

### Réalisation d'un proxy pour le Web (protocole HTTP)

Un serveur « Proxy » est un processus « mandataire », qui va servir d'**intermédiaire** pour la communication entre le client Web (le navigateur) et le serveur Web.

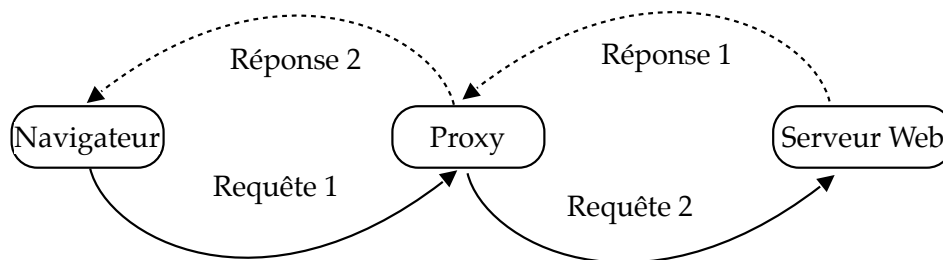
Cette communication est organisée suivant le protocole HTTP, basé TCP.

Le rôle du proxy est de :

- ▷ **filtrer** les communications en interdisant l'accès à des sites en fonctions de leur contenu (suivant une liste de mots-clé) ;
- ▷ servir de « *cache* », c-à-d de **mémoriser** le contenu des pages le plus souvent visitées et éviter ainsi de devoir les récupérer de nouveau auprès du serveur qui les hébergent.

*Le but est de réduire le trafic de données en direction de l'extérieur (le proxy étant en général dans le réseau local).*

- ▷ **modifier** les données transmises (réalisation de censure en vue d'une protection parentale par exemple) ;
- ▷ assurer une **meilleure sécurité**. Les machines clientes ne dialoguent pas directement avec les serveurs situés à l'extérieur du réseau local, mais uniquement avec le proxy (on peut également interdire toute transaction directe, sans passer par le proxy, d'un client vers un serveur extérieur à l'aide d'un *firewall*).



### ■ ■ ■ Fonctionnement du proxy

Le **navigateur** doit d'abord être configuré pour utiliser un proxy pour le protocole (ou service) HTTP :

- ▷ le proxy doit se comporter comme un **serveur** :
  - il attendra des connexions du navigateur ;
  - il doit disposer d'un SAP (adresse IP, n° de port) ;
  - ces informations sont à renseigner au niveau des préférences du navigateur.

Le proxy :

- ▷ crée une « *socket* » pour permettre au navigateur de se connecter à lui ;
- ▷ **reçoit une requête** de la part du navigateur (« *Requête 1* » sur la figure) ;
- ▷ il **décompose** l'URL présente dans cette requête afin de récupérer :
  - ◇ le nom du serveur Web hébergeant la ressource ;
  - ◇ le numéro de port s'il est indiqué (par défaut c'est le port 80) ;
  - ◇ le chemin d'accès au document ;
- ▷ crée une **seconde** « *socket* » pour pouvoir se connecter au serveur Web ;
- ▷ **envoie une requête** au serveur Web en reprenant de manière légèrement modifiée la requête du navigateur (« *Requête 2* » sur la figure) ;
- ▷ **récupère les informations** en réponse du serveur Web (« *Réponse 1* » sur la figure) ;
- ▷ **renvoie ces informations** en réponse au navigateur (« *Réponse 2* » sur la figure).
- ▷ **attend une nouvelle connexion** du navigateur.

## ■ ■ ■ Rappel sur HTTP

Le navigateur peut transmettre des commandes « GET » ou « POST ».

Chacune de ces commandes termine par une ligne vide signifiant que le navigateur n'a plus rien à transmettre au serveur.

La commande « *GET* » est la commande la plus courante permettant de réclamer une ressource à un serveur Web.

La commande « *POST* » est une commande spéciale qui permet de transmettre au serveur Web, non seulement une requête d'accès à une ressource, mais également des données au format MIME. Ces données sont transmises par le navigateur à la suite de sa commande « POST » après une ligne vide.

Elles possèdent en particulier une taille, qui permet de les récupérer, indiquée par « Content-Length : longueur\_en\_octets ».

## ■ ■ ■ Format des requêtes du navigateur dans le cas de l'utilisation d'un proxy

Le

navigateur envoie une requête « POST » ou « GET » légèrement modifiée, dans le fait qu'elle ne contient pas directement le chemin d'accès à la ressource mais l'URL de cette ressource.

Exemple : `GET /index.html HTTP/1.1`

*devient* : `GET http://mon_serveur_web/index.html HTTP/1.1`

## ■ ■ ■ Travail

Écrire un Proxy pour le protocole HTTP qui :

- ▷ réalise un **filtrage** sur le contenu au format HTML : insertion de texte dans le titre, remplacement de texte, blocage de l'accès au contenu, suppression des ressources au format flash, *etc.*
- ▷ soit *configurable* par l'intermédiaire d'un accès Web *spécial* :  
à l'entrée d'une URL particulière dans le navigateur, on accède dans son navigateur à une interface en HTML pour configurer le proxy : (dés)activation de filtrage, édition des mots interdits, *etc.*
- ▷ gère la requête au format « GET » en provenance du navigateur ;
- ▷ gère également la requête au format « POST » (les données du POST en provenance du navigateur sont à faire suivre au serveur auquel se connecte le proxy) ;

Votre programme de proxy devra mémoriser les **paramètres** dans des fichiers de configuration au format texte.

**Remarques** : Pour le bon fonctionnement du proxy, il est nécessaire de :

- ▷ dans la requête transmise au serveur et en provenance du navigateur :
  - ☐ supprimer les lignes commençant par : Connection : Keep-Alive et Proxy-Connection : Keep-Alive ;
  - ☐ supprimer la ligne commençant par Accept-Encoding : gzip ;
- ▷ vous assurer de faire des **requêtes** en version HTTP/1.0 ;
- ▷ pour la **configuration**, vous pouvez envoyer, par exemple, un formulaire contenant un TEXTAREA contenant la liste des mots à filtrer et proposer à l'utilisateur de les modifier.

## ■ ■ ■ Examples

Après avoir configuré Firefox pour utiliser un proxy pour **http** et **https** :

▷ pour une connexion HTTP :

```
GET http://www.unilim.fr/ HTTP/1.1
Host: www.unilim.fr
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.1; rv:87.0) Gecko/20100101 Firefox/93.0
Accept: text/html,application/xhtml+xml,application/xml;...
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
DNT: 1
Connection: keep-alive
Upgrade-Insecure-Requests: 1
```

▷ pour une connexion TLS

```
CONNECT www.unilim.fr:443 HTTP/1.1
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.1; rv:87.0) Gecko/20100101 Firefox/93.0
Proxy-Connection: keep-alive
Connection: keep-alive
Host: www.unilim.fr:443
```

*La commande est CONNECT au lieu de GET.*