



RDFOX[®]
Mantix

RDFOX[®] Mantix operational manual

Guide for administrators

Table of Contents

Introduction	2
Organization of Data	2
Network Topology	2
Deploying RDFox® Mantix	3
Desktop Deployment Demo	3
Server Deployment Quick-Start Guide	3
Set up Service Role	4
Environment Variables	4
Run RDFox® Mantix	6
Reference	8
Authentication	8
Password authentication	8
OIDC authentication	9
Chat for queries	9
Environment Variables	10
Backup and Restore	13
Appendix: Troubleshooting	14
Error Messages in Terminal	14
Environment Variables	15
Full Page Error Messages	16
Tuning for Large Datasets	16
Appendix: Supported AI Providers	17
AI Providers	17

Introduction

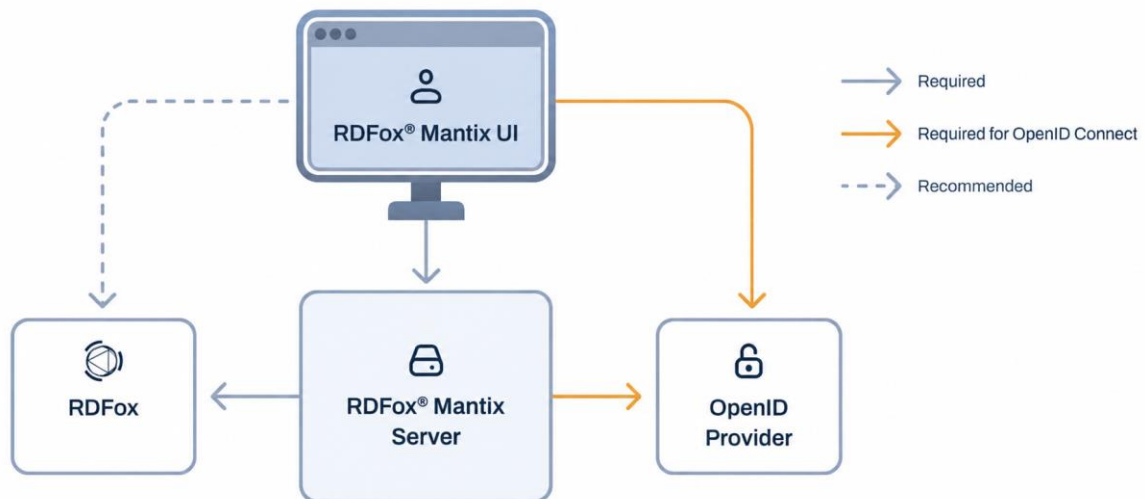
This guide contains information for operators deploying RDFox® Mantix, a new browser-based application for developing rules to use with RDFox®.

RDFox® Mantix requires access to an appropriately configured instance of RDFox®, pre-loaded with the knowledge graph for which rules are to be written. It uses this to query the target data, and to save its own settings and content. The later sections of this guide describe how to configure RDFox® for use with RDFox® Mantix, and how to run RDFox® Mantix itself.

Organization of Data

Data in RDFox® Mantix is organized into workspaces, each mapped one-to-one with your data stores, referred to as "target datastores" in RDFox® Mantix. These workspaces contain rules, terms, ontology views, and other RDFox® Mantix-specific content. Workspace data is stored in special system datastores prefixed with "workspace_" to prevent pollution of your data stores with RDFox® Mantix data. These workspace datastores are not intended for direct access or use except through RDFox® Mantix.

Network Topology



The network topology for RDFox® Mantix is shown in the following diagram. Although RDFox® Mantix does not require an internet connection, as it does not make requests outside of your network, it will require access to your identity provider if you choose to authenticate with OpenID Connect.

Deploying RDFox® Mantix

This section of the guide contains two sets of instructions for deploying RDFox® Mantix: a Docker Compose project suitable for quickly launching RDFox® Mantix and RDFox® together on a single developer's machine, and a server deployment quick start guide.

