

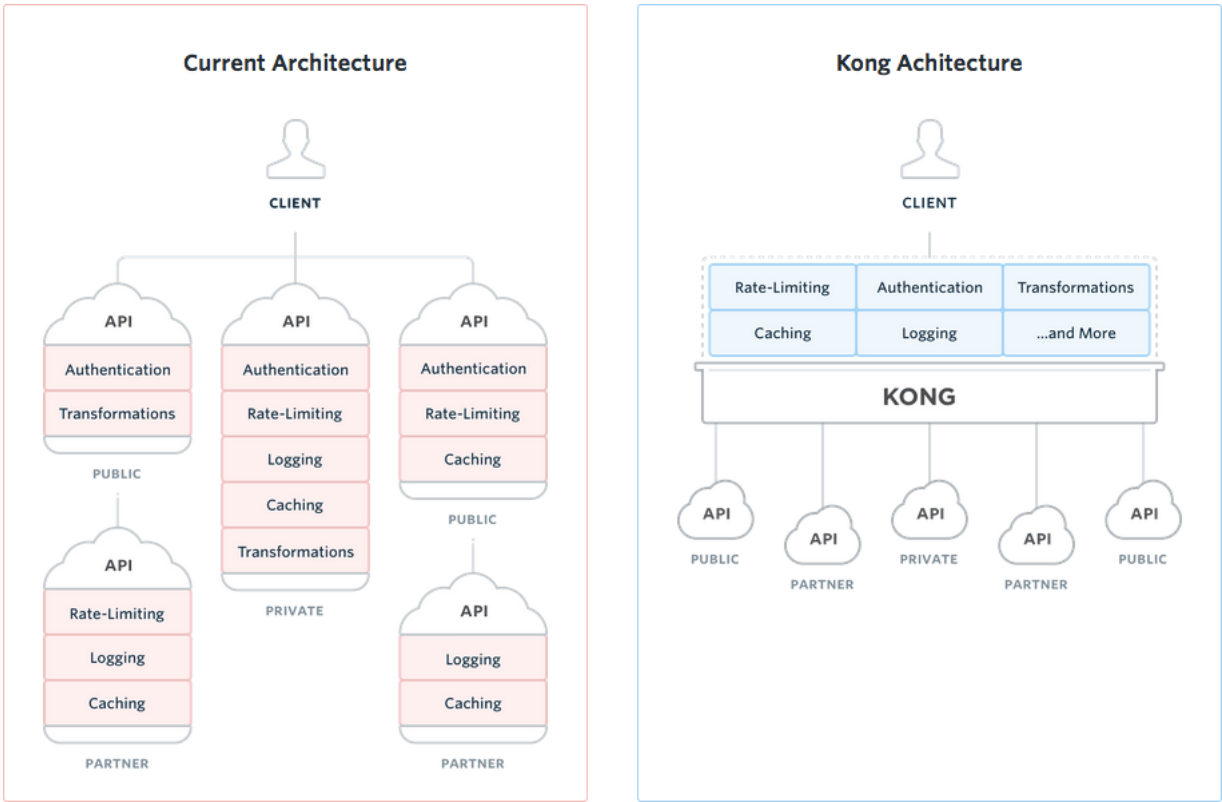
AegonThtf API Gateway

一、简介

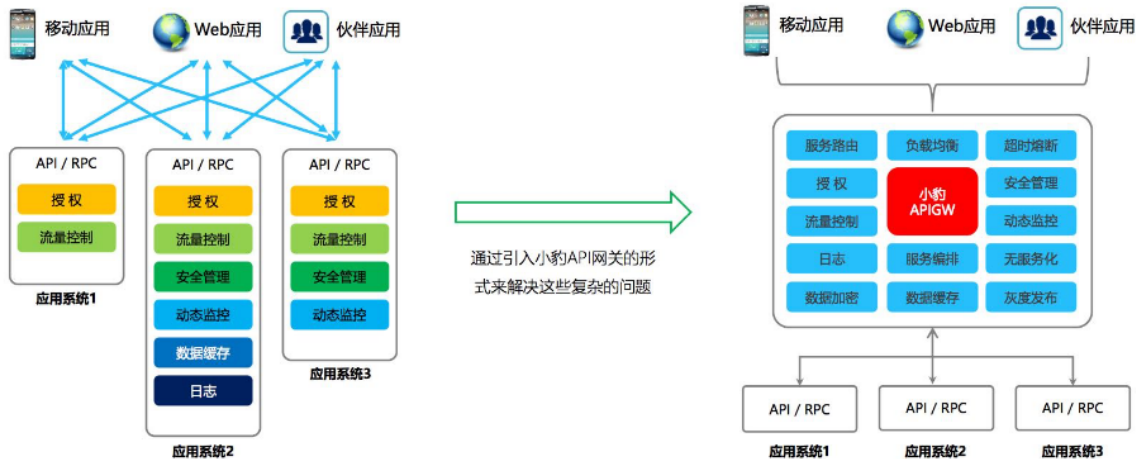
1. 什么是API网关

不知道API网关的确切定义是什么，从了解到的信息以及从实际工作的关切来看，期望API网关能够帮助我们达成如下的目标：

- 1. 为公司提供一个统一的API接入点，所有访问API全部通过这个统一的入口访问，而该入口是可横向扩展高可用的。
- 2. 通过API网关，能够方便建设更多的以Restful API形式提供的服务，更方便实现模块化的系统服务形式。
- 3. 通过对外API网关，可以更好的帮助前后端分离的互联网应用架构在实践中得到应用和实践。
- 4. 通过内部API网关，除内部应用也可以实现前后端分离架构外，也可以使得应用间的交互可以通过API的方式进行。
- 5. 通过API网关, 可以集中的实现认证、授权，流量控制，访问IP白名单、IP控制，熔断, 访问日志记录和分析, 方便跨域访问，请求或响应更改等功能，让开发人员更多的集中于业务逻辑代码的编写，将系统级的功能更多的放到系统级平台上实现。
- 6. 通过API网关，使得微服务架构更加完整，除了具备Zuul这种代码级的网关，能够具备更易于使用的平台级网关产品，使得架构更加的完整和实用



