

Endpoint profiling

- Client-side endpoint profile
 - Client-side endpoint profile update
 - Client-side endpoint profile example
- Server-side endpoint profile
 - Server-side endpoint profile update
 - Server-side endpoint profile example
- System part of endpoint profile

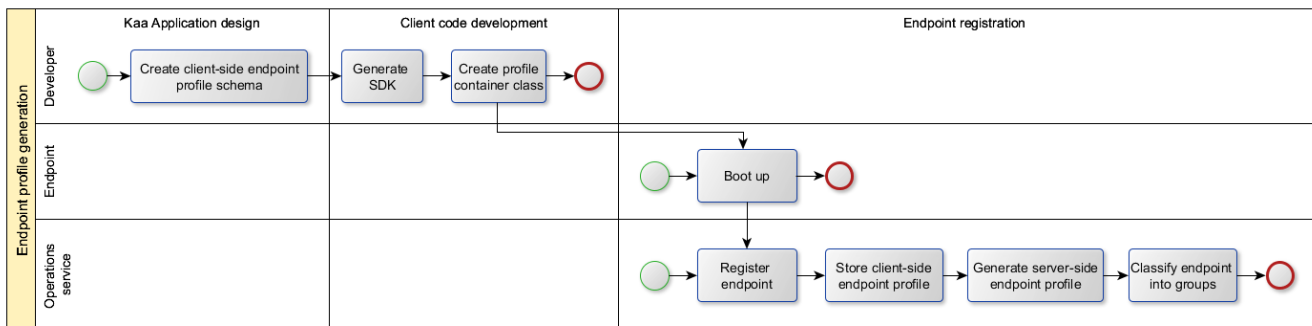
The *endpoint profile* is a customizable structured data set that describes specific characteristics of the endpoint. Endpoint profiles are used to classify endpoints into endpoint groups and are comprised of the client-side, server-side and system part. The structure of both client-side and server-side of endpoint profile is defined by application developer using the [Apache Avro schema](#) format. Application developer may reuse and share certain data structures using [CTL](#).

Client-side structure is used during SDK generation and changes to the client-side structure requires generation of new SDK. Application developer is able to define and change server-side structure of endpoint profile at any time. The structure of the system part is identical across the applications and is used by Kaa internally for its functions. Both client-side and server-side profile schemas are maintained within the corresponding application, with its own version that distinguishes it from the previous schemas. Multiple schema versions and corresponding endpoint profiles created upon those schemas can coexist within a single application.

Client-side endpoint profile

The client-side endpoint profile is initially generated at the stage of a new endpoint registration. Prior to that, the client-side endpoint profile values should be specified by the client developer using the Kaa endpoint SDK. The client-side endpoint profile can be also updated at run time using SDK API call. In this case, SDK executes profile update request and endpoint membership in the endpoint groups is re-evaluated and updated to match the new endpoint profile.

The client-side endpoint profile is unidirectionally synchronized, and thus should **not** be considered as a means to temporarily store endpoint data in the Kaa cluster. There is no way for the endpoint to retrieve the profile information back from the Kaa cluster. At its start, the application must fill in the current endpoint profile with the up-to-date data and execute corresponding SDK API calls. The endpoint SDK does not persist the profile information over the endpoint reboots. However, it detects profile data changes and submits the new data to the Kaa cluster as a profile update.



Client-side endpoint profile update

When the endpoint profile information changes as the result of the client operation or user's actions, it is the client implementation responsibility to update the profile via the endpoint SDK API calls. The endpoint SDK detects profile changes by comparing the new profile hash against the previously persisted one. Should there be a change, the endpoint profile management module passes it to the Operations server, which in turn updates the endpoint profile information in the database and revises the endpoint groups membership.

