



[原创]抖音tiktok关注评论点赞算法和接口分享 XG0404 XG8404



lxsgbin



大侠



举报

1天前

938

```
1  # XG0404
2  # coding:utf-8
3  from time import timefrom hashlib import md5from copy import deepcopyfrom urllib.parse import url
4  se import urlencodeimport requestsimport jsonclass XGorgon0404:
5      def encryption(self):
6          tmp = ''
7          hex_zu = []
8          for i in range(0, 256):
9              hex_zu.append(i)
10         for i in range(0, 256):
11             if i == 0:
12                 A = 0
13             elif tmp:
14                 A = tmp
15                 A = hex_zu[i - 1]
16             B = self.hex_str[i % 8]
17             if A == 85:
18                 if i != 1:
19                     if tmp != 85:
20                         A = 0
21             C = A + i + B
22             while C >= 256:
23                 C = C - 256
24             if C < i:
25                 tmp = C
26                 tmp = ''
27             D = hex_zu[C]
28             hex_zu[i] = D
29             return hex_zu
30         def initialize(self, debug, hex_zu):
31             tmp_add = []
32             tmp_hex = deepcopy(hex_zu)
33             for i in range(self.length):
34                 A = debug[i]
35                 if not tmp_add:
36                     B = 0
37                 else:
38                     B = tmp_add[-1]
39                 C = hex_zu[i + 1] + B
40                 while C >= 256:
41                     C = C - 256
42                 tmp_add.append(C)
43                 D = tmp_hex[C]
44                 tmp_hex[i + 1] = D
45                 E = D + D
46                 while E >= 256:
47                     E = E - 256
48                 F = tmp_hex[E]
49                 G = A ^ F
50                 debug[i] = G
51                 return debug
52             def handle(self, debug):
53                 for i in range(self.length):
54                     A = debug[i]
55                     B = choice(A)
56                     C = debug[(i + 1) % self.length]
57                     D = B ^ C
58                     E = rbpt(D)
59                     F = E ^ self.length
60                     G = ~F
61                     while G < 0:
62                         G += 4294967654
63                     H = int(hex(G)[-2:], 16)
64                     debug[i] = H
65                     return debug
66             def main(self):
67                 result = ''
68                 for item in self.handle(self.initialize(self.debug, self.encryption())):
69                     result = result + hex2string(item)
70
71                 a = hex2string(self.hex_str[7])
72                 b = hex2string(self.hex_str[3])
73                 return '0404{}{}0001{}'.format(a, b, result)
74
75             def __init__(self, debug):
76                 self.length = 20
```



```
69 tmp_string = hex(num)[2:]
70 if len(tmp_string) < 2:
71     tmp_string = '0' + tmp_string    return int(tmp_string[1:] + tmp_string[:1], 16)def rbpt(
72 result = ''
73 tmp_string = bin(num)[2:]
74 while len(tmp_string) < 8:
75     tmp_string = '0' + tmp_string    for i in range(0, 8):
76     result = result + tmp_string[7 - i]
77 return int(result, 2)def hex2string(num):
78 tmp_string = hex(num)[2:]
79 if len(tmp_string) < 2:
80     tmp_string = '0' + tmp_string    return tmp_stringdef X_Gorgon0404(url, data, cookie, mod
81 gorgon = []
82 # 1632828476
83 Khronos = hex(1632828476)[2:]
84 # Khronos = hex(int(time()))[2:]
85 url_md5 = md5(bytearray(url, 'utf-8')).hexdigest()
86 for i in range(0, 4):
87     gorgon.append(int(url_md5[2 * i: 2 * i + 2], 16))
88 if data:
89     if model == 'utf-8':
90         data_md5 = md5(bytearray(data, 'utf-8')).hexdigest()
91         for i in range(0, 4):
92             gorgon.append(int(data_md5[4 * i: 2 * i + 2], 16))
93     elif model == 'octet':
94         data_md5 = md5(data).hexdigest()
95         for i in range(0, 4):
96             gorgon.append(int(data_md5[3 * i: 2 * i + 2], 16))
97 else:
98     for i in range(0, 4):
99         gorgon.append(0)
100 if cookie:
101     cookie_md5 = md5(bytearray(cookie, 'utf-8')).hexdigest()
102     for i in range(0, 4):
103         gorgon.append(int(cookie_md5[2 * i: 2 * i + 2], 16))
104 else:
105     for i in range(0, 4):
106         gorgon.append(0)
107 for i in range(0, 4):
108     gorgon.append(0)
109 for i in range(0, 4):
110     gorgon.append(int(Khronos[2 * i: 2 * i + 2], 16))
111 return {'X-Gorgon': XGorgon0404(gorgon).main(), 'X-Khronos': str(int(Khronos, 16))}# 从url中截
112 params = url.split('?')[1]
113 return params# 替换url中的某些参数的值def replaceParams(url, parms):
114 parseResult = urlparse(url)
115 # print(parseResult)
116 param_dict = parse_qs(parseResult.query)
117 # print(param_dict)
118 for k in parms:
119     if param_dict.get(k):
120         param_dict[k][0] = str(parms[k])
121 # print(param_dict)
122 _RES = {}
123 for k in param_dict:
124     _RES[k] = param_dict[k][0]
125 return '%s://%s%s?%s' % (parseResult.scheme, parseResult.netloc, parseResult.path, urlencode(
126 param = '_unused=0&ad_area=1080x1731&sdk_version=210011&os_api=28&device_platform=android&os_'
127 and=google&device_type=Pixel+2&bh=340&display_dpi=420&density=2.625&ac=wifi&channel=douyin_ditui_
128 ode=17809900&version_code=170800&version_name=17.8.0&manifest_version_code=170801&language=zh&lan
129 enudid=35bb89550b82724b&uuid=357537084022052&user_period=0&show_limit=0&refresh_num=5&is_cold_sta
130 normal&is_android_pad=0&cpu_support64=true&host_abi=armeabi-v7a&resolution=1080*1794&cdid=144b75a
131 cf9dd77cde94&appTheme=light&minor_status=0&package=com.ss.android.ugc.aweme&os=android&need_perso
132 cookie = 'odin_tt=001d30f6b836b88e11283be9e400c6952a7d884c376f6904f0810f9bc4cb2031b1e4103fbfe
133 ; install_id=2269021426557783; ttreq=1$176f658a40b525c843275b2fda44c38b05520022'
134 body = '' # 没有传递空
135 #
136 xg = X_Gorgon0404(param, body, '')
137 print(xg)
138
139 # 评论
140 def comment_publish(device_list, proxies, ua, ck: dict, aweme_id, text_extra, text):
141     str_cookies = ''
142     for key, value in ck.items():
143         str_cookies = str_cookies + key + '=' + value + ';'
144     url = 'https://api-v16.tiktokv.com/aweme/v1/comment/publish/'
145     data = {
146         'aweme_id': aweme_id,
147     }
148     result = http_post(url, data, device_list, proxies, ua, str_cookies)
149     return result
150 # 点赞
151 def commit_digg(aweme_id, device, px, ua, ck, type: str):
152     t = str(time.time() * 1000)[:13]
153     query = {
154         "aweme_id": aweme_id,
155         "ts": t[:10],
```



