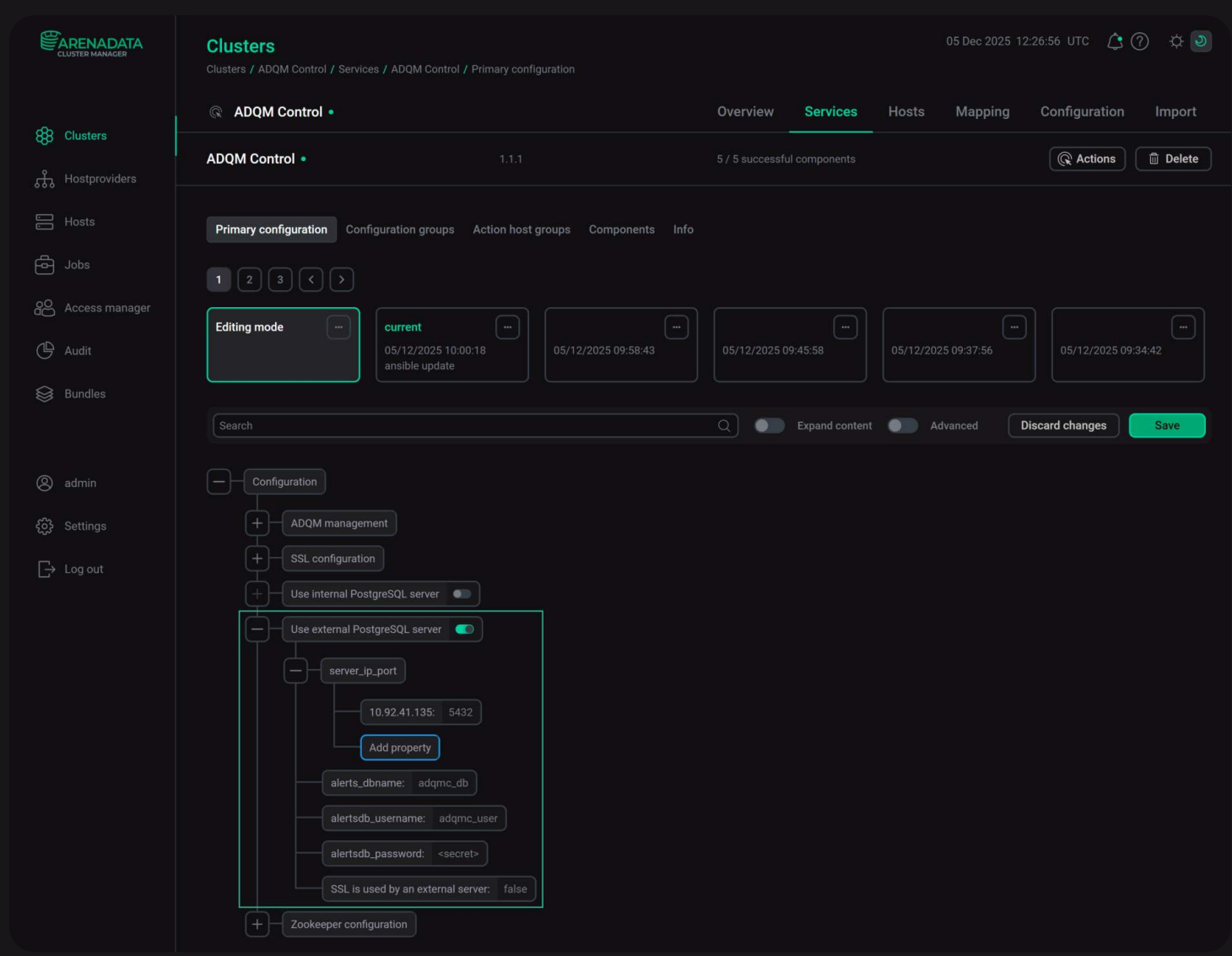
When using higher versions, compatibility testing is required to ensure correct system operation.

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***.

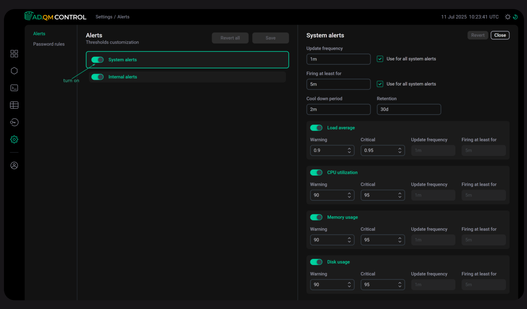
Alerts in ADQM Control are grouped into modules. Currently, ADQM Control can send and allows you to configure alerts of the System alerts and Internal alerts modules.

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.

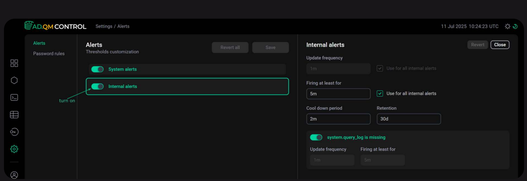
System alerts

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 — 0.9, Critical — 0.95
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 if the `system.query_log` log table with information about executed queries does not exist in 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;
- ***Firing at least for*** — time during which the log table should be missing for the corresponding alert to be 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.

View a dashboard

ADQMCONTROL

Version: 1.0 | Updated: 2022-03-31

Contents

To Table of Contents

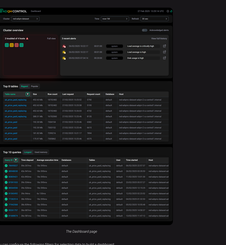
Cluster overview
- Heat map
- Recent alerts
- Acknowledged alerts
- Open alert details
- Go to the Cluster metrics page

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 for which the most queries were sent.
- Top 10 queries – 10 queries that took the longest to execute and 10 queries that used the most memory.



The Dashboard page

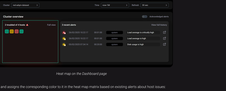
In the fields at the top of the screen, you can configure the following filters for selecting data to build a dashboard:

- **Cluster** – ADQM cluster for which information should be displayed.
- **Time** – time period for which information is required. In the entries 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.