再借用人家一张图尝试说明API网关在系统架构中的位置



✗ 各个应用系统开发了重复的功能

✓ API网关集成公共的功能

✗ 各应用间的调用关系不清晰，很难维护系统

✓ 调用关系在系统中清晰展示，方便维护系统

✗ 一个服务崩溃可能造成其他服务雪崩

✓ 超时熔断等机制保证系统的可用性

✗ 没有统一监控无法了解整体情况

✓ 统一监控了解整体情况，并提供异常告警

2. 关于Kong

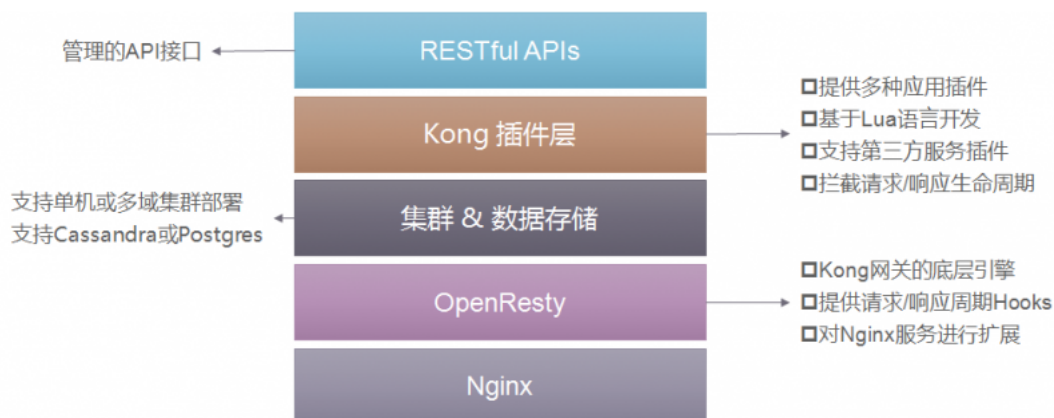
API网关也有多种实现，开源的也有，比如Kong, Zuul等，商用的也有。众多公有云平台也都提供了不同形式的API网关服务。

Kong，是由Mashape公司开源的，基于Nginx的API gateway，通过众多的插件对经由网关的访问施加通用功能影响达成目标。

Kong API的目前理解的主要模块有如下三个：

- Service 业务功能的原始地址（这个地址可以使用其他技术方案，实现仅能被API网关所访问）
- Route API的访问地址，一个Service可以配置多种访问路径。也就是一个Service可以有多个地址可以访问。
- Consumer 访问者。
- Plugins 各种插件。

二、 Kong的架构



三、Kong网关请求流程

为了方便地理解Kong工作原理，使用如下图示说明Kong网关的API接口的典型请求工作流程：



当Kong运行时，每个对API的请求都先到API网关（多实例部署复杂均衡），然后这个请求将会被代理转发到最终的API接口。在请求（Requests）和响应（Responses）之间，Kong将会执行已经事先安装和配置好的插件，比如进行认证、授权、流控以及对请求或响应进行加工。

四、Kong 安装

基于Kubernetes平台，以Helm Chart的形式安装了Kong API community version, 并安装了Konga为 Kong API的管理控制台, 两个模块的数据库分别使用Postgresql和Mongo。具体安装步骤在此不详细描述。

AegonThtf API 域名地址：<https://api.aegonthtf.com>
AegonThtf API 控制台地址：<http://api-dash.internal.aegonthtf.com>

可以看到基于K8s平台API网关具备灵活的横向扩展能力

kubernetes

Search

[+ CREATE](#) |

Workloads > Deployments

Cluster

Namespaces

Nodes

Persistent Volumes

Roles

Storage Classes

Namespace

apigateway

Overview

Workloads

Cron Jobs

Daemon Sets

Deployments

Jobs

Pods

Replica Sets

Name	Labels	Pods	Age	Images
kong-mongo-express	k8s-app: kong-mongo-express	1 / 1	5 days	registry.aegonthtf.com/database/mongo-express:lat
kong-dashboard-konga	k8s-app: kong-dashboard-konga	1 / 1	5 days	registry.aegonthtf.com/thirdparty/konga:0.12.0
apigateway-konga-db-mongodb	app: mongodb chart: mongodb-4.0.4 heritage: Tiller release: apigateway-konga-db	1 / 1	5 days	registry.aegonthtf.com/database/mongodb:3.6.6
kong-api-gateway-kong	app: kong chart: kong-0.3.0 heritage: Tiller release: kong-api-gateway	2 / 2	5 days	registry.aegonthtf.com/thirdparty/kong:0.14.0-alpine
kong-api-gateway-postgresql	app: kong-api-gateway-postgresql chart: postgresql-0.10.0 heritage: Tiller release: kong-api-gateway	1 / 1	5 days	postgres:9.6.2

五. 试用案例

Kong拥有不少插件，大部分免费使用，少部分需要付费（当然这部分基本我们也不会用），一一试用和编写文档需要不少的时间，所以此处仅以少量的测试Case来对其功能做大致评测，当实际应用时可以视需要做进一步的深入评测。

其主要插件如下图示：

Authentication

Protect your services with an authentication layer

Basic Auth



Add Basic Authentication to your APIs

ADD PLUGIN

Key Auth



Add a key authentication to your APIs

ADD PLUGIN

Oauth2



Add an OAuth 2.0 authentication to your APIs

ADD PLUGIN

Hmac Auth



Add HMAC Authentication to your APIs

ADD PLUGIN

Jwt



Verify and authenticate JSON Web Tokens

ADD PLUGIN

Ldap Auth



Integrate Kong with a LDAP server

ADD PLUGIN

Plugins added in this section will be applied Globally.
- If you need to add plugins to a specific Service or Route, you can do it in the respective section.
- If you need to add plugins to a specific Consumer, you can do it in the respective Consumer's page.