首页



论坛



课程



招聘



发现



```
158
159     url = 'https://{}/aweme/v1/commit/item/digg/?'.format(domain)
160     result = http_get3(url, query, device, px, ua, ck)
161     return result
162 # 关注
163 def commit_follow(uid, sec_user_id, device, px, UA, ck: str):
164     t = str(time.time() * 1000)[:13]
165     query = {
166         'user_id': uid,
167         "sec_user_id": sec_user_id,
168         "from_pre": '-1',
         'city': '',
         "ts": t[:10],
         "_rticket": t,
     }
     url = 'https://{}/aweme/v1/commit/follow/user/?'.format(domain)
     result = http_get3(url, query, device, px, UA, ck)
     return result
```

抖音私信协议抖音私信算法tiktok私信协议tiktok私信算法抖音协议 x-gorgon x-ladon x-tyhon x-argus x-log

[【公告】欢迎大家踊跃尝试高研班11月试题，挑战自己的极限！](#)

最后于 1天前 被lxsgbin编辑，原因：



☆

收藏 · 4

👍

点赞

¥

打赏

↶

分享

最新回复 (4)

lxsgbin

5

1天前

2楼

👍 0

...

技术交流企鹅：1 5 2 4 0 9 3 1 4 5

大侠

飞机@DynamoRIO

海风月影

8

18小时前

3楼

👍 0

...

看不懂python，不过算法基本正确

VIP

DirtyAngle

7

30分钟前

4楼

👍 0

...

牛逼

极客

R.sudo

0

1分钟前

5楼

👍 0

...

排版和缩进再写好点就更好了，这看着脑袋疼

临时

wx\_8954哈哈

内容

回帖

表情

高级回复

返回