J'ai découvert quelque chose de fascinant sur Discord récemment.

Tout d'abord, j'avais créé un script Python pour envoyer des messages à grande vitesse sur Discord avec des noms d'utilisateur différents. C'était pour expérimenter sur un canal de mon serveur personnel, un espace de test. Comme vu sur l'image jointe, le résultat était impressionnant. Cependant, j'ai ensuite modifié le programme pour qu'il envoie 100 messages à la suite. C'est là que les ennuis ont commencé : Discord a rapidement détecté une activité anormale et a réagi en bloquant mon script avec l'erreur "rate limited, retry after 0.3 seconds" (le délai pouvant varier). C'était une mesure intelligente pour empêcher le spam, mise en place par les créateurs de Discord.

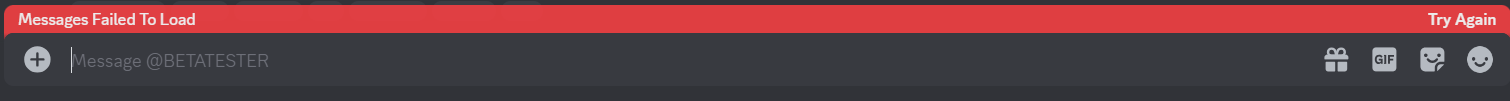
Après avoir ajusté mon code pour respecter ce délai entre chaque message, le script a réussi à fonctionner une première fois. Cependant, à mesure que les 100 messages étaient envoyés, le délai de ralentissement s'est accumulé, réduisant considérablement la vitesse d'exécution vers la fin. J'ai ensuite modifié le code pour générer des noms d'utilisateur aléatoires pour chaque message, mais cela a introduit un nouveau problème. Perdu, je suis retourné à la version précédente du code, mais le problème persistait avec une accumulation d'erreurs dans la console Python : "Rate limited. Retrying after 0.5 seconds.". Malheureusement, le script n'a pas pu reprendre l'envoi des messages toutes les 0.5 secondes comme prévu, car les délais s'étaient probablement accumulés à plusieurs minutes en raison de mon incapacité à stopper le programme immédiatement.

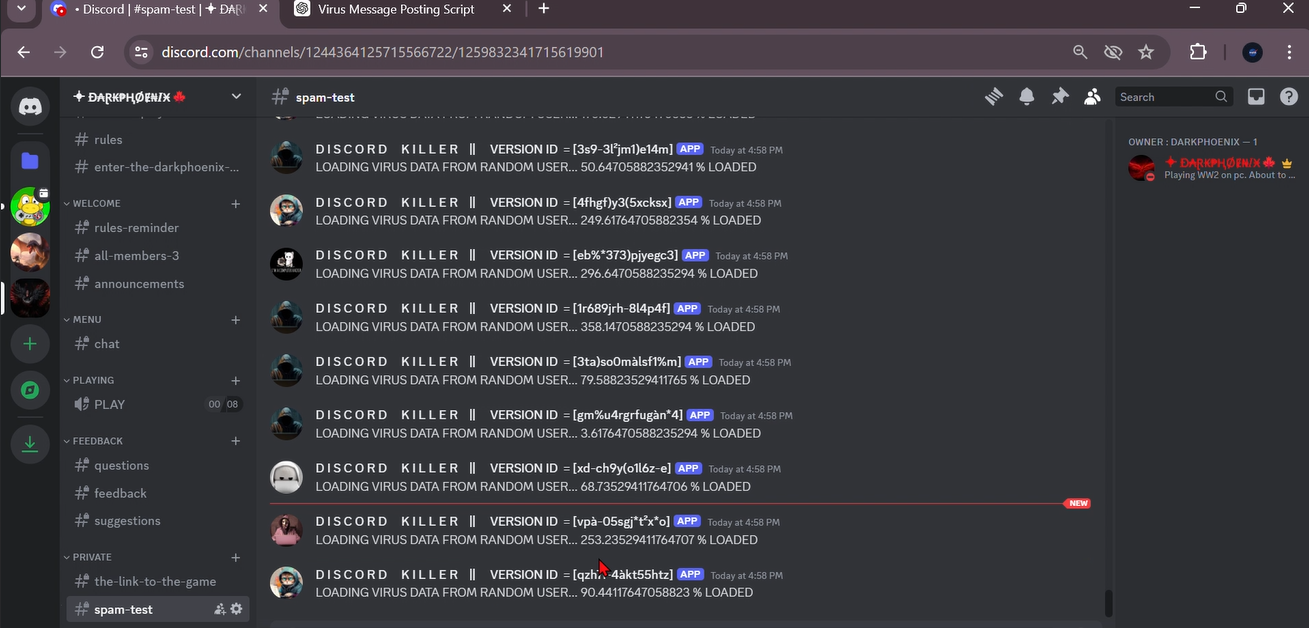
Pensant d'abord à un bug de Discord, j'ai rafraîchi la page, mais cela n'a rien changé. Les messages ne se chargeaient dans aucun des chats, et au bout de quelques minutes, j'ai réalisé que je ne pouvais plus du tout interagir sur Discord : je ne pouvais pas changer mon nom d'utilisateur, envoyer des messages, ajuster les paramètres de mes chaînes sur mon serveur, ni même voir les anciens messages. J'étais comme déconnecté du réseau, bien que tous les autres sites internet fonctionnaient normalement. Ai-je été banni par Discord, sans avertissement ni notification ? Cela semblait improbable, mais j'étais perplexe.