Security

Protect your services with additional security layers

Acl



Control which consumers can access APIs

ADD PLUGIN

Cors



Allow developers to make requests from the browser

ADD PLUGIN

Ip Restriction



Whitelist or blacklist IPs that can make requests

ADD PLUGIN

Bot Detection



Detects and blocks bots or custom clients

ADD PLUGIN

Plugins added in this section will be applied Globally.
- If you need to add plugins to a specific Service or Route, you can do it in the respective section.
- If you need to add plugins to a specific Consumer, you can do it in the respective Consumer's page.



Authentication Security **Traffic Control** Serverless Analytics & Monitoring Transformations Logging Custom

Traffic Control

Manage, throttle and restrict inbound and outbound API traffic

Rate Limiting



Rate-limit how many HTTP requests a developer can make

ADD PLUGIN

Response Ratelimiting



Rate-Limiting based on a custom response header value

ADD PLUGIN

Request Size Limiting



Block requests with bodies greater than a specific size

ADD PLUGIN

Request Termination



This plugin terminates incoming requests with a specified status code...

ADD PLUGIN

Authentication Security Traffic Control Serverless **Analytics & Monitoring** Transformations Logging Custom

Analytics & Monitoring

Visualize, inspect and monitor APIs and microservices traffic

Datadog



Visualize API metrics on Datadog

ADD PLUGIN

Prometheus



Expose metrics related to Kong and proxied upstream services in...

ADD PLUGIN

Zipkin



Propagate Zipkin distributed tracing spans, and report spans to a Zipkin...

ADD PLUGIN

Authentication Security Traffic Control Serverless Analytics & Monitoring **Transformations** Logging Custom

Transformations

Transform request and responses on the fly on Kong

Request Transformer



Modify the request before hitting the upstream server

ADD PLUGIN

Response Transformer



Modify the upstream response before returning it to the client

ADD PLUGIN

Correlation Id



Correlate requests and responses using a unique ID

ADD PLUGIN

AuthenticationSecurityTraffic ControlServerlessAnalytics & MonitoringTransformationsLoggingCustom

Logging

Log requests and response data using the best transport for your infrastructure

Tcp Log

Send request and response logs to a TCP server

ADD PLUGIN

Udp Log

Send request and response logs to an UDP server

ADD PLUGIN

Http Log

Send request and response logs to an HTTP server

ADD PLUGIN

File Log

Append request and response data to a log file on disk

ADD PLUGIN

Syslog

Send request and response logs to Syslog

ADD PLUGIN

Statsd

Send request and response logs to StatsD

ADD PLUGIN

Loggly

Send request and response logs to Loggly

ADD PLUGIN

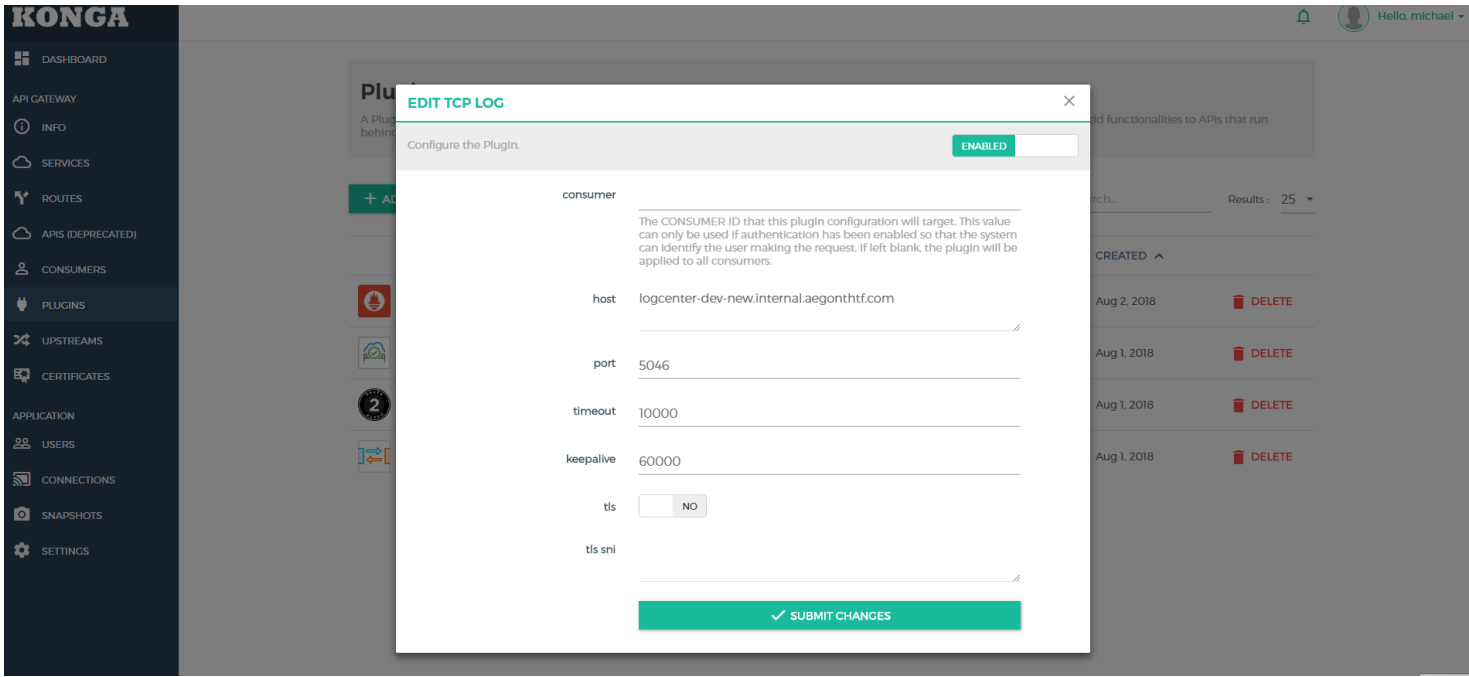
测试案例说明

序号	原始地址	API地址 1	API地址2
1	http://10.72.243.126:31702	https://api.aegonthtf.com/v1/greeting	
2	http://aegonthtf-encrypt-bali.internal.aegonthtf.com/EncryptService/Encrypt.do	https://api.aegonthtf.com/v1/security/encrypt	https://api.aegonthtf.com/v1/security/enc

