Blueking-ESB项目总结

通过自定义组件抓取知乎热帖

论坛ID：lekoood

QQ：@□》:\_\_\_\_\_\_T。

昵称：唐一

目录

[Blueking-ESB项目总结 1](#_Toc478536973)

[通过自定义组件抓取知乎热帖 1](#_Toc478536974)

[3.1 目录结构 3](#_Toc478536975)

[3.2 toolkit.configs.py中配置后面需要的环境变量 3](#_Toc478536976)

[3.3 自定义的数据处理逻辑 3](#_Toc478536977)

[3.4 类中做两件事：数据校验和请求处理 4](#_Toc478536978)

[3.5 数据校验 - Form 5](#_Toc478536979)

[3.6 请求的处理 - handle 6](#_Toc478536980)

[4.1 创建手动接入组件需要的通道 7](#_Toc478536981)

[4.2 接口API 8](#_Toc478536982)

[4.3 访问测试 8](#_Toc478536983)

[5. 组件自助接入（对手动创建的组件通道进行自助接入） 10](#_Toc478536984)

[5.1 申请 10](#_Toc478536985)

[5.2 接口API 12](#_Toc478536986)

[5.3 访问测试 13](#_Toc478536987)

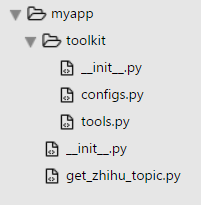
[6. 代码 13](#_Toc478536988)

[7. 问题 13](#_Toc478536989)

1. 添加自定义组件的步骤  
   **(1) 创建系统  
   (2) 把模块代码放到paas机器ESB项目的components/generic/apis/下  
   (3) 重启dashboard让系统注册添加的模块**  
   **(4) 创建通道**
2. 创建自定义组件所属的系统