Desktop Deployment Demo

To get an example instance of RDFox® Mantix up and running quickly, we have a simple Docker Compose project available. This can be useful for understanding how to run the RDFox® Mantix Docker image.

Pre-requisites

- A development machine with docker-compose.

Once you have downloaded the demo, unzip the files and follow the instructions in the README.md.

Server Deployment Quick-Start Guide

This section of the guide contains steps for launching RDFox® Mantix, in a basic configuration, on a server.

Pre-requisites

- A running RDFox® instance that is v7.7 or higher, and that is set up for both rolename and password authentication, See the RDFox® **Getting Started** guide.
- Full privileges for the RDFox® instance,
- A host with Docker where RDFox® Mantix will run.

The users of RDFox® Mantix will need to be able to connect to the RDFox® Mantix server from their browsers. Advanced users will also benefit from being able to connect to RDFox® from their browsers.

Once you have these items, begin with the following steps.

STEP 1

Set up Service Role

You will need to [set up a role in RDFox](#) for RDFox® Mantix and grant it the permissions to do everything it needs to do in RDFox®.

In RDFox®, create the `mantix-service-role` role, note the password you give it for use later and add the following privileges to it in the [RDFox® shell](#):

```
> role create mantix-service-role

(RDFox® will prompt you for the role's password at this point)

> grant privileges read | to mantix-service-role
> grant privileges read,write >datastores to mantix-service-role
> grant privileges read |roles|* to mantix-service-role
```

STEP 2

Environment Variables

All setups will need the following required variables configured:

Environment Variable	Accepted Values	Description
<code>RDFOX_MANTIX_RDFOX_URL</code>	URL	The URL of the RDFox® instance. To suppress TLS config warnings for non https URLs, set <code>RDFOX_MANTIX_RDFOX_URL_ALLOW_INSECURE_CONNECTION=true</code>
<code>RDFOX_MANTIX_AUTH_SECRET</code>	String	A secret used to encrypt session tokens. Generate a secret using <code>openssl rand -base64 33</code> or equivalent cryptographic function.

RDFOX_MANTIX_AUTH_URL	URL	The canonical URL of the RDFox® Mantix instance. This is used for authentication .
RDFOX_MANTIX_SERVICE_ROLE	String	The role name you are using for the service role. For the quick start guide, it is <code>mantix-service-role</code>
RDFOX_MANTIX_SERVICE_PASSWORD	String	The password you gave to the <code>mantix-service-role</code> role

RDFox® using role manager

If you are not using OIDC with RDFox®, you must use the standard role manager provided by RDFox.

To create a user with Editor access (see the section on Authentication below), start by creating the `mantix-editor` role:

```
> role create mantix-editor no-password
```

Next create a role for each end user in RDFox®, allowing them to set their own strong and unique password. For example, the following command creates a user called `my-user`.

```
> role create my-user
```

(RDFox® will prompt you for the role's password at this point)

Finally, make each end user role a member of one of the predefined roles to grant them access to RDFox® Mantix. For example, the following command makes `my-user` a `mantix-editor`:

```
> grant role mantix-editor to my-user
```

Your users can then log into RDFox® Mantix with their RDFox® role name and password.

If provided, the `RDFOX_MANTIX__AUTH_USE_RDFOX_ROLE_MANAGER` environment variable should be set to `true`.

RDFox® using OIDC Authentication

When using RDFox® Mantix with OIDC, you will need to create the `mantix-editor` role within your OIDC provider and make each user who will access RDFox® Mantix a member of this role.

Your users can then log into RDFox® Mantix using OIDC.

The following variables are required when using OIDC:

Environment Variable	Value
<code>RDFOX_MANTIX_AUTH_OIDC_ISSUER</code> <code>RDFOX_MANTIX_AUTH_OIDC_USERINFO</code> <code>RDFOX_MANTIX_AUTH_OIDC_TOKEN</code>	This should be set to the issuer URL, and 'userinfo' and 'token' endpoints, of your OIDC issuer
<code>RDFOX_MANTIX__AUTH_OIDC_CLIENT_ID</code> <code>RDFOX_MANTIX__AUTH_OIDC_CLIENT_SECRET</code>	These should be set to the client ID and secret of RDFox® Mantix in your OIDC setup

For more information on authentication, see [Authentication](#) bookmark://_Authentication below

STEP 3

Run RDFox® Mantix

To start a container with the RDFox® Mantix Docker image, first ensure you have a running instance of RDFox® that will be visible to the container and that you have values for the relevant environment variables.