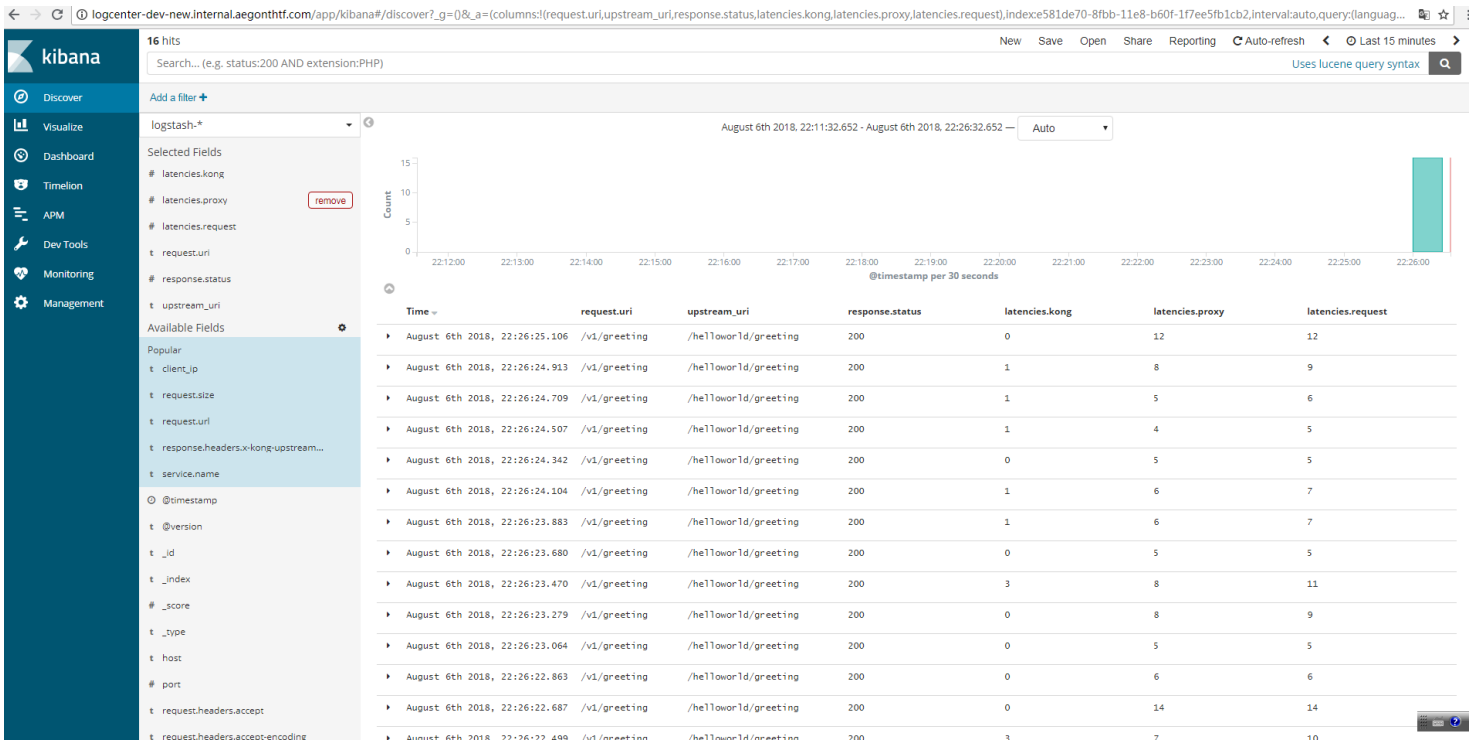
测试案例

1. API访问日志

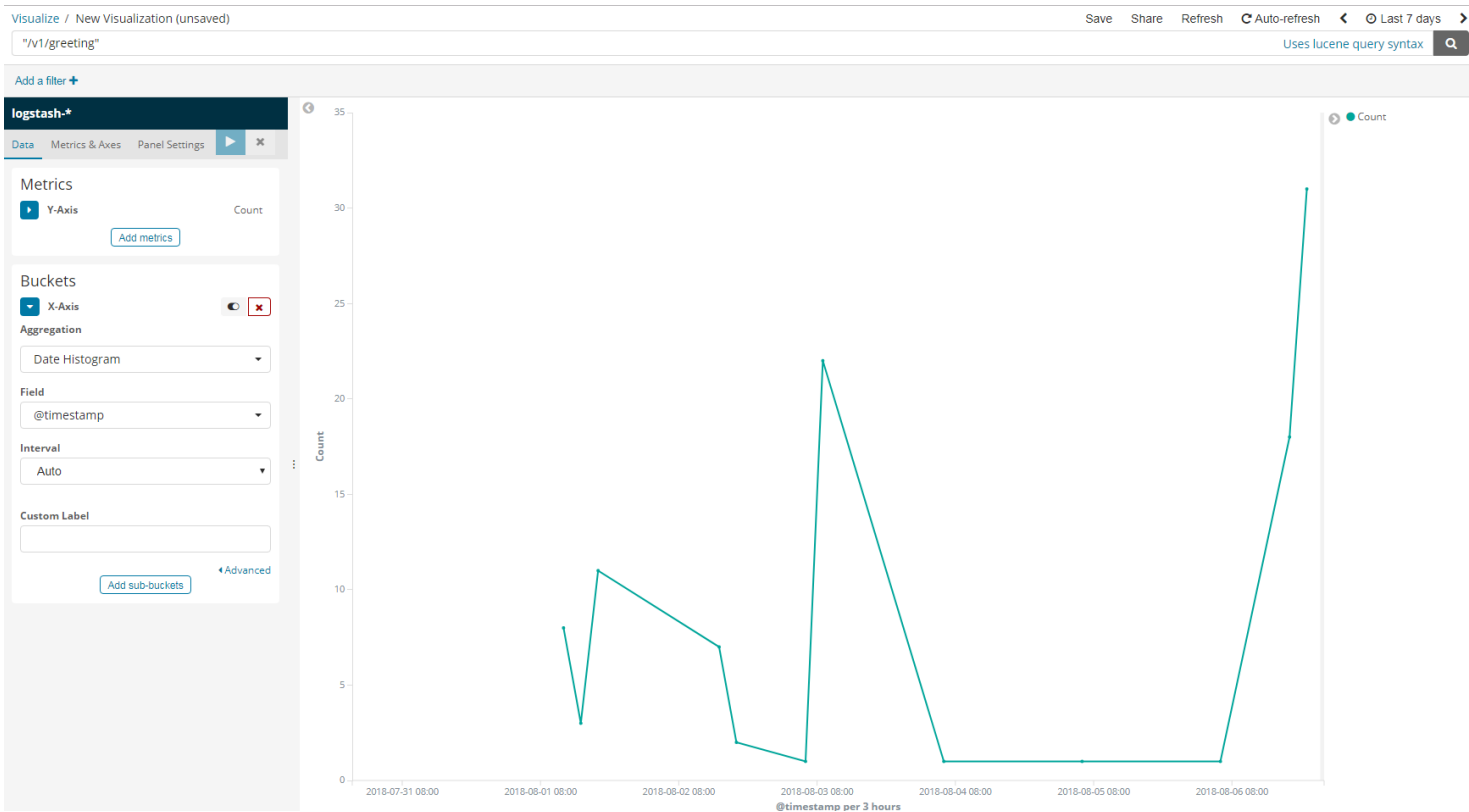
配置如下，为网关配置了全局的日志插件，将API的访问日志全部通过TCP输往ELK日志中心



访问几次<https://api.aegonthtf.com/v1/greeting>后，到日志中心查看，可以查看到访问的日志，如下



基于访问日志，可做访问量及访问响应时间等指标的分析



2. API key 认证

为一个Consumer配置 API Key. (实际场景, 即为每个可以访问我们API的用途比如应用分配一个Consumer, 并为其单独生成API key, 可确保非授权的应用不能访问我们的API)

GA

ARD

S

(PRECATED)

ERS

AMS

ATES

CONSUMER: michaelzhang

consumers / edit consumer

Details Groups Credentials Accessible APIs Accessible Services Accessible Routes Plugins

