Le protocole HTTP

I LE PROTOCOLE HTTP COTÉ CLIENT

Le protocole de transfert hypertexte (HTTP, *Hypertext Transfer Protocol*) est un protocole de niveau application pour les systèmes d'information hypermédia distribués collaboratifs. HTTP a été utilisé par l'initiative d'informations mondiales du World-Wide Web depuis 1990. La présente spécification définit le protocole désigné sous le nom de "HTTP/1.1", qui est une mise à jour de la RFC 2068.

Les communications HTTP ont généralement lieu sur des connexions TCP/IP. L'accès par défaut est TCP 80, mais d'autres ports peuvent être utilisés. Cela n'empêche pas HTTP d'être mis en œuvre par dessus tout autre protocole sur l'Internet, ou sur d'autres réseaux. HTTP présuppose seulement un transport fiable ; tout protocole qui fournit de telles garanties peut être utilisé.

HTTP utilise un schéma de numérotation "<major>.<minor>" pour indiquer les versions du protocole. La politique de versions du protocole est destinée à permettre à l'envoyeur d'indiquer le format d'un message et sa capacité à comprendre la suite de la communication HTTP.

La version d'un message HTTP est indiquée par un champ HTTP-Version dans la première ligne du message.

Noter que les numéros majeurs et mineurs DOIVENT être traités comme des entiers séparés et que chacun PEUT être incrémenté de plus d'un chiffre.

Les URI (Uniform Resource Identifier) ont été appelés de nombreuses façons : adresses WWW, identifiants universels de documents, identifiants de ressource universels, et finalement sont la combinaison des localisateurs uniformes de ressource (URL) et des noms universels de ressource (URN). Pour ce qui concerne HTTP, les identifiants universels de ressource sont simplement des chaînes formatées qui identifient -- via le nom, la localisation, ou toute autre caractéristique -- une ressource.

Dans le protocole HTTP, une méthode est une **Commande** spécifiant un type de requête du client, c'est-à-dire qu'elle demande au serveur d'effectuer une action. En général l'action concerne une ressource identifiée par l'URI qui suit le nom de la méthode. Une requête GET est envoyée pour récupérer une page.

Par défaut, HTTP 1.1 utilise des connexions persistantes, autrement dit la connexion n'est pas immédiatement fermée après une requête, mais reste disponible pour une nouvelle requête. On appelle souvent cette fonctionnalité *keep-alive*.

Les autres paramètres de l'en-tête HTTP 1.1 sont les suivantes :

Accept
: Cet en-tête liste les types MIME de contenu acceptés par le client. Le caractère étoile * peut servir à spécifier tous les types / sous-types.

Accept-Charset
: Spécifie les encodages de caractères acceptés.

Accept-Language
: Spécifie les langues acceptées.

I.1 EXERCICE 1 :

En analysant la capture de trames présentée ci-dessous :

Donner l'adresse IP source et l'adresse Ip destination ;

Donner le port destinataire ;

Préciser si la communication a lieu en TCP ou UDP ;

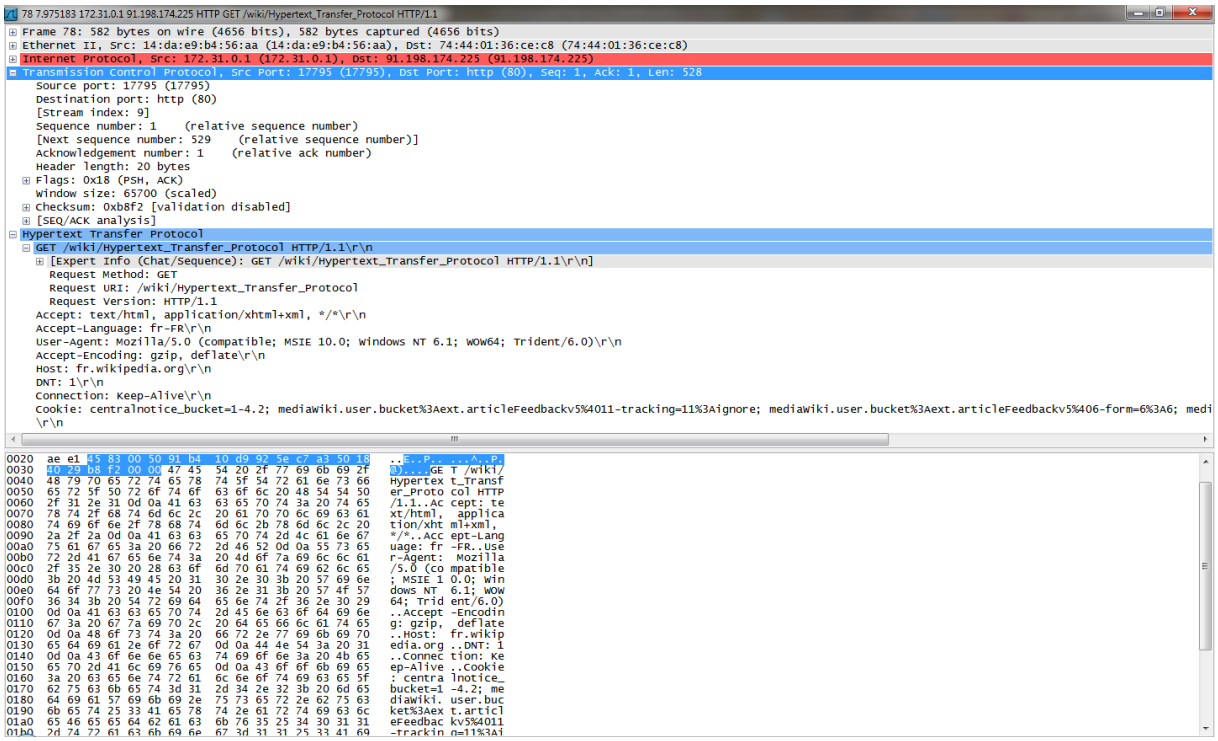
Donner la version du protocole HTTP utilisée ;