The container can be created and started using the standard docker [run](#) command. Ensure that you have the correct environment variables.

Here is an example run command. This assumes that you have a running RDFox® instance at <https://rdfox.acme.com> and that VERSION has been replaced by the correct image tag.

```
docker run -d \  
  --name mantix \  
  
  --env RDX_MANTIX RDX_URL=https://rdfox.acme.com \  
  
  --env RDX_MANTIX_AUTH_URL=https://mantix.acme.com \  
  
  --env RDX_MANTIX_AUTH_SECRET=generated-secret-value \  
  
  --env RDX_MANTIX_SERVICE_ROLE=mantix-service-role \  
  
  --env RDX_MANTIX_SERVICE_PASSWORD=chosen-password \  
  
  --volume ~/.RDX/opt/RDX/RDX.lic\  
  
  -p 3000:3000 \  
  mantix:VERSION
```

Please refer to the Desktop Deployment Demo compose file for an example of how to start the RDX® Mantix container.

Reference

Authentication

RDFox® Mantix supports two forms of authentication, RDFox® role name and password authentication, and OIDC using your chosen OIDC provider.

Warning!

All RDFox® Mantix roles grant read access to all datastores and their contents, including named graphs and tuple tables. RDFox® Mantix does not support fine grained access control of RDFox® resources.

RDFox® Mantix supports three different privilege levels for users, editors, rule-editors, and viewers.

Editors have full access to RDFox® Mantix. They can create, edit, and delete rules and views, can modify the ontology, and can manage workspaces.

Rule-editors have full control over rules. They can create, edit, and delete rules, as well as look at the ontology and views, but they can't modify the ontology or views, and have no control over workspaces.

Finally, viewers can look at rules, views, and the ontology, but can't change anything (or manage workspaces).

Password authentication

RDFox® password authentication is enabled by default.

To configure this, you will need to create the roles `mantix-editor`, `mantixrule-editor`, and `mantix-viewer` within RDFox®.

To make a user an editor in RDFox® Mantix, their RDFox® role will need to be a member of the `mantix-editor` role, and similar for rule-editors and viewers.

The user can then just log in to RDFox® Mantix with their RDFox® role name and password.

To disable role name and password authentication, set the environment variable `RDFOX_MANTIX__AUTH_USE_RDFOX_ROLE_MANAGER=false`.

OIDC authentication



By default, SSO authentication is disabled. RDFox® Mantix will need its own app registration.

To enable OIDC authentication, the following environment variables will need to be set:

- `RDFOX_MANTIX_AUTH_OIDC_ISSUER`
- `RDFOX_MANTIX_AUTH_OIDC_CLIENT_ID`
- `RDFOX_MANTIX_AUTH_OIDC_CLIENT_SECRET`

To make a user an editor in RDFox® Mantix, they will need to have the `mantix-editor` role in your OIDC provider, and your OIDC provider will need to include this in roles claim in JWTs it issues to RDFox® Mantix for that user.

Similarly, to make a user a rule-editor, they will need to have the `mantix-rule-editor` role, and to make them a viewer, they will need the `mantix-viewer` role.



Chat for queries

To enable the chat feature, which generates SPARQL queries via natural language for your data, you must provide a valid configuration for your chosen AI provider. The chat configuration consists of two environment variables listed below. If neither variable is provided, the chat feature will be disabled. If an invalid configuration is provided, the RDFox® Mantix Docker container will fail to start and will return a warning.