Cluster overview

Heat map

A heat map visualizes information about the states of all hosts in the ADQM cluster – each host is shown 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 hosts based on existing alerts about host issues:

- there are no alerts about any issues with a host in ADQM Control (a healthy host);
- 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 about those issues have been generated in ADQM Control;
- 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.

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 that importance.

Recent alerts on the Dashboard page

Alert severity levels

Alert severity level	Description	Condition for generating an alert
warning	A potential problem has been detected on a host	A system metric value exceeds a threshold set via the **Warning** parameter in the System alerts configuration
critical level of importance	A critical problem has been detected on a host	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, click **View full history** in the top right corner within the **Recent alerts** section – it will open the Alert history tab on the **Cluster metrics** page.

Acknowledged alerts

The **Acknowledged alerts** (alert allows you to specify which alerts about problems on ADQM cluster hosts 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** – alerts, on which you have set an acknowledged. ADQM Control highlights relevant alerts (requiring your attention to be resolved) with the bell sign in their icons.

For example, no issues were detected for the specified period of time on the second host of the cluster (it is selected in the image below) or all alerts were set as acknowledged – in this case, with the **All** / **Acknowledged alerts** option enabled, this host is filled with green in **Heat map** and the **Recent alerts** list is empty.

A selected host has no issues that require analysis

Then, the new **Disk usage is high** acknowledgement appeared 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 is yellow in **Heat map** as shown in the following image.

Example of an unresolved issue on the host

If the disk usage issue has been analyzed and resolved, it no longer requires attention and you can add the corresponding alert to the list of acknowledged alerts (on the page with alert details – see the next section). 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 exclude the alert from the **Recent alerts** list for the host.

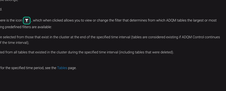
If the **Acknowledged alerts** option is enabled, the **Recent alerts** list shows the latest alerts among all alerts (active and "closed") generated during the specified time period, and all alerts affect the heat map. For example, in the image below, no issues were detected in **Heat map** on host 1 – this means that at least one critical issue was detected on each of these hosts during the selected time period (the last month in this example). The **Recent alerts** list shows that there are no relevant alerts among the most recent alerts about problems on the selected host (alert icons are without the bell sign).

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:

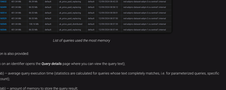
- In **Heat map**, hover the mouse over a host – a pop-up window will show alerts about the last two problems found on that host. Click **Open details** below an alert corresponding to an issue you want to receive more information about.



Summary of host issues

- In the **Recent alerts** list, click the icon in an alert row.

Each of these actions opens a separate page with a detailed description of the corresponding issue. For example, the details of an alert **Disk usage is high**, show which device is exceeding its usage limit. The **Threshold** field contains the metric threshold value set in Settings → Alerts, 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 (in other words, known or resolved).

Go to the Cluster metrics page

From **Heat map** on the **Dashboard** page, you can go to the Cluster metrics tab on the **Cluster metrics** page. It also shows the heat map matrix for an ADQM cluster and additionally displays the hosts' uptime and the number of active alerts for each host. In the left pane, you can select one or more hosts from the searched hosts or the left-hand tree, and then you can get more more information about a specific alert by clicking the corresponding row in the table. Further details will appear below the tree.

Heat map on the "Cluster metrics" page

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

- Click **Show all** in the pop-up window that appears when you hover the mouse over the selected host by clicking+ tool.
- Click **Full view** in the top right corner of **Heat map**.

A set of selected hosts and the **Acknowledged alerts** setting are synchronized between the **Dashboard** and **Cluster metrics** pages – changing these settings on one page automatically updates the same settings on the other page.

Top 10 tables

The **Top 10 tables** section contains two tabs with information about data tables in an ADQM cluster for a specified time interval:

- *Biggest* – 10 tables by the amount of data they store (sorted in descending order by **Size** – size of compressed data in a table at the end of the selected time interval);
- *Popular* – 10 tables to which the most queries were sent (sorted in descending order by **Request count** – number of queries sent to a table during the selected time interval).

List of the largest tables

List of tables received the most queries during the specified time interval

This section also provides the following information about tables:

- **Table name** – table name (click on a table name opens a page with detailed information on table columns);
- **Row count** (for **Biggest** tab) – number of rows over a table at the end of the selected time interval;
- **Last request** – time that any query to a table was run during the selected time interval;
- **Increase** (on the **Popular** tab) – change in a 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** – host where a table is located.

In the header of the **Table name** field, there is the icon, which when clicked allows you to view or change the filter that determines from which ADQM tables the largest or most popular 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 receive their metrics, that are sent every minute);
- *Historical* – top 10 tables are selected from all tables that existed in the cluster during the specified time interval (including tables that were deleted).

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 sent to ADQM cluster databases during a specified time interval:

- *Longest* – top 10 queries with the longest execution time (sorted in descending order by **Duration** – duration of a query execution);
- *Used memory* – top 10 queries required the most memory (sorted in descending order by **Used memory** – amount of memory consumed by a query).

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* (for **Longest** tab) – average query execution time (statistics are calculated for queries whose text completely matches, i.e. for parameterized queries, specific parameter values are taken into account);
- *Peak memory* (for **Used 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 icon, 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 predefined filters are available:

- *Completed* (on the **Longest** tab) – top 10 longest queries are selected from all the completed queries (i.e. from all queries except active ones);
- *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.

To view the full history of queries for the required period of time, switch to the Queries page.

Cluster metrics

Monitoring

ADQM CONTROL

Contents

To Table of Contents

Heat Map
- Heat 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 three tabs: 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 the information is to be 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 or the *Range* tab or set a custom time range in 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*: hosts marked in existing alerts about host issues

- Green are no alerts about issues with a host or all alerts are resolved;
- Light yellow issues have been detected in a host (for example, the increase in the values of some system metrics), but they are not yet critical. Appropriate warning alerts on these issues have been raised and remain relevant in ADQM Control;
- Dark yellow critical issues have been detected in a host. Appropriate alerts of the high importance level about it exists in ADQM Control.