1. 编写自定义组件代码
   1. 目录结构



* 1. toolkit.configs.py中配置后面需要的环境变量



* 1. 自定义的数据处理逻辑

（返回值格式是自定义的，GetZhihuTopic类的handle方法中可以通过自定义返回数据中的标识判断是否成功）

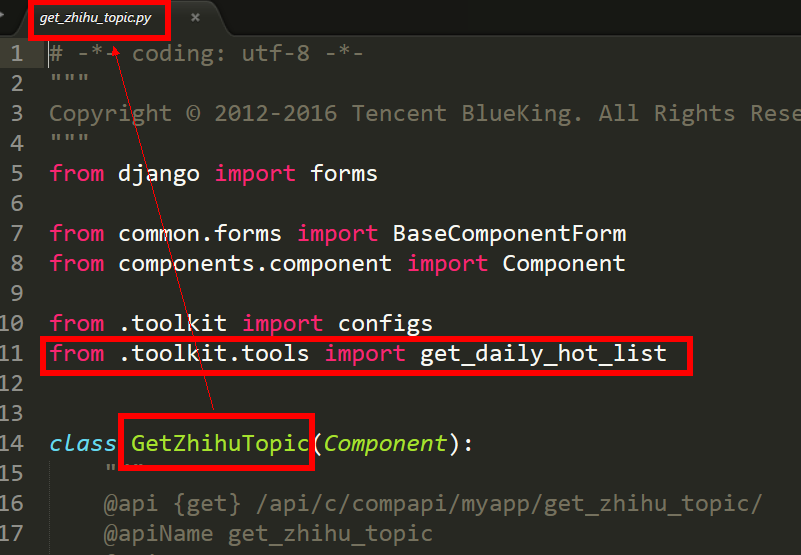


* 1. 类中做两件事：数据校验和请求处理

(1) 类和模块（文件名）的命名

(2) 类名采用驼峰命名方式，模块名是类名中间的大写字母变小写，同时添加一个下划线

(3) 配置通道时填写的都是模块名



* 1. 数据校验 - Form

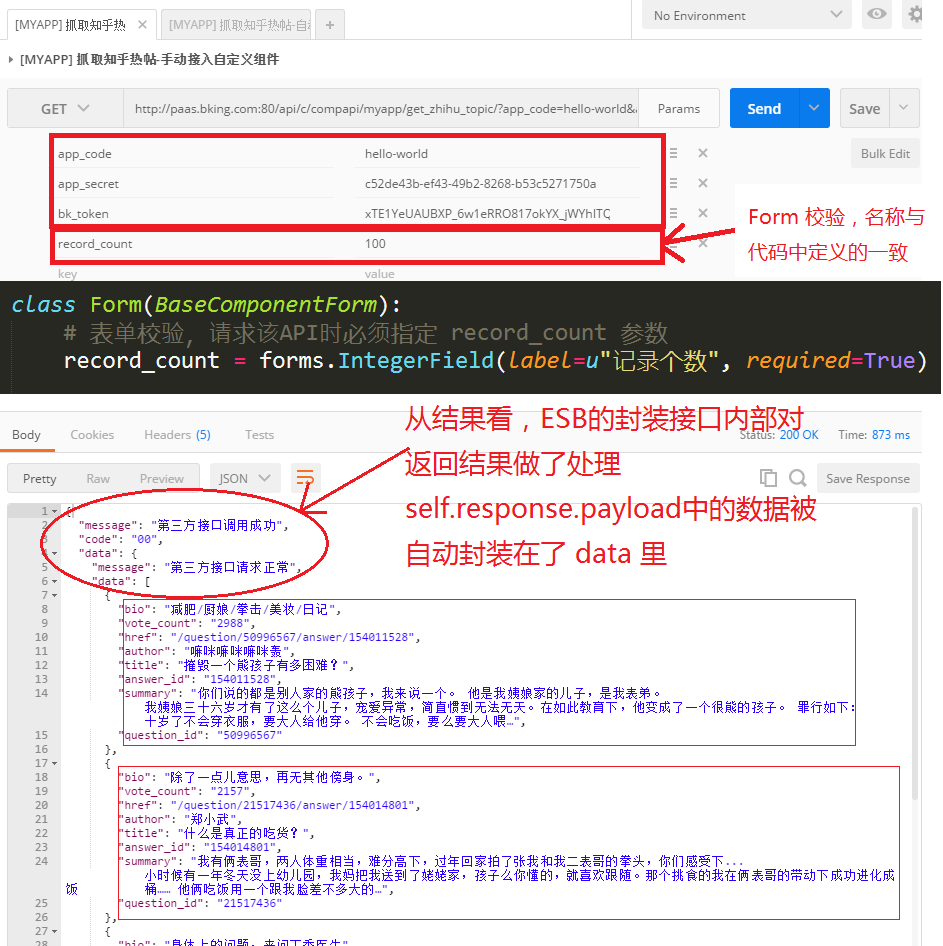
(1) 通过API发出的request请求附带的参数都在这里进行校验（Form校验可以省略）required=True表示必填