RDFox® Mantix AI_Providers

This variable is a JSON string. The JSON should contain any provider specific config as a JSON object, keyed by the provider's configuration key specified in [Appendix: Supported AI Providers](#). Multiple providers may be specified, however only a single provider specified in `RDFOX_MANTIX_LANGUAGE_MODEL_ID` will be used at any one time. Some examples are provided below.

```
RDFOX_MANTIX_AI_PROVIDERS='{ "openai": { "apiKey": "my-api-key" } }'
```

```
RDFOX_MANTIX_AI_PROVIDERS='{ "huggingFace": { "apiKey": "my-api-key",  
"base-url": "https://router.huggingface.co/v1" } }'
```

```
RDFOX_MANTIX_AI_PROVIDERS='{ "ollama": { "baseURL":  
"http://localhost:114345/api" } }'
```

RDFOX_MANTIX_Language_Model_ID

This variable specifies the provider and model to be used by the chat feature. The provider must match the provider given in RDFOX_MANTIX_AI_PROVIDERS. The variable should be a string in the format <provider key>:<model name>. Some examples are provided below.

```
RDFOX_MANTIX_LANGUAGE_MODEL_ID=openai:gpt-4.1
```

```
RDFOX_MANTIX_LANGUAGE_MODEL_ID= anthropic:claude-opus-4-7
```

Server Sent Events

The chat feature benefits from using SSE to stream responses from LLMs. SSE keeps an HTTP connection open so the server can push data continuously. Many network components — proxies, firewalls, load balancers, and CDNs — were designed for request/response HTTP and can inadvertently buffer or terminate these long-lived connections. While the chat feature will continue to work, the user experience will be diminished. Please ensure streaming for SSE is supported by your network for the best user experience.

RDFox® 7.7

The chat feature utilizes the RDFox® SEMSIM function. In RDFox® version 7.7 this feature has a minimum hardware requirement of 4 cores to work as expected with the RDFox® Mantix chat feature. This is not the case in newer versions of RDFox®. Please spec your host machine accordingly.

Environment Variables

The following table contains a complete listing of the environment variables that can be used to configure a RDFox® Mantix instance.

Environment Variable	Required?	Accepted Values	Default Value	Description
RDFOX_MANTIX_AUTH_SECRET	Yes	Any String	N/A	This secret is used to encrypt session token held by users to authenticate with RDFox® Mantix A secure secret can be generated with the following ‘openssl’ command: ‘openssl rand -base64 33’

RDFOX_MANTIX_AUTH_URL	Yes	Any URL	N/A	The canonical URL of the application. For more information, see AUTH_URL in the Auth.js Documentation
RDFOX_MANTIX_RDFOX_URL	Yes	Any URL	N/A	The URL of the RDFox® instance. To suppress TLS warnings for non https URL's set RDFOX_MANTIX_RDFOX_URL_ALLOW_INSECURE_CONNECTION=true
RDFOX_MANTIX_AUTH_USE_RDFOX_ROLE_MANAGER	No	true or false	true	If true, allows users to sign in with RDFox® role name and password. If false, disables role name and password authentication for users. Setting this to false and RDFOX_MANTIX_AUTH_OIDC_ISSUER=none will cause authentication errors.
RDFOX_MANTIX_AUTH_OIDC_ISSUER	No	URL or none	none	The Issuer URL for your OIDC provider
RDFOX_MANTIX_AUTH_OIDC_USERINFO	No	URL or none	none	The userinfo endpoint URL for your OIDC provider
RDFOX_MANTIX_AUTH_OIDC_TOKEN	No	URL or none	none	The token endpoint URL for your OIDC provider