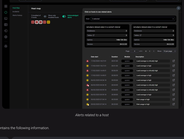
The *Show only troubled hosts* switch at the top of the *Heat map* tab allows you to configure the heat map so that it displays only troubled hosts.

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

Hovering a square on the heat map during there in the heat map displays a table with a list of host-related alerts generated during the specified period of time. Depending on the *Acknowledged alerts* setting, the list includes all alerts or only alerts requiring analysis – see Acknowledged alerts for details. In the *Host* field above the table, you can change a set of hosts for which a 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 listed in 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 `DD/MM/YYYY hh:mm:ss` format)
Duration	Total time that a host issue alert remained active
Module	Module that includes an alert. Currently, ADQM Control only supports these 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 *Severity* column of the table with a list of alerts contains an icon that indicates the severity level of an alert:

- warning – alert that informs about potential issues 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;
- high importance – alert that informs about critical issues when a system metric value exceeds a threshold set via the *Critical* parameter in the system.alerts configuration.

Severe issues and the last-stage indicate alerts about issues that are considered acknowledged (in other words, known/resolved). You can set this status on the alert details page. The icons

- are assigned to new and attention-requiring alerts that are not yet 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.

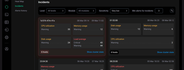
Alert details within the list

- Click the icon within an 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 alert as acknowledged or back to show it 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 timeline – 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 view* tick – refers to the Overview page with the Acknowledged alerts option enabled and the *Time* period equal to the incident interval. If an incident interval is less than an hour, ADQM Control automatically increases it to 1 hour by adding equal parts to each boundary (where 1 hour is the minimum allowed time interval that can be set in the *Time* field). On 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 too long or memory-consuming during this period, 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 defines the maximum allowed time interval, after which the next alert will be added to a new incident: $t_{max_diff_threshold} = Q3 + wins_coeff \cdot IQR$, where $wins_coeff$ is the winsorization coefficient and $Q3$ is the third quartile.

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-02, now-10`), the structure of incidents on the page may change, since the actual boundaries of the time interval shift over time, which means the set of alerts may change (new alerts may occur or the time of previously valid alerts may become outside the current time frame).

Settings

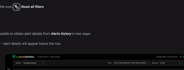
The *Incidents* tab contains the following parameters that affect which alerts are used to build incidents and how alerts are combined:

- *Level* – alert importance level. Possible values: `warning`, `critical`;
- *Module* – module to 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 period are combined into one incident:



Sensitivity is very low

- With the `Very high` sensitivity, alerts for the same time period are grouped into two 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 `DD/MM/YYYY 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 dash in the future.
- *Host* – host on which an issue was detected.

Hovering ... indicates alerts that 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 cluster hosts in the system. You can select multiple or all hosts at once.

To reset all applied filters, click the icon.

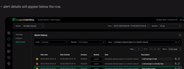
Alert details

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

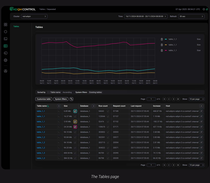
Work with tables in ADQM Control

← back to overview

Contents

To Table of Contents

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



The Tables page

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

- **Cluster** – ADQM cluster where data tables are located.
- **Time** – time period for which information about tables should be displayed. 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 (at least 1 hour) on the **Calendar** tab.
- **Refresh** – frequency of data updates.

ADQM Control supports the following ADQM Clickhouse table engines (i.e. can show tables based on these engines on the **Tables** page):

- engines of the MergeTree family (except versions of the AlterMergeTree engine and replicated table engines)
- engines of the Log family – Log and StripeLog
- the Memory, Merge, Join, Set, null engines.

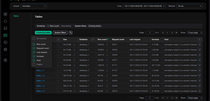
For integration table engines (used to access external data storage systems – for example, MySQL, MongoDB, HDFS, Hive, S3, Kafka, EmbeddedRocksDB, RabbitMQ, PostgreSQL), the correctness and completeness of the information displayed in ADQM Control is currently not guaranteed.

List of tables

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

Field	Description
Table name	Name of an ADQM table
Size	Size of compressed data in a table 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

Additional fields with information about ADQM tables

Field	Description
Row count	Number of data rows in a table at the end of the selected time interval
Request count	Number of queries sent to a table during the selected time interval
Last request	Time the last query to a table was run during the selected time interval
Increase	Change in a table size 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 Clickhouse web interface at http://<adqm_host_name>:8123/play, where <adqm_host_name> is the ADQM host FQDN (the system should be configured so that it can resolve the host FQDN into the IP address).
Engine	Table engine

Graphs

On the **Tables** page, you can get graphs to track the dynamics of changes in the **Size** and **Row count** metrics of ADQM tables (size of compressed data on disk and number of rows, respectively) over a selected period of time. Click the icon in the column of the corresponding metric – above the list of tables, ADQM Control will display a section with a graph built based on this metric of the ADQM table. You can add up to 8 graphs (inclusive) simultaneously.

To see values of metrics at some point in time, hover the mouse over the graph area.

Sample graphs by size of data in tables

To delete a graph, click the icon corresponding to the graph color in the table (for example, or in the image above) or the icon in the legend to the right of the graphs.

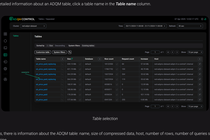
Filter and sort data

In the **Table name**, **Database**, **Host**, and **Engine** 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 its part, or full names of hosts). The icon means that a filter is defined for the column. To reset all filters by columns, click .

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

- **Show system tables** – display ADQM system tables from 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 period).
- **Only tables with graph** – 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 shows the sorting (**Sorted by**) and filtering (**System filters**, **Filtered by**) conditions 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 or the list of the most frequently queried tables in the ADQM cluster in the **Top 10 tables** section on the Dashboard page.

Table details

To open a separate page with detailed information about an ADQM table, click a table name in the **Table name** column.