(2) 校验后的数据放在self.cleaned\_data字典里，key是参数的名称

(3) self.form\_data是 Form中的clean方法返回的json数据



看一眼POSTMAN的请求界面帮助理解



* 1. 请求的处理 - handle

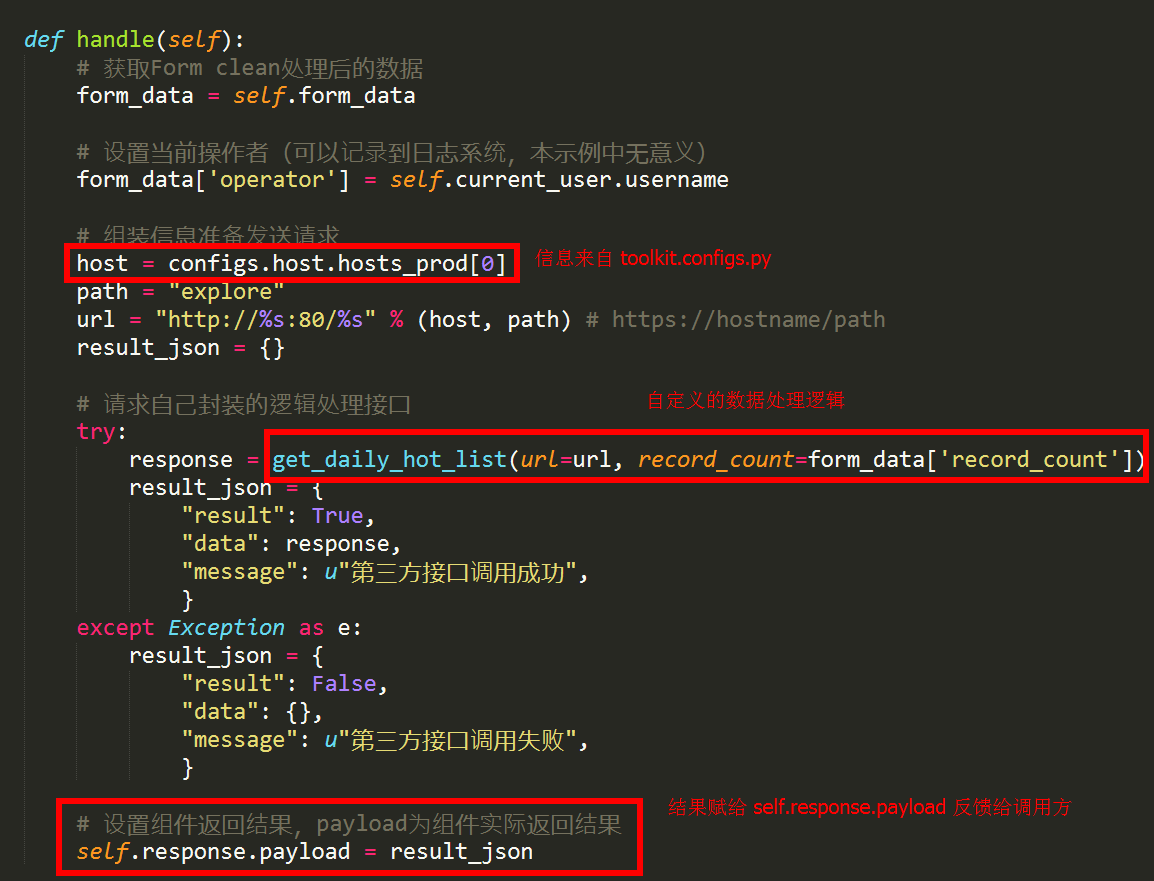
通过API访问的请求，最终在handle方法里完成（逻辑处理，结果数据返回）。

从这里可以看到在toolkit.configs.py中配置的环境变量的作用（web指南提供的demo里，host是SmartHost类型，本次实验可以直接把host配置为字符串格式方便使用）。

同时也可以看到如何进行自定义业务逻辑的处理。

要返回给请求端的数据赋给self.response.payload即可（为了方便处理，返回json格式）。

从上一张图片中可以看出，返回给调用者的数据中，除了在handle方法里赋给self.response.payload的数据外，还会多处2项：code和request\_id（这里了解一下即可）。



至此自定义组件的开发已完成，把开发好的组件放到ESB项目的components/generic/apis/下（默认在s/open\_paas/src/esb/components/generic/apis下）

然后重启dashboard（默认：/data/paas/open\_paas; ./bin/dashboard.sh restart esb）

接下来是手动添加自定义组件的通道

1. 手动添加自定义组件的通道
   1. 创建手动接入组件需要的通道

在通道管理中新建通道

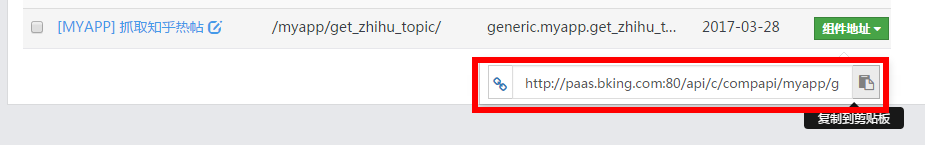
1. 通道名称：任意，符合提示的要求即可
2. 通道路径：/系统名/模块名/
3. 所属组件系统：选择自定义组件myapp所在的系统名
4. 对应组件代号：generic.系统名.模块名