RDFOX_MANTIX_AUTH_OIDC_CLIENT_ID	No	Any string	"	The client ID and secret of RDFox® Mantix in your OIDC setup
RDFOX_MANTIX_AUTH_OIDC_CLIENT_SECRET	No	Any string	"	
RDFOX_MANTIX_RDFOX_URL_ALLOW_INSECURE_CONNECTION	No	true or false	false	Set to true if you require non-TLS connection to the RDFox® instance
RDFOX_MANTIX_AI_PROVIDERS	No	JSON string	N/A	See RDFOX_MANTIX_AI_PROVIDERS
RDFOX_MANTIX_LANGUAGE_MODEL_ID	No	string	N/A	See RDFOX_MANTIX_LANGUAGE_MODEL_ID
RDFOX_MANTIX_DATA_FILE_DIRECTORY	No	string	/data/uploaded-data-files	The path where uploaded data files are persisted.
RDFOX_MANTIX_DATA_FILE_MAX_SIZE_BYTES	No	number	unlimited	The maximum file size for uploading files. The default is no limit.
RDFOX_MANTIX_BACKEND_TIMEOUT	No	number	300000	The maximum time that a backend operation may run before it is cancelled. The default is 5 minutes.

Backup and Restore

All RDFox® Mantix data has persisted to your RDFox® instance; therefore, the RDFox® backup recommendations and procedures are applicable. Please refer to the [RDFox® documentation](#) for more information.

Appendix: Troubleshooting

<\> Error Messages in Terminal

Authentication failed error

If you get a terminal error with:

```
Authentication failed for role mantix-service-role used by RDFox® Mantix.
```

It means the environment variable `RDFOX_MANTIX_SERVICE_ROLE` does not correspond to a valid role in RDFox®.

Fix:

- Make sure you have completed all the steps in Step 1 above.
- Create the role in RDFox® and grant it the required privileges as described there.

Incorrect Service Privileges

If you get a terminal error with:

```
The RDFox Mantix service role has incorrect privileges:
```

Then the privileges for the RDFox® Mantix service role are not set up correctly. This could be because the environment variables `RDFOX_MANTIX_SERVICE_ROLE` or `RDFOX_MANTIX_SERVICE_PASSWORD` are incorrect, or it could be that the service role has the incorrect privileges in RDFox®.

If you encounter this error, please check your values for `RDFOX_MANTIX_SERVICE_ROLE` and `RDFOX_MANTIX_SERVICE_PASSWORD`, and perform the necessary privilege adjustments it asks for in the error message.

Incorrect RDFox® Version

If you get an error message with:

```
The RDFox server version (version) is not supported.
```

Then the RDFox® version you are using is not supported by RDFox® Mantix.

RDFox® Mantix supports RDFox® v7.7 or greater release.

Invalid AI Provider configuration

If you get an error message with:

```
Warnings received when validating AI provider config.
```

RDfox® Mantix has been unable to connect with your configured AI provider. Review the information in [Chat for queries](#). This could potentially be a transient connection. If the issue persists, RDfox® Mantix can be started without chat configured by removing the chat configuration settings.

Environment Variables

These errors occur when some environment variable is missing or is given an incorrect value. See [Step 3 – Environment Variables](#) above for a list of the required environment variables.

Environment variable `RDFOX_MANTIX_AUTH_SECRET` is required. Use `'openssl rand -base64 32'` to generate a secret.

This means you haven't provided the `RDFOX_MANTIX_AUTH_SECRET` environment variable when running the Docker image. Please ensure you pass a value for the `RDFOX_MANTIX_AUTH_SECRET` environment variable when starting up the Docker image.

Environment variable `RDFOX_MANTIX_AUTH_URL` is required. Set this to the canonical URL of the application e.g. `https://example.com`

This means you haven't provided the `RDFOX_MANTIX_AUTH_URL` environment variable when running the Docker image. Please ensure you pass a value for the `RDFOX_MANTIX_AUTH_URL` environment variable when starting up the Docker image.

The environment variable `RDFOX_MANTIX_RDFOX_URL` is required and must use 'https'.

- **If you require an insecure connection to RDfox®, set `RDFOX_MANTIX_RDFOX_URL_ALLOW_INSECURE_CONNECTION=true`**

This means that either you haven't provided the `RDFOX_MANTIX_RDFOX_URL` environment variable, or you have given it an incorrect value. Please ensure the URL you provide for `RDFOX_MANTIX_RDFOX_URL` is an 'https' URL. If you require a 'http' URL for RDfox® you must set `RDFOX_MANTIX_RDFOX_URL_ALLOW_INSECURE_CONNECTION=true`.

