有生云

数字底座接口安全控制

产品版本：V1.0

文档版本：202406

北京有生博大软件股份有限公司

法律声明

有生云提醒您在阅读或使用本文档之前仔细阅读、充分理解本法律声明各条款的内容。如果您阅读或使用本文档，您的阅读或使用行为将被视为对本声明全部内容的认可。

1、您应当通过有生云网站或有生云提供的其他授权通道下载、获取本文档，且仅能用于自身的合法合规的业务活动。本文档的内容视为有生云的保密信息，您应当严格遵守保密义务；未经有生云事先书面同意，您不得向任何第三方披露本手册内容或提供给任何第三方使用。

2、经有生云事先书面许可，任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。

3、由于产品版本升级、调整或其他原因，本文档内容有可能变更。有生云保留在没有任何通知或者提示下对本文档的内容进行修改的权利，并在有生云授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过有生云授权渠道下载、获取最新版的用户文档。

4、本文档仅作为用户使用有生云产品及服务的参考性指引，有生云以产品及服务的“现状”“有缺陷”和“当前功能”的状态提供本文档。有生云在现有技术的基础上尽最大努力提供相应的介绍及操作指引，但有生云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的，有生云不承担任何法律责任。在任何情况下，有生云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或刑罚性的损害，包括用户使用或信赖本文档而遭受的利润损失，承担责任（即使有生云已被告知该等损失的可能性）。

5、有生云文档中所有内容，包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计，均由有生云和/或其关联公司依法拥有其知识产权，包括但不限于商标权、专利权、著作权、商业秘密等。非经有生云和/或其关联公司书面同意，任何人不得擅自使用、修改、复制、公开传播、

改变、散布、发行或公开发表有生云网站、产品程序或内容。此外，未经有生云事先书面同意，任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制有生云的名称（包括但不限于单独为或以组合形式包含“有生云”“youshengyun”“risesoft”“risenet”“有生云数字底座”“Y9-DI”等有生云和/或其关联公司品牌，上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别有生云和/或其关联公司）。

6、如若发现本文档存在任何错误，请与有生云取得直接联系。

目录

1.接口安全控制

risenet-y9boot-support-api-access-control 组件加入了对接口安全的控制，下面介绍组件的使用

1.1.黑名单

对于加入黑名单的客户端访问接口会被拒绝

输入的名单可以是多个，用英文逗号分割，支持具体 IP、IP 网段格式。示例：

- 192.168.1.1
- 192.168.1.0/24
- 192.168.1.1-100
- 192.168.1.*

1.2.白名单

只有在白名单内的客户端才允许访问接口，如果此过滤项开启时，而此时没有添加允许的记录，默认允许所有请求通过

白名单的配置格式可参考黑名单的配置格式

1.3.接口签名

接口签名验证，如果此过滤项开启则对应的请求必须有签名，且客户端的签名需跟服务端计算的签名一致放可放行。

接口签名机制可以识别请求用户的身份，防止请求的篡改，以及防止一定程度的重放攻击

需要在数字底座中申请 API 密钥，方便后续生成接口请求的签名，请求使用 UTF-8 编码

签名计算伪代码：

String stringToSign = appId + path + sortedQueryString + body + timestamp;

String sign = HMAC-SHA256 (app_secret, stringToSign);

接下来以数字底座中的一个接口为例说明

GET http://localhost:7055/platform/services/rest/v1/organization/get?tenantId=11111111-1111-1111-1111-111111111113&

organizationId=11111111-1111-1111-1111-111111111115

参与签名的各个参数

参数名	描述	参考值
app_secret	数字底座生成的 API 密钥中的 appSecret	bfyu9pJxm0M2KfhblOG9QNTns17Vz3yz3v5Jbq
appId	数字底座生成的 API 密钥中的 appId	1732477113216737280
path	请求的 URL 的部分，移除协议、域名及请求参数部分	/platform/services/rest/v1/person/get

sortedQueryString	按 key 名称以字典顺序排序后的查询参数字符串	organizationId=1666895850885423104&tenantId=11111111-1111-1111-1111-111111111113
body	原始的请求 body，如果为空则用 "" 空字符串代替	
timestamp	请求的秒级时间戳	1734329686

参数签名示例

```
String stringToSign = "1732477113216737280"
        + "/platform/services/rest/v1/person/get"
        + "organizationId=1666895850885423104&tenantId=11111111-1111-1111-1111-111111111113"
        + ""
        + "1734329686";
String sign = HMAC-SHA256 ("bfyu9pJxm0M2KfhblOG9QNTns17Vz3yz3v5Jbq", stringToSign);
```

需传递的请求头

请求头	描述	参考值
x-app-id	数字底座生成的密钥对中的 appId	1732477113216737280
x-timestamp	秒级的时间戳，可用	1734329686

	于防止一定程度的重放攻击	
x-signature	客户端计算的签名值，十六进制的大写字符串	5B3C73712158048EA34837B7BFA204AACA2E669ADDA94998A6E532E48AD685FD

最终请求

GET http://localhost:7055/platform/services/rest/v1/organization/get?organizationId=16668958508434800648

tenantId=11111111-1111-1111-1111-111111111113

x-app-id: 1732477113216737280

x-timestamp: 1734329686

x-signature: 5B3C73712158048EA34837B7BFA204AACA2E669ADDA94998A6E532E48AD685FD

1.4.组件使用

工程中需要添加接口安全控制的特性，需引入依赖包，添加配置方可使其生效

1.4.1.Maven pom.xml

添加依赖包

<dependency>

```
    <groupId>net.risesoft</groupId>
    <artifactId>risenet-y9boot-support-api-access-control</artifactId>
    <version>[最新版本]</version>
</dependency>
```

如果需要使用 snapshot 版本，还需添加私服仓库方能下载

```
<repositories>
    <repository>
        <id>nexus</id>
        <name>local private nexus</name>
        <url>https://svn.youshengyun.com:9900/nexus/repository/maven-publi
c/</url>
    </repository>
</repositories>
<pluginRepositories>
<pluginRepository>
    <id>nexus</id>
    <name>local private nexus</name>
    <url>https://svn.youshengyun.com:9900/nexus/repository/maven-public/</
url>
```

```
</pluginRepository>
</pluginRepositories>
```

1.4.2.属性文件 application.yml

可配置的属性可参考

net.risesoft.y9.configuration.feature.apiacl.Y9ApiAccessControlProperties

```
y9:
    feature:
        api-access-control:
            url-patterns: '/services/rest/*'
            black-list:
                enabled: true
            white-list:
                enabled: true
            sign:
                enabled: true
```

1.4.3.处理逻辑

安全控制基于 Servlet Filter 做过滤，需要考虑过滤器之间的优先级

默认优先级黑名单、白名单再到密钥签名，如优先级较高的不通过则不再校验，直接返回失败。当然，可以通过修改配置项调整优先级