* 1. 接口API

保存后，在通道管理页面可以看到刚才创建的手动接入通道的API接口：

手动接入通道产生的API接口路径以**/api/c/compapi**开头



* 1. 访问测试

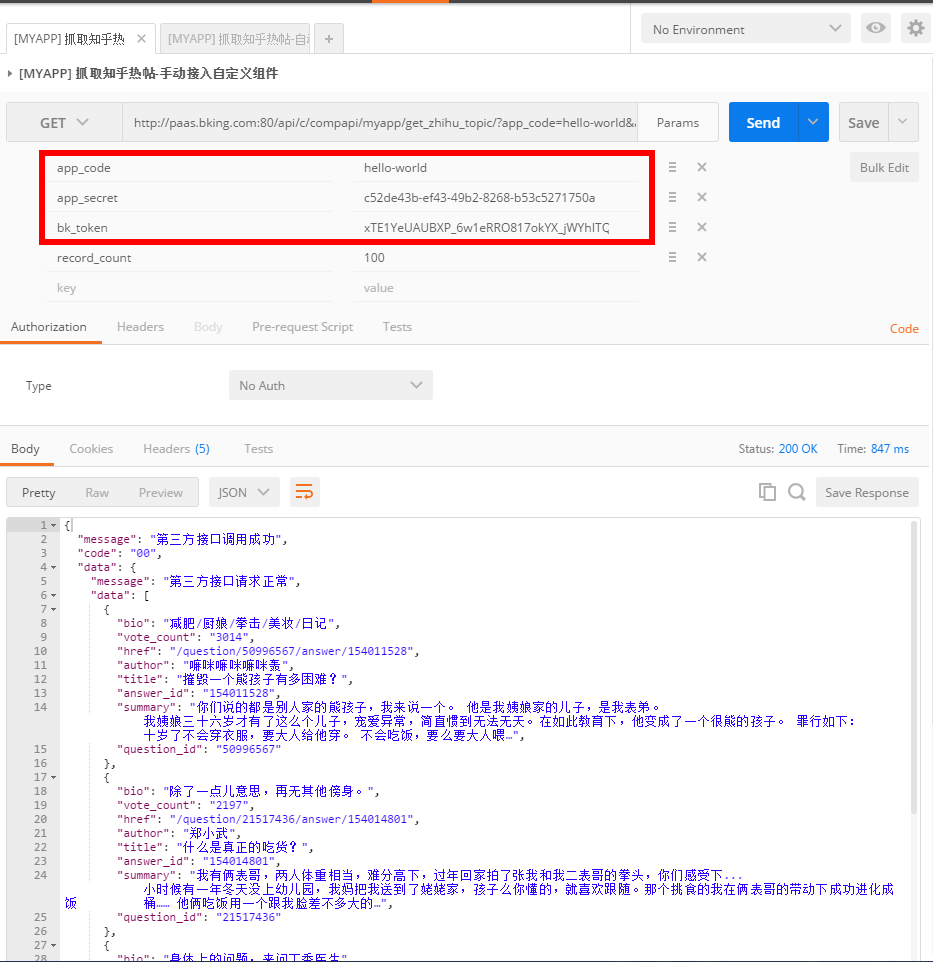
接下来使用POSTMAN访问这个接口，触发自定义组件handle方法中的处理逻辑并获取返回的数据：