BASIC

API KEYS

HMAC

OAUTH2

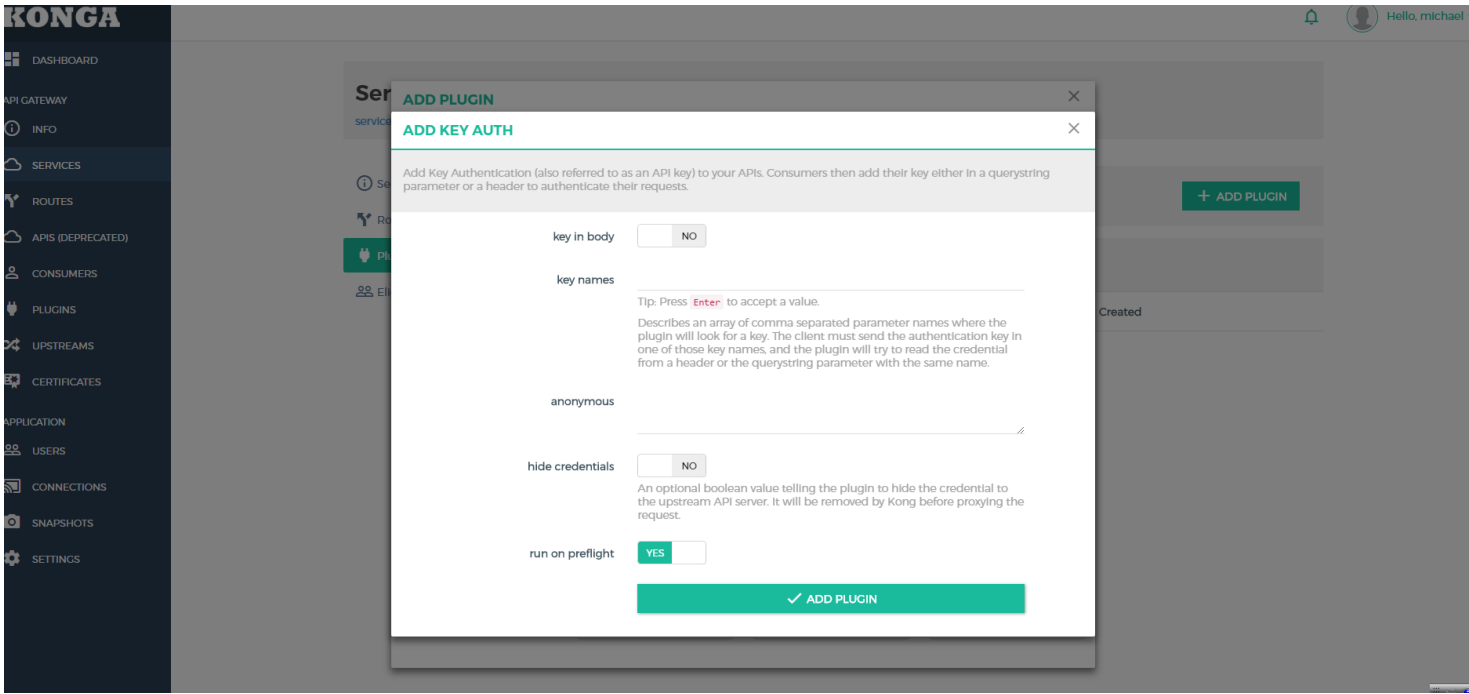
JWT

Api Keys

+ CREATE API KEY

#	key	created
1.	wtBdWHRmzedf5TPCz8WfVCNn8Z3qxZ	Aug 6, 2018 DELETE

为Greeting API增加API Key认证

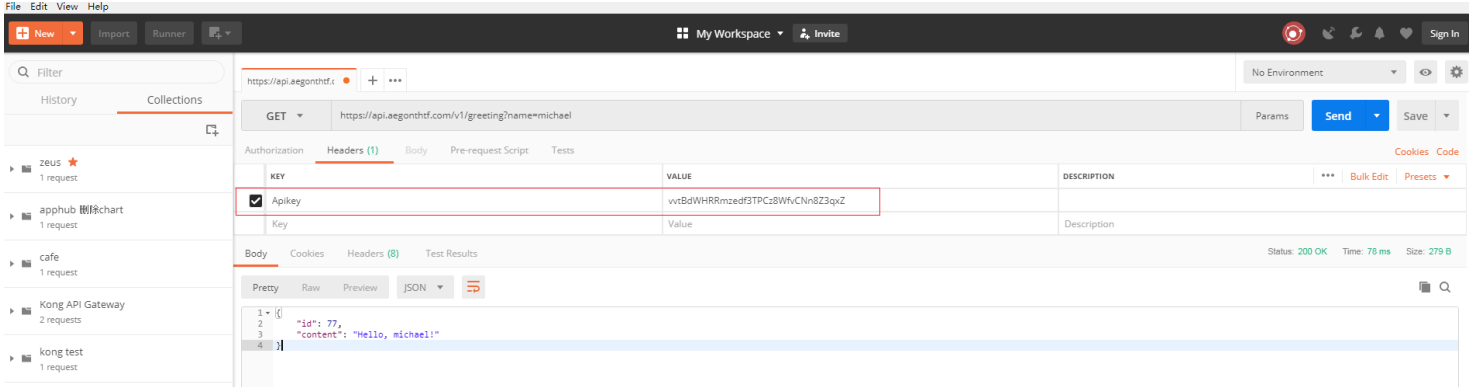


此时，直接访问 <https://api.aegonhtf.com/v1/greeting> 则会拒绝访问

```

{
  message: "No API key found in request"
}
```

而在访问时增加Apikey Header后，则可以正常访问



