

Protocole

Ce fichier définit le protocole de communication de tchatator, qui est utilisé pour le chat de TripEnArvor.

Le protocole est basé sur une architecture client-serveur où le client envoie des commandes au serveur pour effectuer des actions comme l'envoi de messages, la connexion, la déconnexion, etc. Le serveur répond avec un code de statut et des données supplémentaires si nécessaire.

Les commandes, leurs paramètres et les réponses possibles

Le client envoie des commandes au serveur pour réaliser diverses actions. Les commandes sont envoyées sous forme de **chaînes formatées** et incluent des paramètres comme le jeton d'authentification, l'identifiant du destinataire, et le contenu du message.

Voici toutes les commandes disponibles :

- **LOGIN : Authentifie un utilisateur avec un jeton API.**

Paramètres :

- **api-token** : La clé d'API de l'utilisateur.

Réponse possible :

- **200/OK** : L'authentification a réussi.
- **403/DENIED** : L'authentification a échoué (utilisateur déjà connecté, jeton invalide, banni).

- **DISCONNECTED : Informe le serveur qu'un client s'est déconnecté.**

Les commandes suivantes nécessitent que l'utilisateur soit authentifié avec un jeton API valide.

- **SEND_MSG : Envoie un message à un utilisateur.**

Paramètres :

- **token** : Le jeton d'authentification de l'utilisateur.
- **recipient-id** : L'identifiant de l'utilisateur destinataire.
- **message-length** : La longueur du message.
- **content** : Le contenu du message.

Réponse possible :

- **200/OK** : Le message a été envoyé avec succès.
- **401/UNAUTH** : L'utilisateur n'est pas authentifié.

- **UPDT_MSG : Met à jour un message existant.**

Paramètres :

- **token** : Le jeton d'authentification de l'utilisateur.
- **message-id** : L'identifiant du message à mettre à jour.
- **content** : Le nouveau contenu du message.

Réponse possible :

- 200/OK : Le message a été mis à jour avec succès.
- 401/UNAUTH : L'utilisateur n'est pas authentifié.
- 403/DENIED : L'utilisateur n'a pas les permissions nécessaires pour mettre à jour le message.

- **DEL_MSG : Supprime un message.**

Paramètres :

- **token** : Le jeton d'authentification de l'utilisateur.
- **message-id** : L'identifiant du message à supprimer.

Réponse possible :

- 200/OK : Le message a été supprimé avec succès.
- 401/UNAUTH : L'utilisateur n'est pas authentifié.
- 403/DENIED : L'utilisateur n'a pas les permissions nécessaires pour supprimer le message.

- **NEW_CHG_AVAILABLE : Demande au serveur de vérifier si de nouveaux changements** (nouveau message, suppression, modification) **sont disponibles.**

Paramètres :

- **token** : Le jeton d'authentification de l'utilisateur.

Réponse possible :

- 200/OK : Il y a de nouveaux changements. Données supplémentaires :
 - * **type** : Le type de changement (**new_message**, **message_updated**, **message_deleted**).

Plus, selon le type de changement :

Si **type** est **new_message** :

- * **message** : Les informations du nouveau message (voir la table **message**).

Si **type** est **message_updated** :

- * **message** : Les informations du message mis à jour (voir la table **message**).

Si **type** est **message_deleted** :

- * **message-id** : L'identifiant du message supprimé.

Commandes réservées aux professionnels et aux administrateurs :

- **BLOCK_USER : Bloque un utilisateur.**

Paramètres :

- **token** : Le jeton d'authentification
- **user-id** : L'identifiant de l'utilisateur à bloquer.

Commandes réservées aux administrateurs :

- **BAN_USER : Bannit un utilisateur.**

Paramètres :

- **token** : Le jeton d'authentification de l'administrateur.
- **user-id** : L'identifiant de l'utilisateur à bannir.

Format des commandes

Les commandes sont envoyées au serveur sous forme de chaînes de caractères. Chaque commande commence par l'identifiant unique de la commande suivi d'un retour à la ligne, puis de paramètres au format clé-valeur (**<clé>:valeur**). Chaque paramètre est séparé par un retour à la ligne.

Voici par exemple la construction de la commande **LOGIN** :

LOGIN

api-token:b0wv0hMUifIjxAFA1CXI5mcR8pB41L697FfdXjQvs5bG...

Ou encore la commande **SEND_MSG** :

SEND_MSG

token:b0wv0hMUifIjxAFA1CXI5mcR8pB41L697FfdXjQvs5bG...

recipient-id:42

message-length:24

content:Bonjour, comment ça va ?

Les réponses

Après chaque commande envoyée, le serveur renvoie une réponse indiquant le résultat de l'opération.

La réponse comprend les informations suivantes :

- **Statut** : Un [code de statut](#)
- **Données supplémentaires** : Des informations additionnelles en fonction de la commande reçue au format clé-valeur.

Le format des réponses est similaire à celui des commandes, avec le code de statut en première ligne suivi des données supplémentaires. Dans les données

supplémentaires, il y a souvent un champ **message** qui contient des informations supplémentaires et plus de précisions.

Les codes de statut

- **200/OK** : La commande a été exécutée avec succès.
- **403/DENIED** : L'utilisateur n'a pas les permissions nécessaires pour exécuter la commande.
- **401/UNAUTH** : L'utilisateur n'est pas authentifié.
- **416/MISFMT** : La commande est mal formatée (voir le [format des commandes](#)).
- **426/TOOMRQ** : Le serveur a reçu trop de messages en peu de temps (voir la configuration).