app\_code：

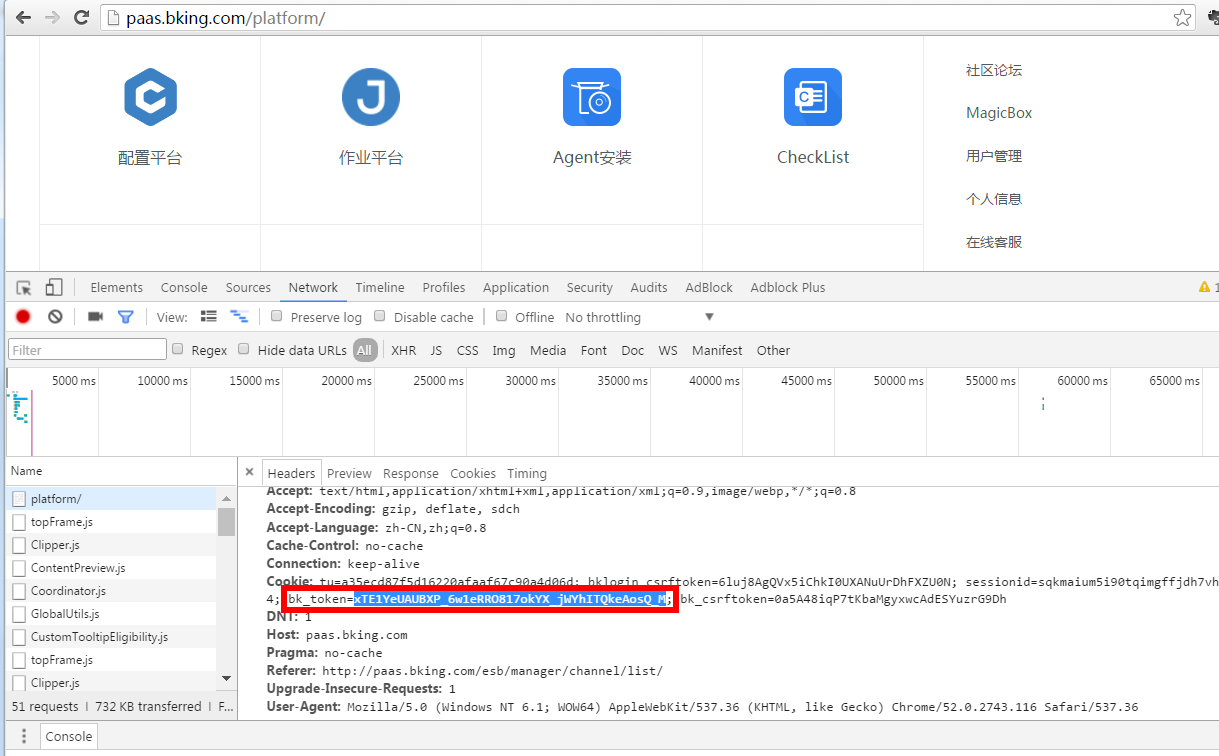
app\_secret：

bk\_token：可以在登录paas平台后通过浏览器插件获取，见后面截图

**除了bk\_token外，我对另外两个参数app\_code和app\_secret的用途不是很清楚，因为我认为自定义组件和之前部署的hello-world应用没有任何关系**



bk\_token的获取：



从前面的结果可以看到，自定义组件的手动接入，已经可以正常访问了，接下来是组件的自助接入，演示的是**对前面手动创建的组件通道进行自助接入**

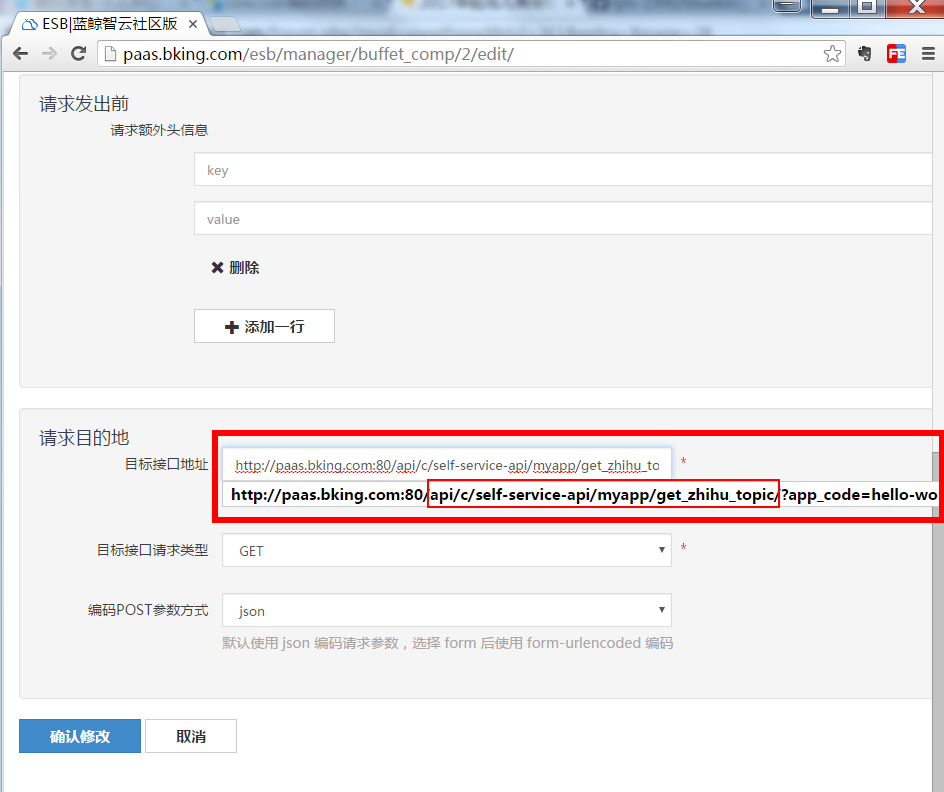
1. 组件自助接入（对手动创建的组件通道进行自助接入）
   1. 申请

(1) 组件名称：任意

(2) 组件系统：选择自定义组件myapp所在的系统名

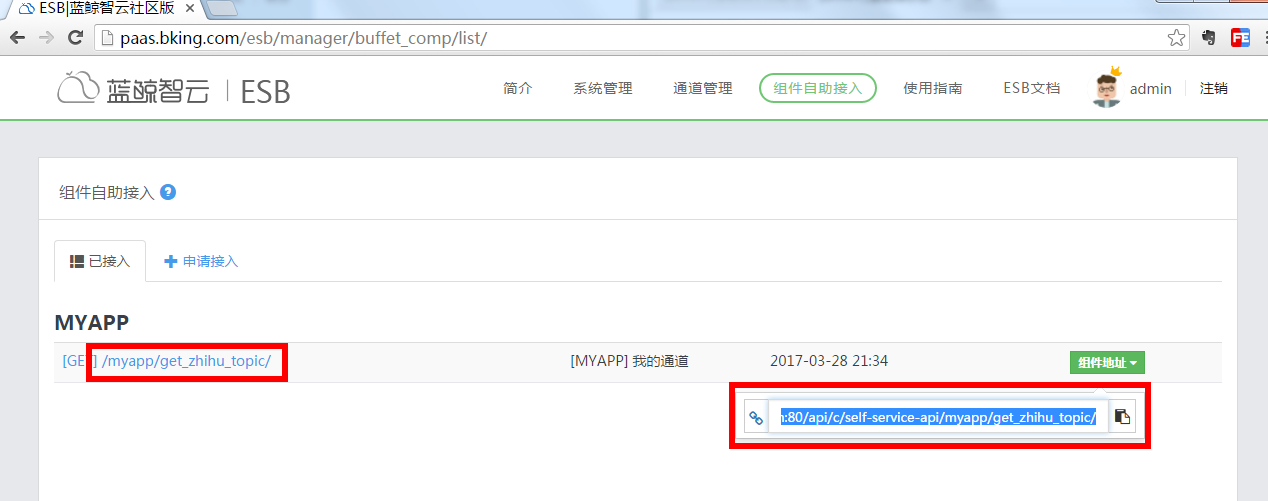


(3) 继续往下滚动，目标接口地址填前面手动接入组件产生的地址



* 1. 接口API

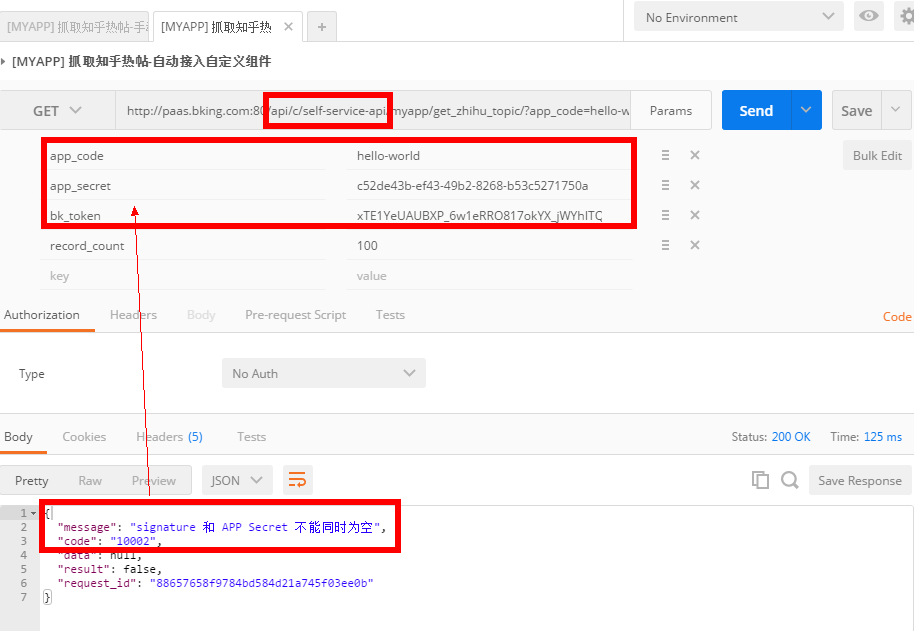
自助接入通道产生的API接口路径以**/api/c/self-service-api**开头：



* 1. 访问测试

接着尝试访问自动接入的API，提示signature和app secret不能同时为空：

（因为对ESB了解不够，**猜测可能是由于handle方法里未对request 中的app\_code和app\_secret处理导致**）



1. 代码
2. 参见: <https://github.com/tjm-1990/blueking-esb>
3. 问题
4. 在对手动接入的自定义组件通道进行访问测试时有3个必填参数，除了bk\_token外，对另外两个参数app\_code和app\_secret的用途不是很清楚，因为**我认为我没有调用内部组件，而自定义组件和之前部署的hello-world应用没有任何关系**（[参见4.3的批注](#手动接入自定义组件通道的访问测试)）
5. 自助接入组件使用的是自定义组件的通道，访问失败，猜测**可能是由于handle方法里未对request 中的app\_code和app\_secret处理导致**（[参见5.3的批注](#自助接入自定义组件通道的访问测试)）