|  |
| --- |
| 420. Technique de l'informatique |
| **Documentation du tunnel HTTP** |
| Projet de fin d'étude en informatique de gestion |

|  |
| --- |
| Clément Bricout et Nicolas Dufour  25/11/2019  C:\Users\bricl1730357\Downloads\dicj.jpg |

Table des matières

Qu'est-ce que Ngrok2

Choix de notre Ngrok2

**Gestion de la sécurité Ngrok3**

Mot de passe pour protéger le tunnel3

Autorisation de connexion avec white List3

Inscription des demandes3

Filtrage des demandes3

Limitation du nombre de connexion4

**Architecture4**

Description4

Type de contenue pris en charge4

Gestion des versions et stabilité de l'API4

Liste des tunnels et connexions5

Service Web Ngrok5

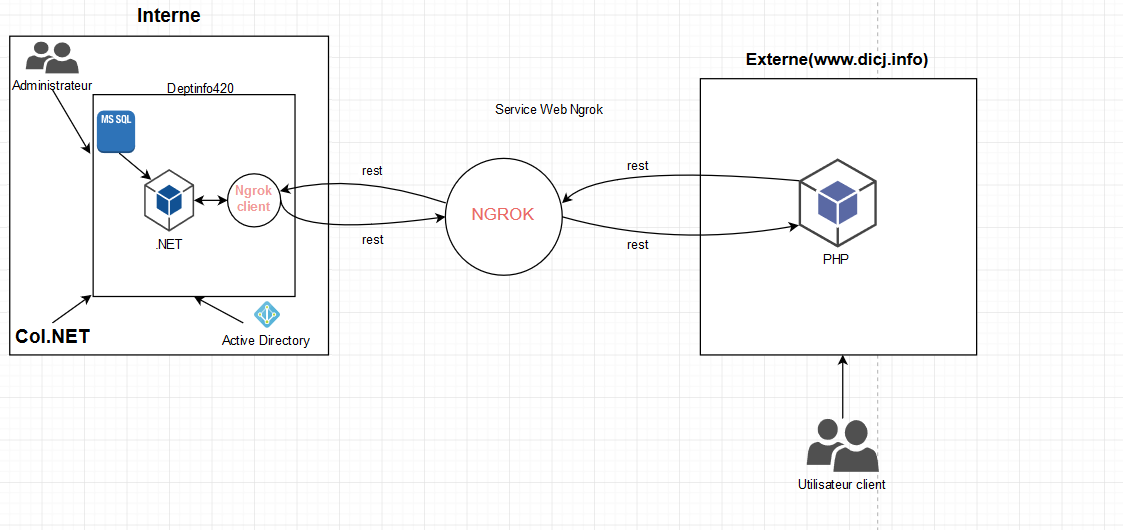
Client (service externe)5

**Qu'est-ce que Ngrok**

Ngrok est une application qui permet de créer un tunnel HTTP à partir d'un URL public. Il crée un reverse proxy sur la machine locale et un URL permettant d'accéder au local Host présent sur votre sur le serveur à partir de l'extérieur. Ce service a pour but de fournir une connexion hors de l’Intranet du cégep de Jonquière, protégé par un pare-feu, afin de pouvoir communiquer avec une application web .Net hébergé sur le serveur deptinfo420.

Pour aider au développement du projet, Ngrok a mis en place un tableau de bord qui répertorie toutes les connexions http effectuées.

Pour avoir un aperçu, nous avons réalisé un schéma technologique de notre application utilisant l'application Ngrok.



**Choix de notre Ngrok**

Pour notre projet nous avons choisi la version Ngrok à 5$ / mois.

Cette version nous donne droit à :

* 3 domaines réservés
* Une procédure Ngrok en ligne
* 60 connexions par minute
* 8 tunnels

**Gestion de la Sécurité NGROK**

**Mots de passe pour protéger le tunnel :**

Toute personne ayant en sa possession URL du tunnel peut accéder au serveur Web local. Afin de sécuriser la connexion une authentification –auth commutateur avec mots de passe est utilisée. Afin de communiquer un nom d’Utilisateur et un mot de passe est nécessaire pour chaque requête.