- If you are setting `RDFOX_MANTIX_RDFOX_URL` as `localhost` for example (`http://localhost:12110`)

The hostname cannot be resolved, it may be due to running RDFox® Mantix inside Docker on macOS while RDFox® runs on the host machine. In this case, use `http://host.docker.internal:12110` instead.

Environment variable `RDFOX_MANTIX__AUTH_USE_RDFOX_ROLE_MANAGER` can only have the values 'true' or 'false'

This means that you have given an incorrect value for the

`RDFOX_MANTIX__AUTH_USE_RDFOX_ROLE_MANAGER` environment variable. This environment variable can only be set to `true` or `false`.

Full Page Error Messages

RDFox® Mantix may give the following error messages.

Can not reach server. Please try again. If error persists, contact your support for assistance.

This error can be caused by a loss of connection to RDFox®. Ensure your RDFox® instance is running and reachable from RDFox® Mantix.

Tuning for Large Datasets

RDFox® Mantix can import large RDF files (over 2GB of Turtle data, and single files contributing over forty thousand terms to the ontology). To successfully add data of this scale you may need to increase the following settings beyond their default values.

- The `RDFOX_MANTIX_BACKEND_TIMEOUT` environment variable
- The RDFox® server parameter `max-memory`
- The RDFox® endpoint parameter `object-keep-alive-time`

Appendix: Supported AI Providers

The chat feature can be used with the following AI providers. Each provider may include specific configuration options which supported in RDFox® Mantix.

AI Providers

Amazon Bedrock

The Amazon Bedrock provider for the AI SDK contains language model support for the Amazon Bedrock APIs

<https://ai-sdk.dev/providers/ai-sdk-providers/amazon-bedrock>

Configuration key: amazonBedrock

Anthropic

The Anthropic provider contains language model support for the Anthropic Messages API.

<https://ai-sdk.dev/providers/ai-sdk-providers/anthropic>

Configuration key: anthropic

Azure OpenAI

The Azure OpenAI provider contains language model support for the Azure OpenAI chat API.

<https://ai-sdk.dev/providers/ai-sdk-providers/azure>

Configuration key: azure

Cohere

The Cohere provider contains language and embedding model support for the Cohere chat API.

<https://ai-sdk.dev/providers/ai-sdk-providers/cohere>

Configuration key: cohere

Vercel AI Gateway

The AI Gateway provider connects you to models from multiple AI providers through a single interface. Instead of integrating with each provider separately, you can access OpenAI, Anthropic, Google, Meta, xAI.

<https://ai-sdk.dev/providers/ai-sdk-providers/ai-gateway>

Configuration key: gateway

Google

The Google Generative AI provider contains language and embedding model support for the Google Generative AI APIs.

<https://ai-sdk.dev/providers/ai-sdk-providers/google-generative-ai>

Configuration key: google

Google Vertex

The Google Vertex provider for the AI SDK contains language model support for the Google Vertex AI APIs.

<https://ai-sdk.dev/providers/ai-sdk-providers/google-vertex>

Configuration key: googleVertex

Hugging Face

The Hugging Face provider offers access to thousands of language models through Hugging Face Inference Providers.

<https://ai-sdk.dev/providers/ai-sdk-providers/huggingface>

Configuration key: huggingFace

Mistral

The Mistral AI provider contains language model support for the Mistral chat API.

<https://ai-sdk.dev/providers/ai-sdk-providers/mistral>

Configuration key: mistral

ollama

A community provider that uses Ollama to provide language model support for the AI SDK.

<https://ai-sdk.dev/providers/community-providers/ollama>

Configuration key: ollama

OpenAI

The OpenAI provider contains language model support for the OpenAI responses, chat, and completion APIs

<https://ai-sdk.dev/providers/ai-sdk-providers/openai>

Configuration key: openai

OpenAI Compatible

You can use the OpenAI Compatible Provider package to use language model providers that implement the OpenAI API.

<https://ai-sdk.dev/providers/openai-compatible-providers>

Configuration key: `openaiCompatible`