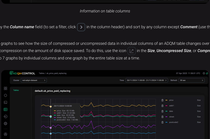
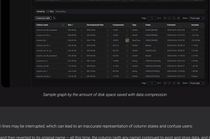


Table selection

At the top of the page that opens, there is information about the ADQM table: name, size of compressed data, host, number of rows, number of queries sent to the table, time of the last query, data size change, and engine.

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

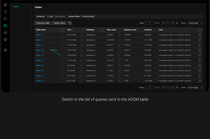
Field	Description
Column name	Column name at the end of the selected time interval
Size	Size of column data in compressed form
Uncompressed Size	Size of column data in uncompressed form
Compression	Percent of disk space saved by data compression
Type	Column data type
Codec	Data compression codecs applied 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 this 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, so 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 reverted to its 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 this case, 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** (all 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 ADQM Control

Anton Kolotovich

Contents

To Table of Contents

Fields of the Queries table

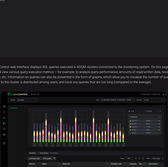
Sort and filter data

Query details

Graphs

- Duration
- User
- Kind

The **Queries** page in the ADCM Control web interface displays SQL queries executed in ClickHouse cluster(s) the monitoring system. On this page, you can get an overview of queries, monitor database activity, and view some query execution metrics. For example, to analyze query performance, amount of resources used, resource usage, errors if they have occurred during query execution, etc. The information on queries is presented in the form of graphs, which allow you to visualize the number of queries of different types, see how the total number of queries changes over time, to distribute among users, and have a quick view on the most long (compared to the average) queries.



The Queries page

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

- **Cluster** - ADQM cluster name to analyze queries.
- **Time** - time period when queries were launched. 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 **Custom** tab.
- **Refresh** - frequency of data updates.

Queries history is always displayed on the **Queries** page according to these filters. It contains all queries run on the selected ADCM cluster during the specified time period. To exclude DDL queries (`ON CLUSTER`), it also includes the queries executed on each host of the cluster and displays the `JV_WHiteFrog\query-log\d` entry (where the host name of each queries).

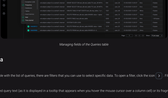
You can also get a list of the longest running queries or queries that used the most memory in the **Top 10 queries** section on the Overview page.

Fields of the Queries table

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

Field	Description
Query ID	Unique query identifier. The column also shows an icon indicating the query status: - is a query was executed by the `Query finished` status; - is a query is being executed at the current time (the `Query started` status); - is an incorrect before a query execution started (the `Exception before start` status); - is an error occurred during a query execution (the `Exception while processing` status); - is a query executed but not completed by some reason (the `Unexpected` status).
Query text	Query text. If a query is too long, it can be cut and put at end, you hover the mouse cursor over the column and, it is displayed as `Query start: Run Query end`. You can see the full text of a query on a separate page. To open it, click the query ID in the **Query ID** column.
Status	Current status of a query. Possible values: - `Query started` - a query execution is in progress; - `Query finished` - query was successfully executed; - `Exception before start` - an error occurred before a query execution started; - `Exception while processing` - an error occurred during a query execution; - `Unexpected` - a query execution was not completed for some reason, i.e. there is no information in the log about the successful completion or error of the query execution (for example, if a host was restarted during the query execution).
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 in the context of all filters currently applied on the page. Queries with the same query text (and completely matches it) are treated as parameterized queries, specific parameter values are taken into account.
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 ADQM user who initiated a query execution. For queries executed on cluster hosts when running a distributed query, a user is not defined – the **User** column shows an empty string (`''`).
Time started	Query execution start time.
Duration	Duration of a query execution on the current execution host (`ORDER BY time_started`). If a query that has not yet been completed is selected, the current time is used for the calculation. Until a query is replaced with the real query execution time since ADQM Control receives information about queries from ADQM the next time.
Average duration	Average query execution time. Statistics are calculated for queries whose text completely matches (i.e. for parameterized queries, specific parameter values are taken into account).
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 bytes read on all replicas. Until a query is replaced with the real query execution time, `Read bytes` value and the server initiator of the query are not received and local values.
Result bytes	Amount of memory used to store a query result.
Rows read	Number of rows in the result of a `SELECT` query or number of rows in an `INSERT` query.
Result 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 replicas – each replica sends its local rows. Until a query is replaced with the real query execution time, `Rows read` value and the server initiator of the query are not received and local values.
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`).
Type	Query type. Possible values: - `ALTER`, `Select`, `CREATE`, `INSERT`, `ALTER TABLE`, `DROP`, `UNKNOWN`, `DELETE`, `KILL MUTATION`, `RENAME`, `SYSTEM`, `OPTIMIZE`, `GRANT`, `REVOKE`, `SHOW(DATABASES)`, `SET`, `SYSTEM KILL QUERY`, `BEGIN`, `COMMIT`, `ROLLBACK`, `SET(TRANSACTION SNAPSHOT)`, `RELOAD(DICTIONARY)`, `RELOAD(FUNCTION)`, `EXTERNAL(DDL)`, `UPDATE`. - `Select` – for `SELECT`, `EXPLAIN`, `DESCRIBE`, `SHOW(TABLES)`, `SHOW(COLUMNS)`, `SHOW(DICT)`, `EXISTS` queries and queries that failed with the `SYNTAX_ERROR` or `INVALID_SYNTAX` errors.

To display 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

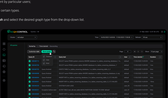
In the column headers of the table with the list of queries, there are filters that you can use to select specific data. To open a filter, click the icon. Filters are available in the following columns:

- **Query text** – enter a text/part text (as it is displayed in a tooltip that appears when you hover the mouse cursor over a column cell) to be frequent.
- **Databases**, **Tables** – enter a name of a database or table. In case with multiple databases or tables separated by commas in the **Databases** or **Tables** column, filtering works only if the search is performed by one item from a list of databases or tables in a cell.
- **Exception code** – enter a code of an error occurred during a query execution. The **Search** button is added to apply the filter.
- **Host name** – you can enter names of multiple hosts or users, respectively. For queries initiated by 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).
- **Status**, **Type** – you can select a value a query status or type from the drop-down list.

Through the button you can reset filters to default values. To reset all filters, click the button.

