Kiuwan On-Premises Distributed System Architecture

Kiuwan On-Premises (KoP) is based on a client-server architecture that includes these major server components:

- Apache web server
- Wildfly application server
- MySQL database
- Redis in-memory data store

Kiuwan On-Premises can be installed on a single host or on multiple hosts (externalizing one, some or all of the provided infrastructure services).

You can also install Kiuwan On-Premises under your own AWS cloud, replacing the provided infrastructure services with compatible AWS services.

Kiuwan On-Premises and its roles

Kiuwan On-Premises defines three different roles for each Kiuwan service instance:

- Kiuwan front: receives and answers HTTPS, HTTP or AJP requests (depending on your configuration) from the Apache web server (or directly from a client, depending on your configuration), both for the web application and REST API calls.
- Kiuwan analyzer: processes analysis results asynchronously.
- Kiuwan scheduler: processes background tasks such as notifications and Kiuwan Insights database updates.

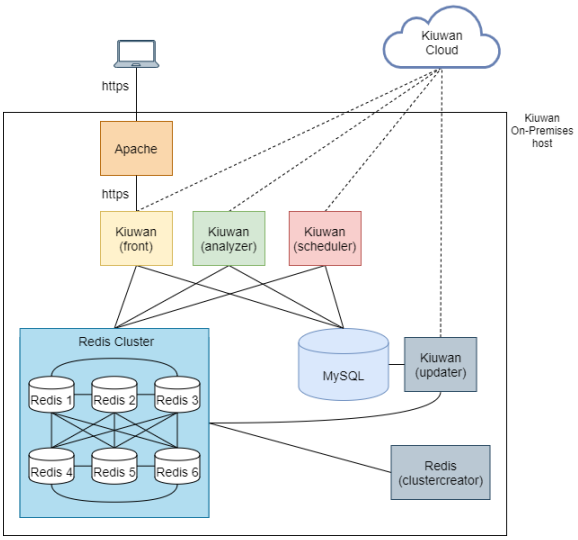
The default installation will create a Kiuwan service instance of each role.

Supplied services

All needed services are supplied as Docker containers. The following table summarizes all the available services:

Service	Docker service name	Description
Kiuwan front	wildflykiuwan-f[n]	A Kiuwan front instance
Kiuwan analyzer	wildflykiuwan-a[n]	A Kiuwan analyzer instance
Kiuwan scheduler	wildflykiuwan-s[n]	A Kiuwan scheduler instance
Kiuwan updater	updaterkiuwan	The Kiuwan container that manages Kiuwan On-Premises installations and updates
Apache	apacheloadbalancer	The Apache web server instance that works as a front-end server and load balancer
MySQL	mysqlkiuwan	The Kiuwan database
Redis	redis_0000[n]	A Redis in-memory data store node
Redis cluster creator	clustercreator	The Redis container that manages the Redis cluster creation

The following diagram shows how Kiuwan On-Premises would be set out **by default** (that is, a single host default installation):



The main purpose of each piece is (service names are shown between parentheses):

- Apache (**apacheloadbalancer**): balances the traffic between Kiuwan frontals (kept in the default installation to allow easy frontal scaling).
- Kiuwan front (**wildflykiuwan-f1**): recieves and processes HTTPS requests.
- Kiuwan analyzer (**wildflykiuwan-a1**): processes analyses.
- Kiuwan scheduler (**wildflykiuwan-s1**): executes background tasks.
- MySQL (**mysqlkiuwan**): stores Kiuwan data.
- Redis cluster (**redis_0000**[**n**]): used by Kiuwan as a data cache.
- Redis cluster creator (**clustercreator**): configures each Redis node as a cluster.
- Kiuwan updater (**updaterkiuwan**): installs, updates and upgrades needed data or configuration.

Note that connection with Kiuwan Cloud is only needed in some situations (see Needed internet connections).

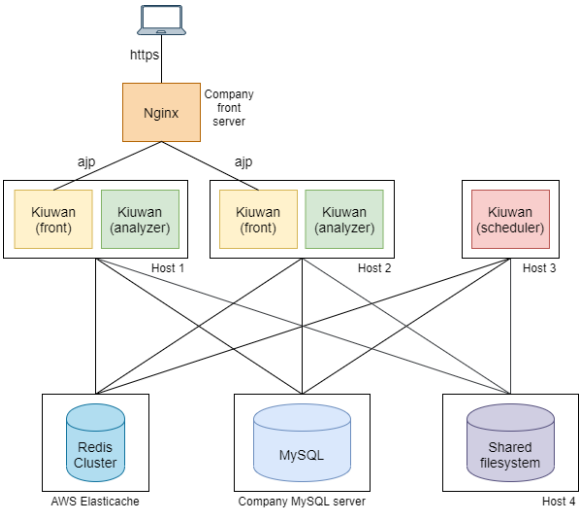
Clustering support

Depending on your needs, the Kiuwan On-Premises architecture supports both clustered environments and single-host installations. You can:

- Keep all services in an all-in-one host installation.
- Decide where each service will be installed, one by one (both Kiuwan services and infrastructure).
- Externalize one, some or all of the provided non-Kiuwan services:
    - Apache load balancer
    - MySQL
    - Redis

All Kiuwan instance containers can be scaled up (to a limit) or down to the default configuration. This means that you can start more containers of each Kiuwan role depending on your load needs.

To illustrate the flexibility that Kiuwan On-Premises installation tool provides, an alternative and possible configuration is depicted in the following image (Kiuwan Cloud and temporal services were ommitted in order to simplify the diagram):



As the picture shows, you can distribute Kiuwan On-Premises services across different hosts and use external services to replace some of the ones provided.

> ⚠ Note that although distributed architecture is feasible, we do not provide explicit documentation on how an installation of this kind has to be performed, as there are plenty of manual steps and usually the single-host approach is valid for most customers. In case your company needs more advanced architectural solutions for a Kiuwan On-Premises Distributed installation, please contact Kiuwan support.