Client-side endpoint profile example

The following code block provides a simple client-side endpoint profile schema example.

```

{
  "namespace": "org.myproject",
  "type": "record",
  "name": "MyClientSideProfile",
  "fields": [
    { "name": "country", "type": "string" },
    { "name": "city", "type": "string" },
    { "name": "age", "type": "int" },
    {
      "name": "hobbies",
      "type": { "type": "array", "items": "string" }
    }
  ]
}

```

The following client-side profile would be compatible with our schema example.

```

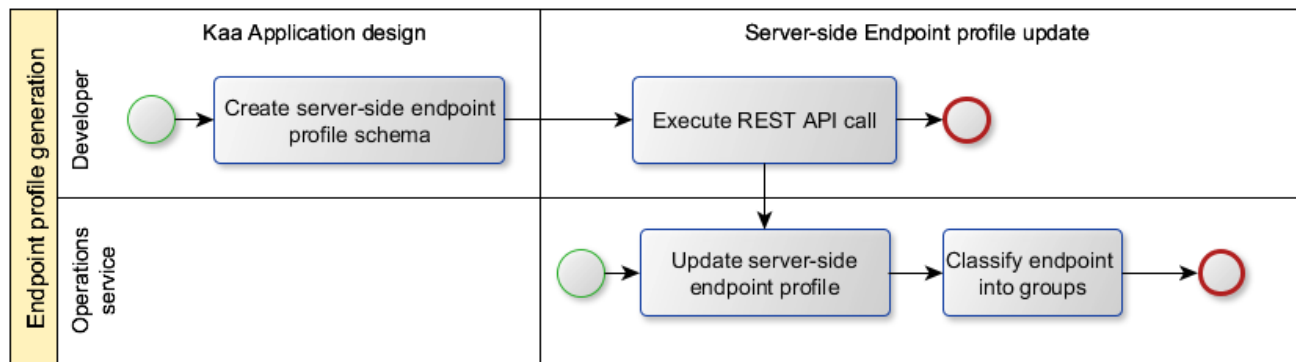
{
  "country": "US",
  "city": "San Francisco",
  "age": 32,
  "hobbies": [
    "skydiving",
    "hiking"
  ]
}

```

The schema structure from our example allows filtering the endpoints by the owner's country (for example, to show only US news), city (for example, to push weather notifications only for the specified cities), age (for example, to apply age restrictions), and hobbies (for example, to push football scores for those whose hobby list contains "football"). It is allowed to create complex filtering conditions by combining as many filtering conditions as needed.

Server-side endpoint profile

The server-side endpoint profile is initially generated at the stage of a new endpoint registration. By default, server side profile record is auto-generated based on the latest server-side profile schema of particular application. Both server-side endpoint profile schema and data can be updated at run time using REST API call. In this case, endpoint membership in the endpoint groups is re-evaluated and updated to match the new endpoint profile.



Server-side endpoint profile update

Server-side endpoint profile is designed to be accessed and modified by various server-side applications through Kaa REST API integration layer. Server-side profile is not accessible from endpoint SDK or other client application logic.

Server-side endpoint profile example

The following code block provides a simple server-side profile schema example.

```
{
  "namespace": "org.myproject",
  "type": "record",
  "name": "MyServerSideProfile",
  "fields": [
    { "name": "serialNumber", "type": "string" },
    { "name": "customerId", "type": "string" }
  ]
}
```

The following client-side profile would be compatible with our schema example.

```
{
  "serialNumber": "SN-777",
  "customerId": "Customer A"
}
```

The schema structure from our example allows filtering the endpoints by the serial number (SN) and customer ID. For example, you may remotely turn on/off features for certain customers or deactivate device with certain SN. It is allowed to create complex filtering conditions by combining as many filtering conditions as needed.

System part of endpoint profile

The system part of the endpoint profile is used by Kaa internally for its functions and contains the following information.

Property	Description
Endpoint Public Key	The public key used for security purposes to validate endpoint requests
SDK token	The endpoint SDK identifier
Client-side Profile Hash	The client-side profile hash used to validate integrity of endpoint profile state
Server-side Profile Hash	The server-side profile hash used to validate integrity of endpoint profile state
Endpoint group state	The list of endpoint groups that contain current endpoint.
Server Key Hash	The hash of the server that was used during the last endpoint request