**Autorisation de connexion avec White List**

Une white liste permet de gérer les connexions vers le tunnel à l'adresse publique. La liste d’adresse IP public permet de filtrer les connexions.

Seul l’adresse public du serveur DICJ.INFO (198.50.100.170) est autorisée à se connecter au tunnel.

**Inspection des demandes**

Chaque demande HTTP/HTTPS est sauvegardée et peut-être accessible via une interface, afin de les consulter.

Détail du log :

* Détail de la demande
* Détail de la réponse
* Paramètre de la requête
* Charge utile de la demande
* L’adresse IP source
* L’heure
* La durée
* En-tête de la demande
* Corps de la réponse
* Octets bruts du réseau

**Filtrage des demandes :**

Toutes les demandes sont filtrées et seules ceux que l’on désire sont acceptées. Des règles de filtrage peuvent être appliquées. Dans notre cas, seul des requêtes de type REST seront utilisées.

**Limitation du nombre de connexions :**

Le nombre de requêtes par seconde peut être limité à un nombre précis.

Le nombre de connexion est limité selon l’abonnement utilisé. Dans notre cas, seule une connexion sera établie entre le service web externe sur DICJ.INFO et le tunnel.

* Gratuit : 40/minutes
* De base : 60/minutes
* Pro : 120/minutes

**Architecture**

**Description**

Le client Ngrok expose une API REST qui accorde un accès avec un programme à:

* Collecter des informations sur le statut et les métriques
* Recueillir et rejouer les demandes capturées
* Démarrer et arrêter les tunnels de manière dynamique

L’application client Ngrok.cli installé sur le serveur dptinfo420, permet l’ouverture du tunnel vers l’adresse web public fournie par Ngrok.

API client Ngrok est exposée dans le cadre de l'interface d'inspection Web locale de Ngrok. Comme il est servi sur une interface locale, l'API n'a aucune authentification. L'URL de base changera si l’on remplace l’adresse web dans les fichiers de configuration.

**Types de contenu pris en charge**

Les paramètres de demande doivent être codés dans l'API à l'aide de « application/json ». Assurez-vous que votre client définit l'en-tête « Content-Type » de la demande de manière appropriée. Toutes les réponses renvoyées par l'API sont « application/json » .

**Gestion des versions et stabilité de l’API**

L'API client Ngrok garantit que les modifications les plus importantes de l'API ne seront jamais effectuées à moins que l'appelant n'accepte explicitement une nouvelle version. Le mécanisme par lequel un appelant opte pour une nouvelle version de l'API sera déterminé à l'avenir lorsque cela deviendra nécessaire. Voici des exemples de modifications ininterrompues de l'API qui ne seront pas acceptés.

* L'ajout de nouvelles ressources
* L'ajout de nouvelles méthodes aux ressources existantes
* L'ajout de nouveaux champs sur les représentations de ressources existant
* Corrections de bugs qui modifient l'API pour correspondre au comportement documenté

**Liste des tunnels et connections**

L’application possède une interface client web de contrôle afin de voir toutes les connexions, ainsi que les requêtes effectuées, afin de les administrer.

**Service Web Ngrok**

Service web permettant d’ouvrir un tunnel entre un local host ou un host en réseau fermé et un web service, afin de rediriger des requêtes via une adresse publique. En utilisant les protocoles TLS, http et https. Dans notre cas, le protocole https sera utilisé.

Avec la version payante, un nom de domaine personnalisé nous est fourni, afin de nous connecter au domaine.

**Client (Service Externe)**

Le client est le serveur externe hébergé chez un hébergeur DICJ.INFO qui communiquera avec le serveur web à l’interne (deptinfo420) via le tunnel Ngrok, avec des requêtes REST.