Red Hat 3scale 2.1

Terminology

For Use with Red Hat 3scale 2.1

Last Updated: 2018-04-08

Red Hat 3scale 2.1 Terminology

For Use with Red Hat 3scale 2.1

Legal Notice

Abstract

This guide documents the terminology used with Red Hat 3scale 2.1.

Table of Contents

CHAPTER 1. 3SCALE TERMINOLOGY

This section acts as a glossary of common terms used within the 3scale platform.

1.1. ACCESS TOKENS

Access tokens are keys used to authenticate when calling the Account Management API, the Analytics API and the Billing API.

1.2. ACTIVEDOCS

ActiveDocs is the 3scale specification for documenting REST APIs. ActiveDocs is based on the popular Swagger framework. With ActiveDocs, APIs get interactive documentation which makes it easier for developers to understand web services and also to test them without any installation.

1.3. ADMIN PORTAL

The Admin Portal is the 3scale dashboard. From the Admin Portal, API providers can configure the integration of their API with 3scale, manage their application plans, give access to internal members and external customer, limit traffic per application keys, customize their developer portal… This is the central console for API providers to manage and secure the access to their API.

1.4. API

An API is a logical bundle of one or more methods which are accessible. A given API provider may expose one or more such APIs. The term API is used interchangeably with the word Service.

3scale also has it's own APIs which are located in the 3scale ActiveDocs, available in your Admin Portal, under the **Documentation → 3scale API Docs** section.

1.5. APICAST GATEWAY

APIcast is an NGINX based API gateway used to integrate your internal and external API services with 3scale API Management Platform. An API Gateway is the interface that handles API requests. Depending on its configuration it can handle access control, rate limiting, security filtering, logging, routing, caching…

For more information about APIcast and its deployment options (hosted, self-managed and different configuration options), check out the APIcast Overview.

1.6. API CREDENTIALS

Credentials are a set of keys, secrets or identifiers associated with an Application which permit the application to access an API. The form the credentials take depends upon the authentication pattern in use for the API.

1.7. API CONSUMER

An individual, group or company that accesses an API provided by an API provider. The term may refer both to the organization and the software applications written by the organizations to use the API. A given organization may have one or more applications which access the API.

1.8. API KEY

An API Key is a type of credential for an application to be allowed to make calls on a specific API. API Keys are a specific type of authentication pattern.

- **Application API key**: Generated when a new application is created on 3scale (via user or API). Provides access to the API(s). The type of access is determined by an application plan.
- **Provider API key**: Generated when a new provider account is created on 3scale. Gives full access to the 3scale API(s) associated with the provider account.

NOTE

We strongly suggest using service tokens to authenticate against the Service Management API and access tokens to authenticate against the Account Management API, the Analytics API and the Billing API. Their custom/limited access scopes make them inherently more secure than using this provider API key.

1.9. API PROVIDER

In our documentation and guides, the “API provider” (3scale customer) is the individual, group, or company that owns one or more APIs and manages them using the 3scale API Management Platform. An API provider may make their APIs accessible to other teams internally within their organization or externally to third-party developers, partners, or the public.

1.10. APPLICATION

An application is piece of software code which is developed by an API Consumer to access an API. The application will normally have associated to it within the 3scale system a unique set of api credentials for the API, a traffic history of the calls sent to the API and meta-data captured at Application creation. Applications are linked to developer accounts.

1.11. AUTHENTICATION

Authentication is the process of verifying the identity of a user or server requesting access. This process precedes the authorization process which determines the resources that can be accessed.

1.12. AUTHENTICATION PATTERN

A technical scheme used for a given API to express the credentials needed by an API Consumer to access a methods on an API. Each API within 3scale can have one and only one authentication pattern selected.

1.13. BASE URL

This is the host URL e.g. https://my-api.com

- **Public base URL**: This is the root of the endpoint which is called by the API clients.
- **Private base URL**: This is the internal host for the API backend services.

1.14. CODE PLUGIN

Plugins provide a wrapper for the 3scale API that enables API Access Control and API Traffic Reporting without having to run another server. A plugin lives inside the code that powers your API. The plugin calls the 3scale API for Access Control & Traffic Reporting.

1.15. DEPLOYMENT OPTIONS

Same as “Integration options”. In 3scale, we usually refer to ‘deployment options’ when talking about ways of deploying the APIcast gateway (not the 3scale API Management Platform).

1.16. DEVELOPER ACCOUNT

Developer accounts are the accounts subscribed to a particular API. Developer Accounts are the parents of the applications. When new developers subscribe to an API, an application is automatically created for them which will allow them to make calls to the API it is subscribed to.

