

# 数据库

创建数据库

```
1 CREATE DATABASE searchable_crypt;
```

## 用户信息

id	user_id	mobile	en_key	password
bigint,PRI	var_char(250),UNI	varchar(250)	varchar(250)	varchar(250)

```
1 CREATE TABLE user_info
2 (
3 id BIGINT NOT NULL AUTO_INCREMENT PRIMARY KEY,
4 user_id VARCHAR(250) UNIQUE,
5 mobile VARCHAR(250),
6 password varchar(250)
7 );
```

## 文件信息

id	user_id	path	file_name	i1	i2
bigint,PRI	var_char(250)	var_char(250)	var_char(250)	longblob	longblob

```
1 CREATE TABLE file_info
2 (
3 id BIGINT NOT NULL AUTO_INCREMENT PRIMARY KEY,
4 user_id VARCHAR(250),
5 path VARCHAR(250),
6 file_name varchar(250),
7 i1 longblob,
8 i2 longblob
9 );
```

# 请求

## signup

url:<http://123.56.185.106:8080/signup>

参数名	参数意义
-----	------

参数名	参数意义
uid	用户名, string
mobile	手机号, string
password	密码, string, hash保护

POST

http://123.56.185.106:8080/signup

发送

H 请求头

参数

x-www-form-urlencoded;charset=UTF-8

自定义格式

Key	Value
uid	muling
password	asdsadhwqiewoqe
mobile	18512379999

响应体

H 响应头

Cookie

原始数据

格式化

JSON:

全部展开

展开一级

展开二级

展开三级

展开四级

自动换行

帮助文档

Hello, muling

## login

url:<http://123.56.185.106:8080/signup>

参数名	参数意义
uid	用户名, string
password	密码, string, hash保护

返回token

POST

http://123.56.185.106:8080/login

发送

保存

H 请求头

参数

x-www-form-urlencoded;charset=UTF-8

自定义格式

Key	Value	批量编辑
uid	muling	言
password	asdsadhwqiewoqe	言

响应体

H 响应头

Cookie

原始数据

格式化

JSON:

全部展开

展开一级

展开二级

展开三级

展开四级

自动换行

帮助文档

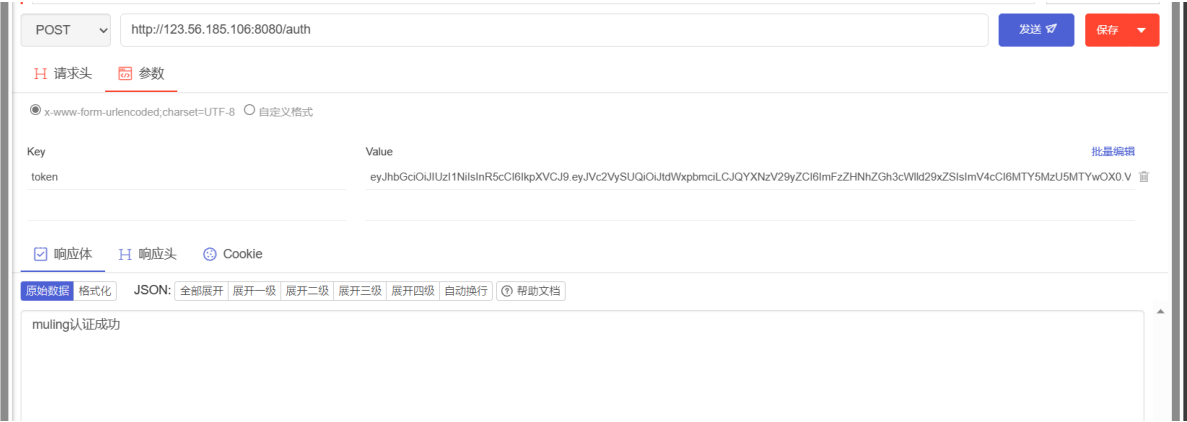
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VySUQiOiJldWxpbmciLCJQYXNzV29yZCI6ImFzZHNhZGh3cWlld29xZSIsImV4cCI6MTY5MzU5MTYwOX0.VEM9ZgQOyVPsDbOyQ1gCzy0SOpxJlQeFsy\_r5GOR9mc

## auth

url:<http://123.56.185.106:8080/auth>

参数名	参数意义
-----	------

参数名	参数意义
token	string,用于验证权限，访问文件夹时需要使用，有有效时间，需要定时申请（2小时），该接口仅用于测试认证功能。



# upload

url:<http://123.56.185.106:8080/upload>

报文修改头部:

//Content-Type:multipart/form-data

键	值
Content-Type	multipart/form-data

参数名	参数意义
file	上传的文件密文
token	用户认证token,string
path	用户需要存储文件的路径，实际路径为uid/path/filename
index	构建的索引，

index构建方法：

```
Index := index.BuildIndex(keyWord, key)
```

```
byteIndex:=tool.Encode(indexAndID[i].Index)
```

```
indexstring := hex.EncodeToString(byteIndex)
```

传输该index即可

POST127.0.0.1:8080/upload

发送

参数

授权

Header (10)

Body

预请求脚本

测试

设置

Cookie

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

	键	值	描述	...	批量修改
<input checked="" type="checkbox"/>	file	2.txt			
<input checked="" type="checkbox"/>	token	eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.e...			
<input checked="" type="checkbox"/>	path	myfiles			
<input checked="" type="checkbox"/>	index	14ff8102010105496e64657801ff820001ff...			
<input type="checkbox"/>	uid	muling			

Body

Cookie

Header (3)

测试结果

200 OK32 ms142 B保存响应

美化

原

预览

可视化

Text

1 上传文件 2.txt 成功

## search

url:<http://123.56.185.106:8080/search>

参数

参数名	参数意义
token	用户认证token,string
path	用户存储文件的路径
trapdoor	陷门参数
exceptnum	期望的结果数，结果不足则输出所有结果

POST127.0.0.1:8080/search

发送

参数

授权

Header (10)

Body

预请求脚本

测试

设置

Cookie

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

	键	值	描述
<input type="checkbox"/>	password	wqepjsakdjasd	
<input checked="" type="checkbox"/>	token	eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eJYVc2VySUQiOi...	
<input checked="" type="checkbox"/>	path	myfiles	
<input checked="" type="checkbox"/>	trapdoor	0dff85020102ff860001ff800000097f060102ff84000000...	
<input checked="" type="checkbox"/>	exceptnum	10	

Body

Cookie

Header (3)

测试结果

状态: 200 OK 时间: 21 ms 大小: 2.45 KB 保存响应

美化

原

预览

可视化

Text

1 16ff8102010107526573756c747301ff820001ff800000257f030102ff8000010301024964010c00010553636f7265010000010444617465010c000000fe046aff82000001013401f857bb948caa0604001ffd1687474703a2f2f3132332e35362e3138352e3130363a383038302f646f776e6c6f61643f746f6b656e3d65794a68624763694f694a49557a49314e694973496e523563434936496b705856434a392e65794a515958526f496a6f69625856736157356e5846787465575a706247567a584677694c434a476157786c546d46745a534936496a557564486830496977695a586877496a6f784e6a6b7a4e6a59314d54593266512e4471566c73544d42744a4d68385472377057394236705761626166336e736567634d6c66416a5f424e33770001013001f9e1c68b0fa4f3bf01ffd1687474703a2f2f3132332e35362e3138352e3130363a383038302f646f776e6c6f61643f746f6b656e3d65794a68624763694f694a49557a49314e694973496e523563434936496b705856434a392e65794a515958526f496a6f69625856736157356e5846787465575a706247567a584677694c434a476157786c546d46745a534936496a497564486830496977695a586877496a6f784e6a6b7a4e6a59314d54593266512e77434d68336d4c2d6548465765645673796a4d2d4c7a465855392d7665764d396274477253466263c71340001013101f9e1c68b0fa4f3bf01ffd1687474703a2f2f3132332e35362e3138352e3130363a383038302f646f776e6c6f61643f746f6b656e3d65794a68624763694f694a49557a49314e694973496e523563434936496b705856434a392e65794a515958526f496a6f69625856736157356e5846787465575a706247567a584677694c434a476157786c546d46745a534936496a457564486830496977695a586877496a6f784e6a6b7a4e6a59314d54593266512e4131474262304c5833566358613457367076596b526279963053624d5932566646423279597a5f494562380001013201f9e1c68b0fa4f3b

trapdoor生成:

```
1 T := trapdoor.GenTrapdoor(strs, key)
2 byteTrapdoor, err := utils.Encode(T)
3 trapdoorString :=hex.EncodeToString(byteTrapdoor)
```

记得处理err

结果解码函数:

```
1 resultBytes2, err := hex.DecodeString(resultStr)
2 var results2 search.Results
3 err = tool.Decode(resultBytes2, &results2)
4 fmt.Println(results2[0])
```

得到的结果为Result结构数组，Result结构体如下:

```
type Result struct { 2 个用法  march-791
    Id      string
    Score   float64
    Date    string
}

type Results []*Result 4 个用法  march-791
```

ID没有用，Date为下载url，Score为相似度