3. Oauth2 认证











为encrypt服务的/v1/security/encrypt访问路径设置了Oauth2 认证（仅设置了Password类别做测试用，实际支持Oauth2四种类别的认证）和跨域，如下

Plugins

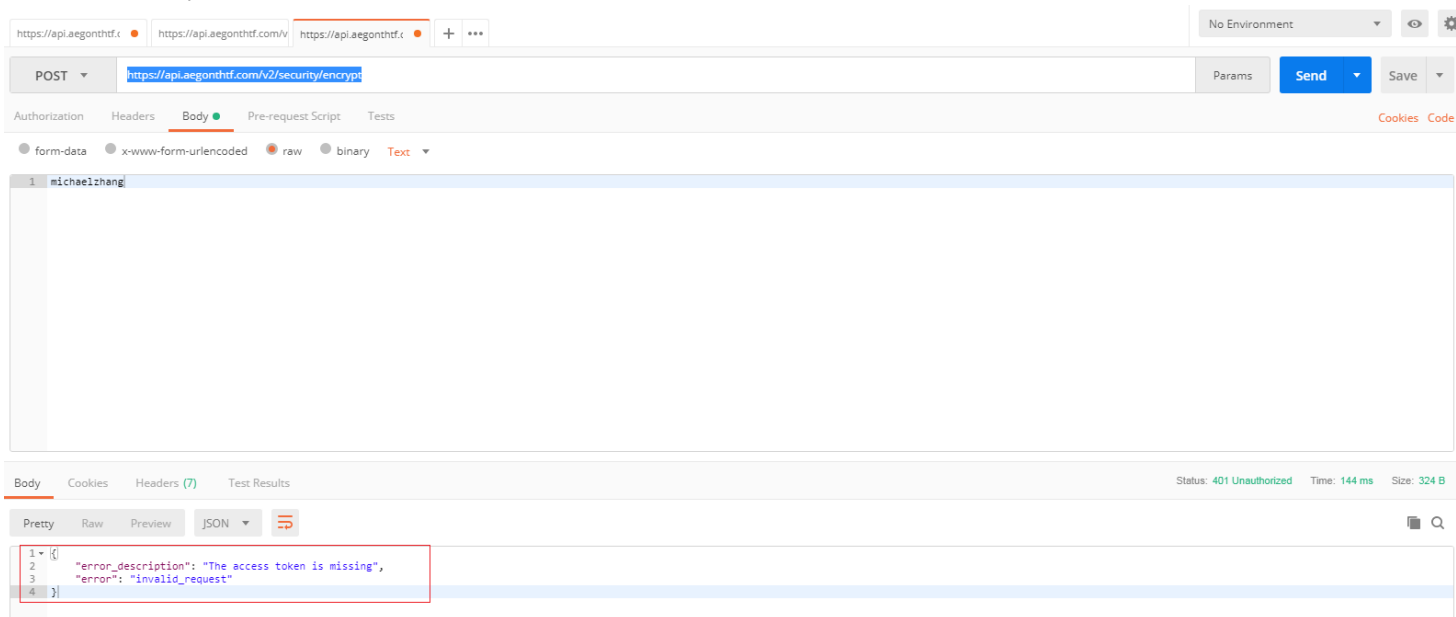
A Plugin entity represents a plugin configuration that will be executed during the HTTP request/response workflow, and it's how you can add functionalities to APIs that run behind Kong, like Authentication or Rate Limiting for example.

