DCAP CENTRAL

jSONAR Simplifying Security

tech note

jSonar Integration with MongoDB Atlas

Yael Kadoshi, jSonar

tech note

CONTENTS

tech note

1 OVERVIEW

Auditing allows administrators to track system activity for deployments with multiple users. Atlas administrators can select the actions that they want to audit, as well as the MongoDB users, Atlas roles, and LDAP groups whose actions they want audited.

Atlas Auditing is enabled on a Project level. The Atlas account that is used to configure the Auditing should have permissions of "Project Owner" role.

NOTE: Currently Authentication Failed (result 18) are not in Atlas log files

Some of the audit option are limited or not available on some of the cluster tiers.

For more information see: https://docs.atlas.mongodb.com/database-auditing/

2 ENABLE AUDIT LOGGING ON MONGODB ATLAS

Login to Mongo Atlas Account to https://cloud.mongodb.com

Auditing is configured on a Project level – this should be done for any project that contains a database that should be audited.

2.1 ENABLE API WHITELISTING FOR YOUR ORGANIZATION

It is recommended that you configure Atlas to require API whitelisting at the organization level. When this setting is enabled, all API calls within that organization must originate from an entry on each respective Atlas user's API whitelist.

To perform any of the following actions, you must have the Organization Owner role.

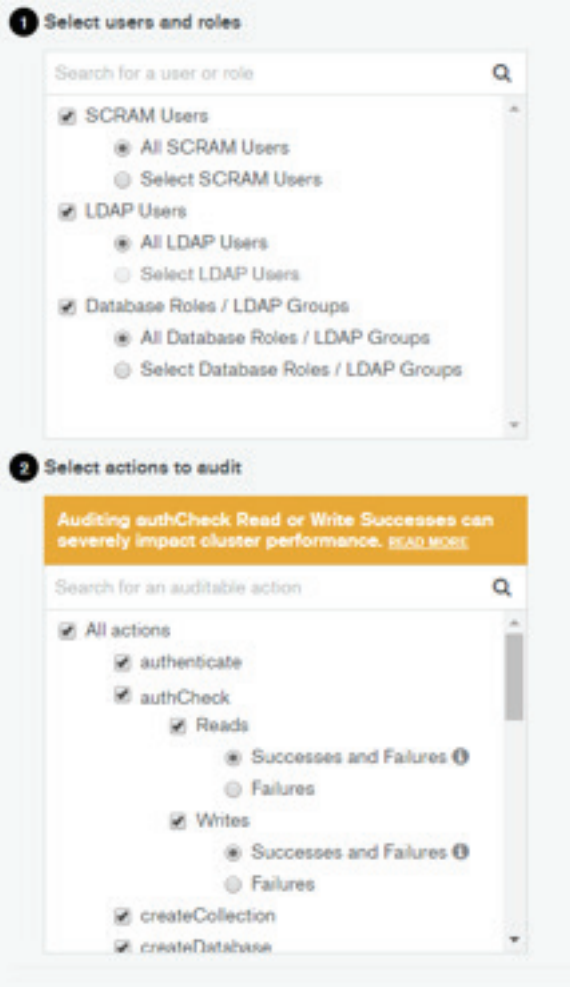
Access the Organization using the **Context** picker in the top-left hand corner of the Atlas UI.
Click **Settings** from the left-hand navigation. Toggle the **Require IP Whitelist for Public API** setting to On.

For more information see:
https://docs.atlas.mongodb.com/configure-api-access/#application-api-keys

2.2 TURN ON DATABASE AUDITING FEATURE

Select the Project that contains the database to audit. Go to "Clusters", click on "Security" tab, then on "Enterprise Security" tab. Turn "On" Database Auditing

Click on "Audit Filter Settings". Select all users and roles options for auditing. Select all actions for auditing. Make sure to enable Audit on Read/Write Success events.



2.3 USING CUSTOM JSON FILTER TO FILTER OUT MMS EVENTS

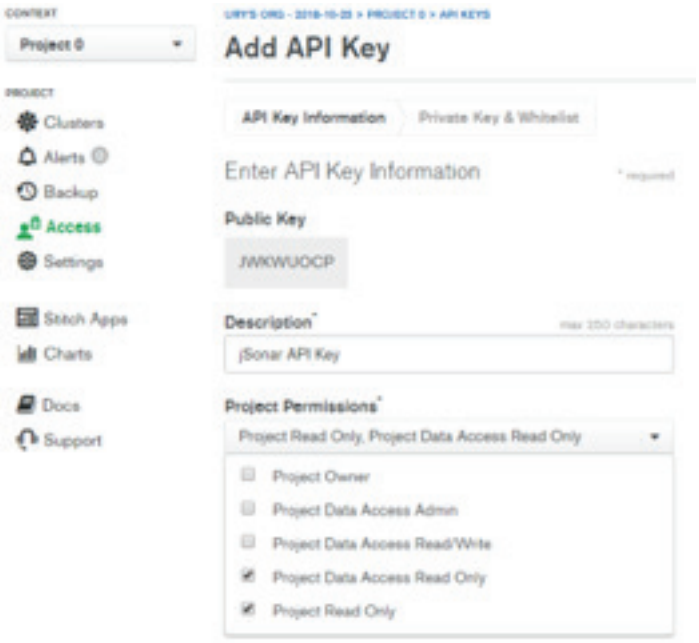
On "Database Auditing" page, click on "USE CUSTOM JSON FILE". Add the following clause to the $and condition –

```
{
 "$or": [
  {
   "users.user": {
    "$nin": [
     "mms-monitoring-agent",
     "mms-backup-agent"
    ]
   }
  },
  {
   "atype": {
    "$nin": [
     "authenticate",
     "authCheck"
    ]
   }
  }
 ]
},
```

