بسم الله الرحمن الرحیم

راهنمای developer ماژول Hash

مرداد 97

فصل 1

1-1 نگارش

در جدول زیر تاریخچه نگارش های مختلف ماژول آورده شده است.

|  |  |  |
| --- | --- | --- |
| تاریخ | نسخه | توضیحات |
| تیر 97 | 1.0 | نسخه اولیه document آماده شد. |

۱-۲ توضیحات اولیه

این ماژول یک عبارت Hash دریافت می‌کنو و در خروجی پسورد حل شده‌ی آن را می‌دهد. حل Hash به سایت CMD5 برون سپاری می‌شود و خروجی از آن دریافت می‌گردد. توضیحات بیشتر در مورد متود های این ماژول در زیر آورده شده است. کد مربوط به این ماژول 113 می باشد. [جدول ماژول ها.](refrence_links/modules_code.pdf)

۱-۳ متد hash\_deocde\_by\_hrobot()

این متد یک object از hrobot می‌سازد و به سایت cmd5.org می‌رود و hash داده شده به ماژول را برای حل به cmd5 پاس می‌دهد. اما سایت cmd5 معمولا با نشان دادن capcha جلوی فعالیت‌های robotها را می‌گیرد. در صورت رخداد captcha، hrobot یک screen از صفحه می‌گیرد و به سرویس 2captcha برای حل کردن آن داده می‌شود و نتیجه آن را به cmd5 بر می‌گرداند. در جواب حل hash، سه حالت رخ می‌دهد: 1- جواب کامل برمی‌گردد. 2- ارور “payment” می‌دهد که نشان می‌د‌‌هد برای حل آن باید پرداختی صورت گیرد. 3- cmd5 نمی‌تواند hash را حل کند.

|  |  |
| --- | --- |
| ورودی | توضیحات |
| Method\_id | شماره تابع. مقدار این پارامتر برای این متد برابر 1 می باشد |
| Hash\_code | عبارت hash مورد نظر |
| username | Email ثبت نام شده در cmd5 |
| password | Password ثبت نام شده در cmd5 |
| Hash\_type | این مقدار نوع hash را مشخص می‌کند و در صورتی که ارسال نشود برابر md5 قرار می‌گیرد. |

جدول مربوط به ورودی متد hash\_deocde\_by\_hrobot()

|  |  |  |
| --- | --- | --- |
| Output\_parameters | method | |
| {  "results": [ {  "data": "decoded hash",  "type": 7  }  ]  "properties": [ {}  ]  } | | hash\_deocde\_by\_hrobot() |

جدول مربوط به پارامترهای خروجی متد hash\_deocde\_by\_hrobot()

نمونه خروجی کامل این ماژول در این [لینک](refrence_links/output_sample%20getipinfo.txt) موجود است.

۱-4 متد hash\_decode\_by\_api

این متد سه آرگومان hash و api\_key و email را به عنوان ورودی دریافت می کند و به api مربوط به cmd5 تحویل می‌دهد و در خروجی جواب حل شده را می‌گرد.

|  |  |
| --- | --- |
| ورودی | توضیحات |
| Method\_id | شماره تابع. مقدار این پارامتر برای این متد برابر 2 می باشد |
| hash | hash مورد نظر جهت حل |
| api\_key | جهت اتصال به سرویس مورد نظر نیاز به api\_key آن می باشد. |
| email | Email ثبت نام شده در cmd5 |

جدول مربوط به ورودی متد hash\_decode\_by\_api

|  |  |  |
| --- | --- | --- |
| Output\_parameters | | method |
| {  "results": [ {  "data": "decoded hash",  "type": 7  }  ]  "properties": [ {}  ]  } | hash\_decode\_by\_api | |

جدول مربوط به پارامترهای خروجی متد hash\_decode\_by\_api

نمونه خروجی کامل این ماژول در این [لینک](refrence_links/output_sample%20ipisproxy.txt) موجود است.

1-5 محتوای موجود در درخواست get

برای استفاده ازاین ماژول می بایست یک درخواست از نوع get جهت دریافت اطلاعات به api مورد نظر به صورت زیر ارسال شود. بدنه درخواست ارسالی شامل پارامترهای dataو type است. پارامتر type مشخص کننده نوع ورودی ماژول(unstructue,file,domain,path,…) است که id مربوط به انواع مختلف ورودی ها و توضیحات مربوطه که در این [لینک](refrence_links/entities_list.pdf) موجود است، در این فیلد قرار می گیرد. پارامتر data هم همان داده ارسالی است که شامل پارامترهای ورودی ماژول می باشد.

**{**

**“data”:{“hash”:” asadasdasdasdw”, “API\_KEY”:” C571380781”, “email”:” C571380781” },**

**“type”:0**

**}**

نمونه body یک socket request

در تصویر نمونه بالا داریم:

Data: داده ارسالی برای استفاده در ماژول که شامل email و hash و API\_KEY می باشد.

Type: نوع ورودی ماژول را مشخص می کند که می تواند از نوع unstructured,path,file و یا انواع دیگر باشد و باید id ورودی موردنظر در این قسمت قرار گیرد.

1-5 جدول خطاها

جدول زیر شامل خطاهایی است که ماژول Hash در پاسخ به کلاینت بر می گرداند. خطا ها با شماره مشخصی نشان داده می شوند.

|  |  |  |
| --- | --- | --- |
| توضیحات | نام | کد |
| زمانی که فرمت داده ورودی اشتباه باشد. | InvalidInputError | 102 |
| زمانی که مقدار API کافی نباشد. | InsufficientAPI | 115 |
| زمانی که شبکه دچار تداخل شود. | NetworkError | 107 |