[+ ADD GLOBAL PLUGINS](#)

search... Results: 25

	NAME	SCOPE	APPLY TO	CONSUMER	CREATED	
	key-auth	services	441e3691-1a69-422b-aa4a-92d819fda886	All consumers	Aug 6, 2018	 DELETE
	prometheus	global	All Entrypoints	All consumers	Aug 2, 2018	 DELETE
	cors	routes	a428bccb-68f9-4041-bd7c-f57f7a6d8b91	All consumers	Aug 1, 2018	 DELETE
	oauth2	routes	a428bccb-68f9-4041-bd7c-f57f7a6d8b91	All consumers	Aug 1, 2018	 DELETE
	tcp-log	global	All Entrypoints	All consumers	Aug 1, 2018	 DELETE

此时访问加密服务api的v2路径，则会被拒绝访问，显示如下：



https://api.aegonhdf.com/v2/security/encrypt

POST

Authorization Headers Body Pre-request Script Tests

form-data x-www-form-urlencoded raw binary Text

Body Cookies Headers (7) Test Results

Status: 401 Unauthorized Time: 144 ms Size: 324 B

```
1 {
2   "error_description": "The access token is missing",
3   "error": "invalid_request"
4 }
```

可以通过对应的Token端点获取Token后，使用Token再次访问，则就可以成功访问

https://api.aegonthf.com/v2/security/encrypt/oauth2/token

POST https://api.aegonthf.com/v2/security/encrypt/oauth2/token

Authorization Headers Body Pre-request Script Tests

form-data x-www-form-urlencoded raw binary

KEY	VALUE	DESCRIPTION
client_id	ReyTDRkCWooSasPDv1a8wHPTIEib66nL	
client_secret	adnc85VQLPMX75jIED95WfnizsF3Une3	
provision_key	function	
authenticated_userid	f3742e94-9b22-42ee-a24e-98e125c3f27b	
grant_type	password	
scope	read	
Key	Value	Description

Body Cookies Headers (8) Test Results

Status: 200 OK Time: 169 ms Size: 381 B

Pretty Raw Preview JSON

```
1 {
2   "refresh_token": "QkLUPTGwlcZ3RCLN0a1QINyId1GhJb515",
3   "token_type": "bearer",
4   "access_token": "jtdMh1K65Z1D55HQ9H0x8KZnp83UX4w",
5   "expires_in": 7200
6 }
```

https://api.aegonthf.com/v2/security/encrypt

POST https://api.aegonthf.com/v2/security/encrypt

Authorization Headers (1) Body Pre-request Script Tests

TYPE OAuth 2.0 Access Token 1Q4VdGVaOlVwUV1N1sD52wrQFCKjDRV Available Tokens

The authorization data will be automatically generated when you send the request. Learn more about authorization

Add authorization data to Request Headers

Preview Request

Body Cookies Headers (10) Test Results

Status: 200 OK Time: 127 ms Size: 407 B

Pretty Raw Preview JSON

```
1 {
2   "code": "0",
3   "message": "请求成功！",
4   "data": "/sL5kkRyqfK48Pduqx40A=="
5 }
```

以ajax方式实现oauth2认证后访问可实现（最简单的示例，实际应该更复杂）

localhost:63342/oauth2-demo/index.html?_ijt=55sq4ce5c5f5vle9ccn60khhjl

Get Token

IPaaz9PsxfvQltxHhZix0VX8BFYrWo3F

Run

Results:

```
{\code\:"0\","message\:"请求成功！\","data\:"tdm4HhiWlZ7GPxT5TpY7w==\"}
```

html、js 页面代码如下

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>JavaScript Test Page</title>
  <script src="http://code.jquery.com/jquery-latest.min.js"
    type="text/JavaScript">
  </script>
  <script type="text/JavaScript" language="JavaScript">

    //Get inContact Token
    var accessToken = '';
    var baseURI = '';

    function getToken() {
      var url_base =
        'https://api.aegonhttf.com/v2/security/encrypt/oauth2/token';

      // The auth_token is the base64 encoded string for the API
      // application.
      //var auth_token = 'jack:123456';

      //auth_token = window.btoa(auth_token);
      var requestPayload = {
        // Enter your inContact credentials for the 'username' and
        // 'password' fields.
        'grant_type': 'password',
        'client_id': 'ReyTDRkCWooSasPDv1a8wHPTIEIb66nL',
        'client_secret': 'adnc85VQLPMX75j1ED9SWfnizsF3Une3',
        'provision_key': 'function',
        'scope': 'read',
        'authenticated_userid': 'f3742a94-9b22-42ee-a24e-98e125c3f27b'
      }

      alert("dddd")
      $.ajax({
        'url': url_base,
        'type': 'POST',
        'content-Type': 'form-data',
        'dataType': 'json',
        /* 'headers': {
          // Use access_token previously retrieved from inContact token
          // service.
          'Authorization': 'basic ' + auth_token
        }, */
        'data': requestPayload,
        'success': function (result) {
          //Process success actions
          accessToken = result.access_token;
          baseURI = result.resource_server_base_uri;
          alert('Success!\r\nAccess Token:\r' + accessToken +
            '\r\nBase URI:\r' + baseURI)
          document.getElementById('pageDiv').innerHTML = result.access_token;
          return result;
        },
        'error': function (XMLHttpRequest, textStatus, errorThrown) {
          //Process error actions
          alert('Error: ' + errorThrown);
          console.log(XMLHttpRequest.status + ' ' +
            XMLHttpRequest.statusText);
          return false;
        }
      });
    }

    // PUT CALL BELOW HERE!!!