Data sorting is available in the following columns: **Exception** (virtually sorting is performed by error code), **Exception code**, **Used memory**, **Time started**, **Duration**, **Read bytes**, **Result bytes**, **Read rows**, **Result rows**, **Written rows**, and **Written bytes**. To sort data, click the column name. The sorting is performed on the server side.

Above the table with queries, there is a panel that shows the sorting (**Sorted by**) and filtering (**Filtered by**) conditions currently applied to data.



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 an error if it has occurred during the query processing. The page also displays query metrics in the columns selected on the **Queries** page via **Customize table**.

The Query details page

Graphs

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

- **Duration** – query execution time;
- **User** – number of queries sent by particular users;
- **Kind** – number of queries of certain types.

To build a graph, click **Show graph** and select the desired graph type 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 level of detail (number of points on the graph: `Low`, `High`).

Change the graph type and select the level of detail

Duration

The **Duration** 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 need to be analyzed).

Graph of queries by execution duration

This graph displays the following information on the execution time of queries sent to an ADCM cluster during a specified period of time:

- **95th Quantile** – 95th quantile of query execution time
- **Average** – average query execution time
- **Extreme rows** (one scheduled in red if the **Average** line is above the **95th Quantile** line) – time interval when there were one or more long queries relative to most others. For example, a line graph above such a case is observed in the time range from `12:00` to `12:30`. In this queries data, one can see that at `12:00:00` there was a query with a duration of `2s` (`2s`), while all other queries were completed much faster (the average in the table are sorted by the **Duration** column in descending order).

Calm rows (only between the **Average** and **95th Quantile** lines) – time interval when duration of all queries was comparable. A peak in this zone means that all queries in a given time interval have become longer to complete.

- **Query** – number of queries
- **No data** (was shaded in gray) – time interval for which there is no data about queries according to the applied filters. The **No data** item appears in the legend only if the graph contains one or more time intervals without information.

Example of a graph with no data on queries

Graph including "No data" time intervals

For the **Duration** graph, you can select the scale type: `Normal`, `Log` (the logarithmic scale can be more convenient for large time intervals – for example, a month or a week).

User

The **User** graph allows you to see the number of queries sent to the cluster by each user and evaluate the number relative to the number of queries from other users and to the total of all queries. The graph is built using data on queries from users whose names are listed in the filter by the **User** column. You can also specify user names in the fields to the right of the graph the list of selected users is displayed on the filter by the **User** column.

Graph showing the number of queries from different users

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 `SHARTOM` user for the selected time period.

Configuration parameters

Contents

ADSM Control
ADPS
Vectorizing

To hide or show the table of contents click on the button in the upper right corner.

This page contains the parameters that can be configured in ADSM Control and ADPS. To learn about the configuration process, refer to the Configuration guide. Refer to the glossary if you encounter any unfamiliar terms.

ADSM Control

ADSM Management

Parameter	Description	Default value
ADSM username	Username to connect to ADSM	adsm
ADSM user password	Password for the **ADSM USERNAME**, used to connect to ADSM Control. Encrypted using the **ADSM ENCRYPTION KEY** and stored in ADSM Control.	
ADSM listen port	Default port for ADSM Management on ADSM Control	9200
ADSM auth port	Default port for ADSM Authentication on ADSM Control	9201
Connection type	Type of connection between ADSM Control and ADSM Core. Possible values: **DIRECTLY**, **PROXY**, **PROXY_CERT_AUTHORITY**	Inherit from CTX

NOTE

Ports are also accessed through the ADSM Control URL, which is by default https://adsm.local/ui/. Changing the ADSM Control URL requires updating the ADSM Core configuration. Refer to the Reverse proxy guide for more information.

```
CREATE VIEW adsm TO CLUSTER default_cluster CONFIGURED WITH ( shards_amount = 1 replicas = 0 ) AS ( * where: );
```

NOTE

Ports are also accessed through the ADSM Core URL, which is by default https://adsm.local:9200/. Changing the ADSM Core URL requires updating the ADSM Control configuration.

Backup

Parameter	Description	Default value
Backup timeout	Global timeout for backup operations (minutes) (in seconds)	1440
Enable CBT validation	Enables CBT validation	Enabled

VM configuration

NOTE

VM details are stored in a Manager-wide database. For management of ADSM Control instances and clusters, see the Manager guide.

Parameter	Description	Default value
ADSM Control URL	Enables TLS for the Backup proxy	Disabled
ADSM Control URL fingerprint	Enables TLS for the Backup proxy	Disabled
ADSM Control URL trusted certs	Enables TLS for the Backup proxy	
ADSM Control URL client certificate	Enables TLS for the Backup Gateway connection	Disabled
ADSM Control URL client fingerprint	Enables TLS for the Backup Gateway connection	Disabled
ADSM Control URL client private key	Enables TLS for the Backup Gateway connection	
[Manager/agent] - Enable log file	Path to the file with the proxy log of the server. Set certificate for the Backup proxy. Enable/disable Backup proxy logging. Change logging level.	/var/log/adsm/backup.log
[Manager/agent] - Certificate file	Path to the server SSL certificate for the Backup proxy. Set certificate for the Backup proxy.	/var/lib/adsm/backup/cert.pem
[Manager/agent] - Private key file	Path to the server SSL private key for the Backup proxy. Set certificate for the Backup proxy.	/var/lib/adsm/backup/key.pem
[Manager/agent] - Certificate authority file	Path to the server SSL certificate for the Backup proxy. Set certificate for the Backup proxy.	/var/lib/adsm/backup/ca.pem
[Manager/agent] - Certificate authority file	Path to the file with the CA certificate for the Backup proxy. Set certificate for the Backup proxy.	/var/lib/adsm/backup/client-ca.pem