2.4 GENERATE APPLICATION API KEY FOR A PROJECT

From the Context menu, select the project that you want the jSonar API to access. Click Access. Click the tab for API Keys. Select Create API Key from the Manage button menu. From the API Key Information step of the Add API Key page, enter a description for the new API Key in the Description box. Check the permissions "Project Data Access Read Only" (to read audit logs) and "Project Read Only" (to read events).

Copy the Public Key – this will serve as the username when configuring jSonar access. Click Next.



Copy the Private Key now - store in a secure location. It will not be displayed again after you leave this page.

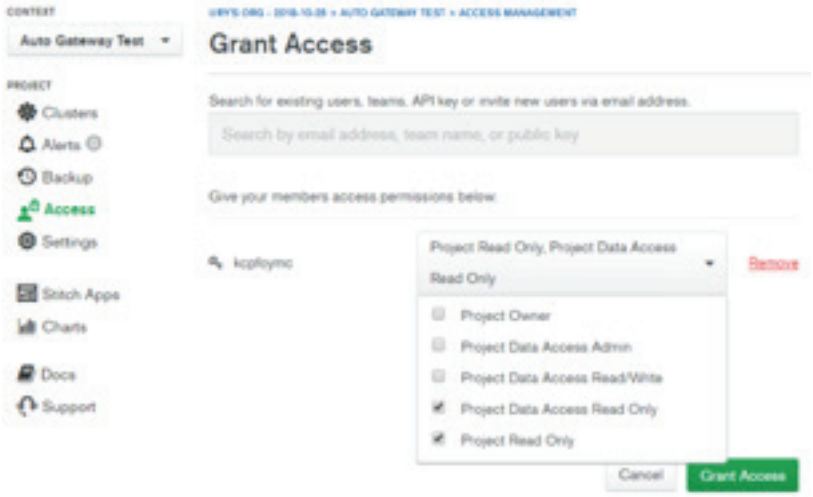
Click Add Whitelist Entry. Enter the Public IP address of the jSonar server where the audit logs will be ingested (make sure it is static IP). Click Save.

Note: In case the IP will be changed, you should add the new IP to the whitelist.

tech note

2.5 GRANT ACCESS TO ADDITIONAL PROJECT FOR THE API KEY

From the **Context** menu, select any other project that you want the jSonar API to access. Click **Access.** Click the tab for **API Keys**. Search and select the API Key you just created for jSonar access. Select **Grant Access** from the **Manage** button menu. Check the permissions "Project Data Access Read Only" (to read audit logs) and "Project Read Only" (to read events). Click **Grant Access.**



3 JSONAR CONFIGURATION

From the **Context** menu, select any other project that you want the jSonar API to access. Click **Access.** Click the tab for **API Keys**. Search and select the API Key you just created for jSonar access. Select **Grant Access** from the **Manage** button menu. Check the permissions "Project Data Access Read Only" (to read audit logs) and "Project Read Only" (to read events). Click **Grant Access.**

3.1 CONFIGURE SONARGATEWAY SERVICE

Make sure that the atlas gateway service is enabled:
systemctl enable gateway-mongodb@atlas.service

3.2 ADD CLOUD SOURCE

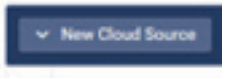
Login to jSonar GUI as an administrator. In the homepage under "Administration" section, click on "Cloud Sources". Check if the service is running by looking at "Mongodb Atlas" bar on the top, if it's green then the service is alive. You can also check the status of the atlas gateway service in the shell using command:

systemctl status gateway-mongodb@atlas.service -l

tech note

tech note

Click on "New Cloud Source" on the top left.



Select from the dropdown "MongoDB Atlas".

Enter the username/Public Key and Private API key generated from MongoDB Atlas. Click "Add MongoDB Atlas".

A new cloud source should be added to the list. On the right side click on the enable button and watch the status icon, it might take a few seconds to start importing logs and by then the status icon will change to green, you can always click on the refresh to check the status icon.

4 FILE LOCATIONS

Log file: /var/log/sonar/gateway/cloud/mongodb/atlas/sonargateway.log

Logging conf file: /etc/sonar/gateway/cloud/gateway-mongodb-logging.conf

Mapping file: /etc/sonar/gateway/cloud/mongodb/atlas.json