Après environ 15 minutes d'essais pour résoudre le problème, tout est revenu miraculeusement à la normale. J'ai relancé mon script et cette fois-ci, il a fonctionné comme prévu, envoyant les 100 messages sans problème. C'est alors que j'ai compris ce qui s'était passé : Discord avait mis en pause toutes mes activités en raison des délais accumulés et de leurs mesures de sécurité. Le fait de lancer le programme plusieurs fois a probablement contribué à déclencher cette réaction de sécurité.

Pour tester cette théorie, j'ai relancé le script une dernière fois et, comme prévu, la console affichait de nouveau les erreurs de délai d'attente. J'ai immédiatement arrêté le code pour éviter d'accumuler davantage de délais. En conclusion, mon script a temporairement bloqué mon compte Discord en raison de ces délais accumulés.

La leçon que j'en tire est de réduire le nombre de messages envoyés à la fois. Passer de 100 à 10 ou 5 messages peut éviter ces problèmes et préserver l'intégrité de ton compte Discord.





import aiohttp

import asyncio

import random

import time

class Setup:

    def \_\_init\_\_(self) -> None:

        self.chars = 'abcdefghijklmnopqrstuvwxyz134567890%ù²(-à)\*'

        self.webhook\_url = 'https://discord.com/api/webhooks/1259832964989325352/Krej5cxAF4AKxrP1GAHX4Lir46ITGZm7uVwRUD-rZR1kptKZbN6fre5Kix4yiUsQvYXz'

        self.usernames = []

        self.contents = []

        self.avatar\_urls = []

        self.generate\_usernames\_list()

        self.generate\_contents\_list()

        self.generate\_avatar\_urls\_list()

    def generate\_usernames\_list(self):

        for \_ in range(100):

            username = ''.join(random.choice(self.chars) for \_ in range(15))

            username = 'D I S C O R D     K I L L E R    ||     VERSION ID  = [' + str(username) + ']'

            self.usernames.append(username)

    def generate\_contents\_list(self):

        for i in range(100):

            content = "LOADING VIRUS DATA FROM RANDOM USER..." + " " + str(i \* (123 / 34)) + " % LOADED"  # random factor

            self.contents.append(content)

    def generate\_avatar\_urls\_list(self):

        any\_avatar\_url = "https://www.radiofrance.fr/s3/cruiser-production/2020/03/7495829e-7534-4ada-8632-7ee75844e531/870x489\_gettyimages-513088279.jpg"

        self.avatar\_urls.append(any\_avatar\_url)

        any\_avatar\_url = "https://media.teachprivacy.com/wp-content/uploads/2023/07/28143141/Hacker-Funny-AdobeStock\_574954979-scaled.jpeg"

        self.avatar\_urls.append(any\_avatar\_url)

        any\_avatar\_url = "https://as1.ftcdn.net/v2/jpg/05/80/66/52/1000\_F\_580665262\_SbO69iupamlna60JTP3QEUn2I1S9Myi3.jpg"

        self.avatar\_urls.append(any\_avatar\_url)

        any\_avatar\_url = "https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcTOSbhqsghVrKH5620pPh1wvOw7Fngy4TXi-w&s"

        self.avatar\_urls.append(any\_avatar\_url)

        any\_avatar\_url = "https://t3.ftcdn.net/jpg/05/74/95/50/360\_F\_574955028\_Y2qisbczKSUUAMkvC4eAIomVPWga22Ix.jpg"

        self.avatar\_urls.append(any\_avatar\_url)

        any\_avatar\_url = "https://nerdshizzle.com/wp-content/uploads/2021/01/im-a-hacker.jpg"

        self.avatar\_urls.append(any\_avatar\_url)

    def create\_random\_payload(self):

        payload = {

            "username": self.usernames[random.randint(0, len(self.usernames) - 1)],

            "content": self.contents[random.randint(0, len(self.contents) - 1)],

            "avatar\_url": self.avatar\_urls[random.randint(0, len(self.avatar\_urls) - 1)]

        }

        return payload, self.webhook\_url

class Sender:

    def \_\_init\_\_(self, webhook\_url, payload):

        self.webhook\_url = webhook\_url

        self.payload = payload

    async def send\_webhook\_message(self):

        async with aiohttp.ClientSession() as session:

            while True:

                async with session.post(self.webhook\_url, json=self.payload) as response:

                    if response.status == 204:

                        print(f"MESSAGE = {self.payload['content']} || SENT SUCCESSFULLY.")

                        return

                    elif response.status == 429:  # Rate limited

                        data = await response.json()

                        retry\_after = data.get("retry\_after", 0.5)

                        print(f"Rate limited. Retrying after {retry\_after} seconds.")

                        await asyncio.sleep(retry\_after)

                    else:

                        print(f"Failed to send message. Status code: {response.status}")

                        print(await response.text())

                        return

async def main():

    keep\_data = False

    number\_of\_messages = 3

    setup = Setup()

    tasks = []

    if not keep\_data:

        for \_ in range(number\_of\_messages):

            payload, webhook = setup.create\_random\_payload()

            sender = Sender(webhook, payload)

            tasks.append(sender.send\_webhook\_message())

    else:

        payload, webhook = setup.create\_random\_payload()

        sender = Sender(webhook, payload)

        for \_ in range(number\_of\_messages):

            tasks.append(sender.send\_webhook\_message())

    await asyncio.gather(\*tasks)

if \_\_name\_\_ == "\_\_main\_\_":

    asyncio.run(main())