Network configuration

Parameter	Description	Default value
Backup connection timeout	Timeout of connections to Prometheus, database of the Backup proxy (seconds)	5000
Backup connection keepalive interval	Interval of keepalive connections to Prometheus, database of the Backup proxy (seconds)	5
Keepalive timeout	Timeout for queries and keepalives (in seconds)	120
Cleanup timeout	Period of timeout for detecting stale database queries. Stale queries are terminated. (in seconds)	86400
Migration timeout	Maximum connection timeout for migrations (seconds)	3600
Proxmoxbackup timeout	Proxmoxbackup timeout (in seconds)	60
Vmwarebackup timeout	Vmwarebackup connection timeout (in seconds)	60

Internal Prometheus settings

Parameter	Description	Default value
Prometheus scrape interval	Frequency of scraping information about the ADSM Core metrics (in seconds)	60
ADSMRB scrape interval	Frequency of scraping information about ADSM Backup Gateway (in seconds)	60
Metrics retention period	Retention period of Prometheus metrics (in days)	15
Query timeout interval	Frequency of querying Prometheus (in seconds)	120
Client query timeout	Frequency of querying agents (in seconds)	5
Client query log interval	Frequency of checking agents (in seconds)	60

Limits

Parameter	Description	Default value
Maximum public key-free operations per second	Maximum number of public key-free operations per second	50000
Limiter queue size	Maximum number of requests in the queue	1000
Maximum concurrent requests	Maximum number of requests	500
Counting retries limit	Number of attempts to get statistics or update data. If ADSM cannot get statistics or update data for the specified number of times, the corresponding VM will be marked as **STATISTICS UNKNOWN** or **STATISTICS FAILED**. Refer to the Backup guide for more information.	2
Tasks cancel limit	Size of the queue	32768
Tasks cancel invalidate interval	Time interval of the tasks cache invalidation (in seconds)	28800
Tasks cancel invalidate interval after failure	Time interval of the tasks cache invalidation after a failure (in seconds)	300
Quarantine cache size	Size of the quarantine cache	8192
Batch size of quarantine invalidations	Quarantine batch size for invalidations	1024
Capacity of backup storage (megabytes)	Capacity of backup storage	9223372036854
Counting agents queue multiplier (percent)	Percent of the counting tasks queue for counting agents	64000
Tasks counting queue for counting tasks (percent)	Percent of a tasks counters in accounts for the counting tasks queue	1779833
Maximum VM number in cluster	Maximum number of cluster accounts	5

Internal PostgreSQL server

NOTE

To change the configuration parameters, connect to the **internal PostgreSQL** server using psql.

Parameter	Description	Default value
server_ip_port	**internal PostgreSQL** server IP address (for example, 127.0.0.1:5432)	
dbms_dbname	Database name for the core database (**ADSM**)	adsm
dbms_dbuser	Database name for the core database (**ADSM**)	adsm
DbFreeMb (percent)	Minimum free space on the disk for the **ADSM** database	10
DbFreeMb (percent)	Minimum free space on the disk for the **ADSM** database. Set to zero to disable. Minimum value: 1 GB. Maximum value: 50 GB. Default value: 10 GB.	10
SSL is used by an external server	Specifies whether SSL is used by an external server	Disabled

ADPS

ADPS configuration

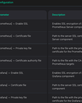
NOTE

To change the ADPS parameters, connect to the **internal PostgreSQL** server using psql.