1.17. DEVELOPER PORTAL

The developer portal is the site where developers can subscribe to APIs. From the developer portal, developers can manage their subscription, have access to their API key, create applications, access the interactive API documentation (ActiveDocs), see their API consumption, etc. The 3scale CMS, with its out of the box features allows to quickly create a developer portal with all what is required for on-boarding new customers.

1.18. ENDPOINT (API ENDPOINT)

An endpoint is a specific call or transaction which can be made on an API by an application. A method usually corresponds to one specific action that can be taken such as getting a list of objects or creating a new object. An API normally has multiple endpoints. The term is equivalent to the "method" term.

1.19. END USERS

End users are the users of an application which calls one or multiple API(s).

1.20. FIELD DEFINITIONS

From the Field Definition section in the Admin Portal it is possible to add and/or create new fields for gathering data from internal admins/members, from developer accounts and from applications. These are the fields which are displayed when a new account is created. For example, to the developer signup form on the developer portal, additional fields can be added, such as for example ‘address’. These fields can be made optional or required.

1.21. INTEGRATION ERRORS

Integration errors are generated by 3scale when calls are performed with an incorrect set of credentials or token, or for calls with incorrect Ids, URLs… These errors might be errors coming from the application which is calling the API or can be a configuration error with the integration of the API with 3scale.

1.22. INTEGRATION OPTIONS

Integration options, also known as deployment options, are the available options for integrating an API with the 3scale Management Platform (NGINX, plugins…).

1.23. MAPPING RULES

The mapping rules map incoming calls from specific endpoints to the corresponding methods and metrics created in 3scale.

Usage tracking, endpoint access and limits are based on the methods and metric[metrics] that are configured with these mapping rules.

See our documentation on Mapping Rules.

1.24. METHOD (API METHOD)

Methods let you track the usage of your API on 3scale. You can add a method for each of the HTTP methods available on the API endpoints for your API. Methods are hit-based. By default on 3scale, method calls trigger the built-in Hits-metric. To track other types of events you can add Metrics that report in different units. Usage limits and pricing rules for individual methods are defined from within each Application plan.

See our documentation Defining Your API (Methods And Metrics).

1.25. METRIC

Metrics let you track specific calls to your API by mapping them to one or more URL patterns in the Mapping Rules section of the integration page. Metrics are cumulative and not discrete. The built-in top level metric on 3scale is Hits. Other top level metrics can be added if needed.

See our documentation Defining Your API (Methods And Metrics).

1.26. MONETIZATION

The term “Monetization” or “API monetization” refers to the fact of charging customers based on API access and/or on usage. API Monetization can be done through plans.

1.27. PLAN

Plans are used for granting access to specific APIs and endpoints, limiting traffic and monetizing API usage. 3scale has four different types of plans that can be used on their own or in conjunction:

- **Application plans:** This is the most common type of plans in 3scale. They let you configure access rights to an API by specifying rate limits and pricing rules. All applications must be associated with a plan. Application plans can be customized for one application. To learn how to customize an application plan for one application, see the /docs/api-bizops/customize-plans['Customize plans' tutorial].
- **Account plans:** They establish pricing and features on the account level and are thus not limited to a specific API service. Account plans can be customized for one account.
- **Service plans:** They establish pricing & features on the service level and are thus not limited to a specific application.
- **End-user plans:** They establish usage limits and pricing rules for end users of an API. That allows you to balance the allowed number of hits specified in the application plans, preventing one single user to consume all the quota by themselves.

1.28. POTENTIAL UPGRADE

Potential upgrades are developer accounts that triggered traffic limit alerts. These developer accounts could potentially be upgraded to a plan meeting their needs in terms of traffic volume.

1.29. PRICING RULE

A pricing rule is a logical expression set to determine the cost of a particular transaction in the system. To charge for use of your API, you will create pricing rules within plans to determine billing and levels of access.

1.30. RATE LIMIT OR USAGE LIMIT

A rate limit is associated with a plan and is a logical expression which definies a threshold on usage by API Consumers of a particular method, endpoint or metric.

These limits are configured in the Application Plans, you can read more here.

1.31. SERVICE

In 3scale, both the term 'API' and 'Service' are used to refer to API services. An API service is a logical bundle of one or more methods (or endpoints) which are accessible. A given API provider may expose one or more such API services.

3scale also has it's own APIs in the 3scale ActiveDocs, available in your Admin Portal, under the **Documentation → 3scale API Docs** section.

1.32. SERVICE TOKENS

Service tokens are keys used to authenticate when calling the 3scale Service Management API.

1.33. SUBSCRIPTION

A contract (plan) between an account and a service.

1.34. SWAGGER

See ActiveDocs.

1.35. WEBHOOK

Webhooks are processes which are triggered following an event, webhooks work as an instant notification mechanism.

See our documentation on Webhooks.