

Collecting endpoint profiles

- [Basic architecture](#)
- [Configuring Kaa](#)
- [Working with Client-side endpoint profiles](#)
- [Working with Server-side endpoint profiles](#)

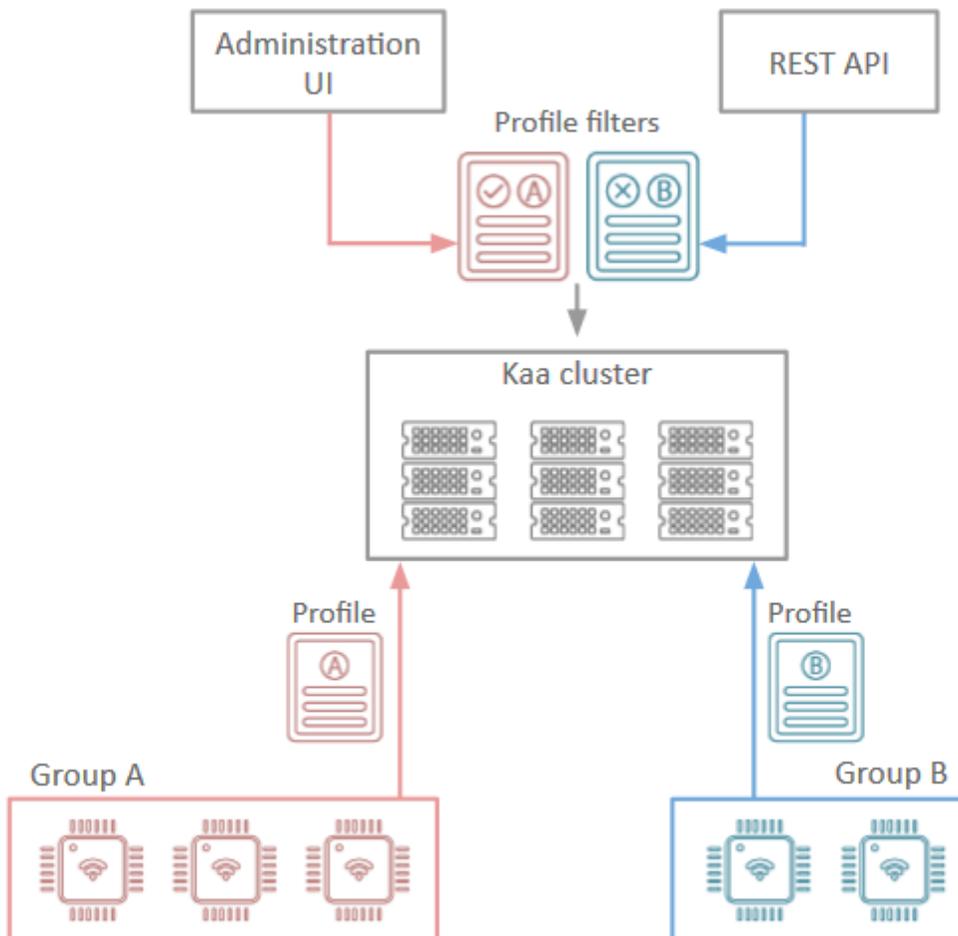
Endpoint profile is a virtual identity or "passport" of the endpoint. By filtering against the data in the profiles, endpoints may be aggregated into independently managed groups. Please review the [Kaa profiling design reference](#) for more background.

This guide will familiarize you with the basic concepts of designing endpoint profiles and programming the Kaa profiling subsystem. It is assumed that you have either set up a [Kaa Sandbox](#), or a [full-blown Kaa cluster](#) already and that you have created a [tenant](#) and an [application](#) in Kaa.

Basic architecture

The following diagram illustrates basic entities and data flows in scope of the endpoint profile management:

- Endpoint profiles are generated based on the [client-side](#) and [server-side](#) endpoint profile schemas and corresponding profile data.
- Endpoints send their client-side profiles to the Kaa server during the [registration](#).
- Server-side applications and scripts may use Kaa REST API in order to modify server-side profile of endpoints.
- The Kaa tenant developer specifies [profile filters](#) for endpoint groups using either [Admin UI](#) or [REST API](#)
- Kaa Operations service classifies endpoints into groups based on their profiles and the group profile filters



Configuring Kaa

The default client-side profile schema installed for Kaa applications is empty. In order to make use of the Kaa profiling / identity management capabilities, you should load at least either client-side or server-side profile schema that reflects the application you are designing. Please note that client-side structure is used during SDK generation and changes to the client-side structure requires generation of new SDK. However, Application developer is able to define and change server-side structure of endpoint profile at any time.

Working with Client-side endpoint profiles

Think about the client-side profile schema as of a structured data set of your endpoint application that will later be available to you in Kaa server and may change due to your client application logic or device state.

You can configure your own client-side profile schema using the [Admin UI](#) or [REST API](#). For the purpose of this guide we will use a fairly abstract client-side profile schema shown below.

```
{
  "type": "record",
  "name": "Profile",
  "namespace": "org.kaaproject.kaa.schema.sample.profile",
  "fields": [
    {
      "name": "id",
      "type": "string"
    },
    {
      "name": "os",
      "type": {
        "type": "enum",
        "name": "OS",
        "symbols": [
          "Android",
          "iOS",
          "Linux"
        ]
      }
    },
    {
      "name": "os_version",
      "type": "string"
    },
    {
      "name": "build",
      "type": "string"
    }
  ]
}
```

Client-side endpoint profile updates are reported to the endpoint SDK using a profile container. The profile related API varies depending on the target SDK platform, however the general approach is the same.

Working with Server-side endpoint profiles

Think about the server-side profile schema as of a set of your endpoint properties the are controlled by your server-side applications. For example, client subscription plan, device activation flag, etc. You may also use server-side endpoint profile to store properties that are set during manufacturing and should not be controlled by client application.

You can configure your own server-side profile schema using the [Admin UI](#) or [REST API](#). For the purpose of this guide we will use a fairly abstract server-side profile schema shown below.

```
{
  "type": "record",
  "name": "ServerProfile",
  "namespace": "org.kaaproject.kaa.schema.sample.profile",
  "fields": [
    {
      "name": "subscriptionPlan",
      "type": "string"
    },
    {
      "name": "activationFlag",
      "type": "boolean"
    }
  ]
}
```

Once this schema is configured, you are able to assign server-side endpoint profile body to certain endpoints based on their ids using [Admin UI](#) or [REST API](#).