Parameter	Description	Default value
Data directory	Data directory path for the **internal PostgreSQL** server	/var/lib/adsm/postgresql/data
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 buffers	128MB
max_worker_processes	Maximum number of background processes that the system can support	32
max_parallel_workers	Maximum number of parallel threads that can be supported by the parallel query subsystem	32
max_parallel_workers_per_gather	Maximum number of parallel threads per one gathering or single-parent of parallel query	16
max_parallel_maintenance_workers	Maximum number of parallel workers that can be started by a single utility command	2
effective_cache_size	Sets the ADPS query planner's assumption about the effective size of the disk cache available for a single query. This is a free-form hint, neither of allocated nor shared memory is actually reserved. Set to 75% of the total RAM. Minimum value: 128 MB. Maximum value: 512 GB. Default value: 75% of the total RAM.	4294967295
maintenance_work_mem	Memory limit for maintenance operations. Increasing this value may improve performance of vacuuming and restoring database dumps. Minimum value: 1 MB. Maximum value: 2 GB. Default value: 1 GB.	1073741824
work_mem	Base (dequeued) memory limit used for internal operations when the sort operations, bitmap operations, hashes and materialized views require temporary disk files. Increasing this value may improve performance for sorting and hashing. Set to 5% of the total RAM. Minimum value: 1 MB. Maximum value: 512 MB. Default value: 5% of the total RAM.	5242148
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. Minimum value: 32 MB. Default value: 1 GB.	1073148
max_wal_size	Maximum WAL size that triggers a checkpoint. These are usually checkpoints spaced more than **CHECKPOINT_TIMEOUT** seconds apart. If this value is reached, a checkpoint is triggered even if checkpoints would be spaced closer than **CHECKPOINT_TIMEOUT** seconds apart. Minimum value: 32 MB. Default value: 4 GB.	4096 MB
wal_keep_size	Specifies the minimum size of recent WAL files to keep, in megabytes, to be available for fetching by standby servers. Minimum value: 0 (disabled). Default value: 1 GB.	1048 MB
Page weight	Defines the page weight. Possible values: 1 (very low), 2 (low), 3 (normal), 4 (high), 5 (very high). Default value: 3 (normal).	3
	1 - 1 in-flight page request fails, the worst delay not seen	
	2 - 2 in-flight page requests fail, the worst delay not seen	
	3 - 3 in-flight page requests fail	
	4 - 4 in-flight page requests fail	
	5 - 5 in-flight page requests fail	
expires_interval_seconds	Number of connection "slots" that can be reserved for PostgreSQL superuser maintenance. Default value: 3	3
limited_work_mem_backgroud_workers	Maximum number of background worker processes allocated to PostgreSQL	10
vacuum_cost_limit	Amount of spent memory units, by which the cost limit is reached. Default value: 200	10485
default_statistics_target	Sets the default statistics target for columns specified by ALTER TABLE SET STATISTICS. Default value: 100	100
vacuum_cost_page_hit	Estimated cost for vacuuming a buffer found in the shared buffer cache. Default value: 1	1
checkpoint_completion_target	Target time of a checkpoint in a fraction of total time between checkpoints. Default value: 0.9	0.9
log_lock_waits_per_transaction	If this parameter is enabled, then every **LOG_LOCK_WAIT_STRUCTURE** **LOCK_WAIT_TIMEOUT** milliseconds, the server logs information about the lock that the transaction is waiting for, and the locks held by other transactions that block the transaction from acquiring the lock. This information is useful for troubleshooting lock contention issues. Default value: 1 (enabled). Minimum value: 0 (disabled). Maximum value: 1 (enabled). Set to 1 for production systems. Set to 0 only for debugging lock issues. Enabling this parameter adds overhead to lock operations. Disabling this parameter reduces lock contention diagnostic capabilities. Default value: 1 (enabled). Minimum value: 0 (disabled). Maximum value: 1 (enabled). Set to 1 for production systems. Set to 0 only for debugging lock issues. Enabling this parameter adds overhead to lock operations. Disabling this parameter reduces lock contention diagnostic capabilities.	1
log_lock_wait	Controls the logging of lock waits. If enabled, logs a message when a lock wait exceeds **LOG_LOCK_WAIT_STRUCTURE** milliseconds. Default value: 1 (enabled). Minimum value: 0 (disabled). Maximum value: 1 (enabled). Set to 1 for debugging lock issues. Enabling this parameter adds overhead to lock operations. Disabling this parameter reduces lock contention diagnostic capabilities.	1
log_lock_wait_structure	Controls the logging of lock wait structures. Default value: 1 (enabled). Minimum value: 0 (disabled). Maximum value: 1 (enabled). Set to 1 for debugging lock issues. Enabling this parameter adds overhead to lock operations. Disabling this parameter reduces lock contention diagnostic capabilities.	1
log_min_duration_statement	Specifies minimum execution time in milliseconds for a statement to be logged. Default value: **DEFAULT**. Minimum value: **DEFAULT**. Maximum value: **DEFAULT**. Set to **DEFAULT** for production systems. Set to 0 for debugging. Set to -1 to disable. Setting to 0 may significantly increase log volume. Setting to **DEFAULT** may miss slow queries. Logging adds overhead. Default value: **DEFAULT**. Minimum value: **DEFAULT**. Maximum value: **DEFAULT**. Set to **DEFAULT** for production systems. Set to 0 for debugging. Setting to 0 may significantly increase log volume. Setting to **DEFAULT** may miss slow queries. Logging adds overhead.	1000
log_min_error_statement	Controls which SQL statements that cause an error condition are recorded in the server log. Default value: **ERROR**. Minimum value: **DEBUG5**. Maximum value: **PANIC**. Set to **ERROR** for production systems. Set to **DEBUG5** for debugging. Setting to **DEBUG5** may significantly increase log volume. Setting to **PANIC** may miss critical errors. Default value: **ERROR**. Minimum value: **DEBUG5**. Maximum value: **PANIC**. Set to **ERROR** for production systems. Set to **DEBUG5** for debugging. Setting to **DEBUG5** may significantly increase log volume. Setting to **PANIC** may miss critical errors.	error
log_min_messages	Controls which messages are written to the server log. Default value: **WARNING**. Minimum value: **DEBUG5**. Maximum value: **PANIC**. Set to **WARNING** for production systems. Set to **DEBUG5** for debugging. Setting to **DEBUG5** may significantly increase log volume. Setting to **PANIC** may miss critical messages. Default value: **WARNING**. Minimum value: **DEBUG5**. Maximum value: **PANIC**. Set to **WARNING** for production systems. Set to **DEBUG5** for debugging. Setting to **DEBUG5** may significantly increase log volume. Setting to **PANIC** may miss critical messages.	warning
log_min_duration_statement	Specifies minimum execution time in milliseconds for a statement to be logged. Default value: **DEFAULT**. Minimum value: **DEFAULT**. Maximum value: **DEFAULT**. Set to **DEFAULT** for production systems. Set to 0 for debugging. Set to -1 to disable. Setting to 0 may significantly increase log volume. Setting to **DEFAULT** may miss slow queries. Logging adds overhead. Default value: **DEFAULT**. Minimum value: **DEFAULT**. Maximum value: **DEFAULT**. Set to **DEFAULT** for production systems. Set to 0 for debugging. Setting to 0 may significantly increase log volume. Setting to **DEFAULT** may miss slow queries. Logging adds overhead.	4410

Custom postgresql.conf

If the **Custom postgresql.conf** setting is set, all configuration parameters are written to /etc/pg-ctl.conf. The path cannot be edited. **Edit** opens it.



Custom postgresql.conf section

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

Window to add parameter

You can add multiple parameters.

Shared postgresql.libraries

Specifies the name(s) of shared libraries that are to be preloaded at connection time. Default value depends on **TIMESCALEDB**, and **TIMESCALEDB_TOOLKIT** settings. Set to empty to disable. Refer to the PostgreSQL documentation.

The "Shared postgresql.libraries" section

Custom pg_hba.conf

The field allows you to add entries to the /var/lib/pgsql/data/pg_hba.conf file. Refer to the PostgreSQL documentation.

VM configuration

NOTE

Any VM parameters can be changed using the API or ADSM Control, using the Edit button.