    // BU Agents List
    function getEncryptedInfo() {
      // The baseURI variable is created by the result.base_server_base_uri
      // which is returned when getting a token and should be used to
      // create the url_base.
      /* var url_base = 'https://api.internal.aegonhttf.com';
      $.ajax({
        'url': url_base + '/v1/security/encrypt',
        'type': 'POST',

```

```

        'content-Type': 'raw',
        'dataType': 'json',
        'data': 'ddd',
        'headers': {
            // Use access_token previously retrieved from inContact token
            // service.
            'Authorization': 'Bearer YcNlk11sHbb2oBguCJTaB9M3fVSpIARy',
            'Access-Control-Allow-Origin': '*'
        },
        'success': function (result) {
            //Process success actions
            var returnResult = JSON.stringify(result);
            alert('Success!\r\n' + returnResult);
            document.getElementById('callResults').innerHTML = returnResult;
            return result;
        },
        'error': function (XMLHttpRequest, textStatus, errorThrown) {
            //Process error actions
            alert('Error: ' + errorThrown);
            console.log(XMLHttpRequest.status + ' ' +
                XMLHttpRequest.statusText);
            return false;
        }
    });

/* var url = 'https://api.internal.aegonthtf.com/v1/security/encrypt';
var xhr = createCORSRequest('POST', url);
xhr.setRequestHeader(
    'Authorization', 'Bearer YcNlk11sHbb2oBguCJTaB9M3fVSpIARy');
xhr.send(); */

var url_base =
    'https://api.aegonthtf.com';

// The auth_token is the base64 encoded string for the API
// application.

$.ajax({
    'url': url_base + '/v2/security/encrypt',
    'type': 'POST',
    'content-Type': 'text/plain',
    'dataType': 'text',
    'headers': {
        // Use access_token previously retrieved from inContact token
        // service.
        'Authorization': 'Bearer ' + accessToken
    },
    'data': 'ddd',
    'success': function (result) {
        var returnResult = JSON.stringify(result);
        alert('Success!\r\n' + returnResult);
        document.getElementById('callResults').innerHTML = returnResult;
        return result;
    },
    'error': function (XMLHttpRequest, textStatus, errorThrown) {
        //Process error actions
        alert('Error: ' + errorThrown);
        console.log(XMLHttpRequest.status + ' ' +
            XMLHttpRequest.statusText);
        return false;
    }
});

}

//END CALL ABOVE HERE

</script>
<style>
.outer {
    width: 960px;
    min-height: 45px;
    height: auto;

```

```

        color: black;
        background-color: lightgray;
        border: 1px solid black;
        padding: 5px;
        word-wrap: break-word;
    }
</style>
</head>
<body>

<!-- List of Agents within a BU -->
<button onclick="getToken()">Get Token</button>
<br />
<div class="outer" id="pageDiv"></div>
<br />
<!-- Make sure to update the "button onclick" with the method being
    tested -->
<button onclick="getEncryptedInfo()">Run</button>
<br />Results:
<br />
<div class="outer" id="callResults"></div>
</body>
</html>

```

4. 流量控制 Rate Limiting

为greeting api增加流量控制的设置，每分钟仅允许访问60次

Route 0a6389e5-3927-4889-baf0-bdac17e4ebef

[routes](#) / [route](#)

Route Details

Plugins

Eligible consumers beta

Assigned plugins

search plugins...

Name	Consumer	Created	
<div><div><div><div></div><div></div></div><div><div></div><div></div></div></div><div><div></div><div></div></div></div> <div>rate-limiting</div>	All consumers	Aug 6, 2018	<div><div></div><div>DELETE</div></div>

从Postman中发起120次访问（访问很快，1分钟内），可以从下图中看出从第61次后访问失败，并提示超过流量控制，所以无法正确返回结果了。因为测试案例未配置日志输往日志中心，所以未能验证，但相信后面60次访问不会被访问实际的业务服务了，从网关出应该会被弹回了。

Collection Runner Run Results My Workspace Run In Command Line Docs

0 PASSED 0 FAILED greeting No Environment just now Run Summary Export Results Retry New

Iteration 55	GET	https://api.aegonhtf.com/v1/greeting?name=michael	https://api.aegonhtf.co...	...f.com/v1/greeting?name=michael	200 OK	27 ms	38 B	54	
This request does not have any tests.									
Iteration 57	GET	https://api.aegonhtf.com/v1/greeting?name=michael	https://api.aegonhtf.co...	...f.com/v1/greeting?name=michael	200 OK	23 ms	38 B	56	
This request does not have any tests.									
Iteration 58	GET	https://api.aegonhtf.com/v1/greeting?name=michael	https://api.aegonhtf.co...	...f.com/v1/greeting?name=michael	200 OK	23 ms	38 B	57	
This request does not have any tests.									
Iteration 59	GET	https://api.aegonhtf.com/v1/greeting?name=michael	https://api.aegonhtf.co...	...f.com/v1/greeting?name=michael	200 OK	53 ms	38 B	58	
This request does not have any tests.									
Iteration 60	GET	https://api.aegonhtf.com/v1/greeting?name=michael	https://api.aegonhtf.co...	...f.com/v1/greeting?name=michael	200 OK	18 ms	38 B	59	
This request does not have any tests.									
Iteration 61	GET	https://api.aegonhtf.com/v1/greeting?name=michael	https://api.aegonhtf.co...	...f.com/v1/greeting?name=michael	429 Too Many Requests	15 ms	38 B	60	
This request does not have any tests.									
Iteration 62	Request URL	Request Headers (7)							61
Iteration 62	Request Body	Request Headers (7)							62
Iteration 62	Response Headers (7)	Response Headers (7)							63
Iteration 62	Response Body	Response Body							64
Iteration 63	GET	{	"message": "API rate limit exceeded"					65	
This request does not have any tests.									

其他更多的功能暂时就没有一一测试和编写文档了，但预估大部分功能可以使用。

可以使用开源API网关用于部分先行的影响较小的项目，积累更多经验后，可更多的使用并转向商用的或云上的Api网关产品。