Parameter	Description	Default value
Enable SSL	Enables SSL decryption of incoming traffic for the ADPS service	Disabled
Certificate file	Path to the server SSL certificate for the ADPS service	/var/lib/adsm/cert.pem
Private key file	Path to the file with the proxy log of the server SSL certificate	/var/lib/adsm/key.pem
Certificate authority file	Path to the CA certificate file	/var/lib/ca-trust/ca.pem

Monitoring

Prometheus settings

Parameter	Description	Default value
Scrape interval	How often Prometheus scrapes its target targets	15s
Evaluation interval	Specifies how long to wait until a rule or alert is considered stable	15s
Password for prometheus user	Password of a Grafana user in connect to Prometheus	
Allowed IP/IP range for connection to Prometheus	Sets a whitelist for logging into the Prometheus Web interface	
Prometheus port	Prometheus port	9090
Service parameters	Prometheus parameters	

Grafana settings

Parameter	Description	Default value
Grafana administrator password	Password of a Grafana administrator user	
Grafana listen port	Port to access the Grafana web interface	3000

Node Exporter settings

Parameter	Description	Default value
Listen port	Port to listen for scrapes. Listen metrics in the Prometheus format	9100
Metrics path	Endpoint to serve system metrics	/metrics

VM configuration

Parameter	Description	Default value
[Prometheus] - Enable SSL	Enables SSL decryption of incoming traffic for the Prometheus Server component	Disabled
[Prometheus] - Certificate file	Path to the server SSL certificate for the Prometheus Server component	/etc/prometheus/prometheus.crt
[Prometheus] - Certificate authority file	Path to the CA certificate file for the Prometheus Server component	/etc/prometheus/prometheus.crt
[Grafana] - Enable SSL	Enables SSL decryption of incoming traffic for the Grafana component	Disabled
[Grafana] - Certificate file	Path to the server SSL certificate for the Grafana component	/var/lib/grafana/cert.pem
[Grafana] - Private key file	Path to the file with the proxy log of the server SSL certificate for the Grafana component	/var/lib/grafana/grafana.key
[Grafana] - Certificate authority file	Path to the file with the CA certificate for ADPS	/var/lib/grafana/ca.crt
[Node exporter] - Enable SSL	Enables SSL decryption of incoming traffic for the Node Exporter component	Disabled
[Node exporter] - Certificate file	Path to the server SSL certificate for the Node Exporter component	/etc/node-exp.crt
[Node exporter] - Private key file	Path to the file with the proxy log of the server SSL certificate for the Node Exporter component	/etc/node-exp.key

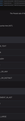
Quark exception codes

Contents

Syntax Processing
Semantics
Runtime Operations
Resource Governing
Metadata Operations
Data Types
Data Operations
Data Management
Scripting Functions
Scripting Operations
Transaction Control
Transaction Operations
Triggers
Scripting Classes

To Table of Contents

Quark is a data-centric scripting language and runtime environment for building, testing, and deploying business logic and data services. Quark exposes structured exceptions to the runtime environment via numeric error codes. Each exception code is associated with a specific subsystem, such as syntax processing, semantics, runtime operations, or resource governing, and may be accompanied by additional context such as the offending statement, expected versus actual values, or a stack trace. Exceptions can be caught and handled in Quark scripts using `try` / `catch` blocks, or they can propagate to the runtime environment where they are logged and reported to the operator console.


Quark Console
Connected to localhost:7000
[2026-04-12 09:14:03] [INFO] Quark runtime v1.4.2 ready
[2026-04-12 09:14:03] [INFO] Listening on port 7000
[2026-04-12 09:14:03] [INFO] Loaded 12 modules, 0 errors
[2026-04-12 09:14:03] [INFO] Loaded 3 triggers
[2026-04-12 09:14:03] [INFO] Loaded 2 scripts
[2026-04-12 09:14:03] [INFO] Loaded 1 schedule
[2026-04-12 09:14:03] [INFO] Loaded 4 data sources
[2026-04-12 09:14:03] [INFO] Loaded 2 transaction handlers
[2026-04-12 09:14:03] [INFO] Loaded 1 REST endpoint
[2026-04-12 09:14:03] [INFO] Loaded 1 WebSocket endpoint
[2026-04-12 09:14:03] [INFO] Loaded 1 scheduled job
[2026-04-12 09:14:03] [INFO] Loaded 1 file watcher
[2026-04-12 09:14:03] [INFO] Loaded 1 mail handler
[2026-04-12 09:14:03] [INFO] Loaded 1 queue consumer
[2026-04-12 09:14:03] [INFO] Loaded 1 JMX bean
[2026-04-12 09:14:03] [INFO] Loaded 1 SNMP agent
[2026-04-12 09:14:03] [INFO] Loaded 1 LDAP connector
[2026-04-12 09:14:03] [INFO] Loaded 1 Kafka consumer
[2026-04-12 09:14:03] [INFO] Loaded 1 Kafka producer
[2026-04-12 09:14:03] [INFO] Loaded 1 Redis connector
[2026-04-12 09:14:03] [INFO] Loaded 1 Elasticsearch connector
[2026-04-12 09:14:03] [INFO] Loaded 1 Slack webhook
[2026-04-12 09:14:03] [INFO] Loaded 1 Twilio SMS connector
[2026-04-12 09:14:03] [INFO] Loaded 1 Stripe webhook
[2026-04-12 09:14:03] [INFO] Loaded 1 GitHub webhook
[2026-04-12 09:14:03] [INFO] Loaded 1 Jira connector
[2026-04-12 09:14:03] [INFO] Loaded 1 Slack connector
[2026-04-12 09:14:03] [INFO] Loaded 1 Twilio connector
[2026-04-12 09:14:03] [INFO] Loaded 1 Stripe connector
[2026-04-12 09:14:03] [INFO] Loaded 1 GitHub connector

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>:71203/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_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

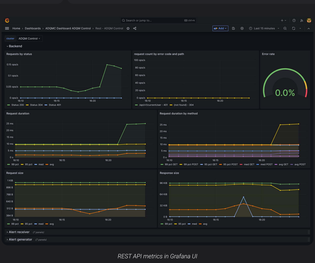
Alerts

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

1.2.0
1.1.1
1.1.0
1.0.0

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

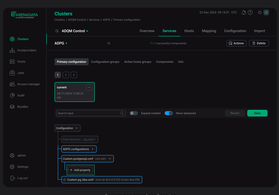
TIP

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 the disk are used. This limit is also a simple resource control, preventing the host from going into swapping due to overallocation. Since the allocated 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

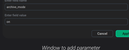
Custom postgresql.conf

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**

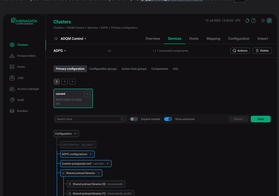


Window to add parameter

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 < 1 >.



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 ADGM 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	500000000
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	77760000
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