Préciser l'URI demandé ;

Donner la méthode utilisée ;

Lister les langages acceptés

Préciser si la connexion est persistante.



II LES MÉTHODES DU PROTOCOLE HTTP

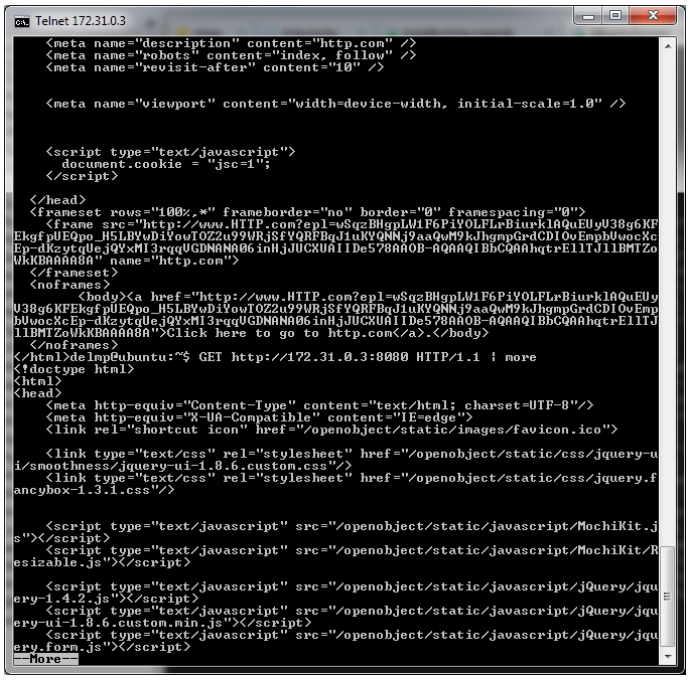
Il existe de nombreuses méthodes, les plus courantes étant GET, HEAD et POST :

GET
: C'est la méthode la plus courante pour demander une ressource. Une requête GET est sans effet sur la ressource, il doit être possible de répéter la requête sans effet.

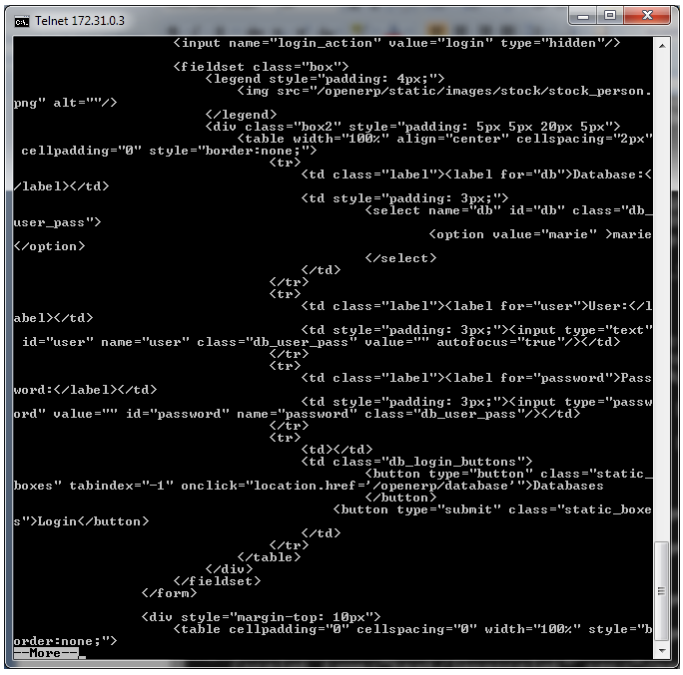
HEAD
: Cette méthode ne demande que des informations sur la ressource, sans demander la ressource elle-même.

POST
: Cette méthode est utilisée pour transmettre des données en vue d'un traitement à une ressource (le plus souvent depuis un formulaire HTML). L'URI fournie est l'URI d'une ressource à laquelle s'appliqueront les données envoyées. Le résultat peut être la création de nouvelles ressources ou la modification de ressources existantes. À cause de la mauvaise implémentation des méthodes HTTP (pour Ajax) par certains navigateurs (et la norme HTML qui ne supporte que les méthodes GET et POST pour les formulaires), cette méthode est souvent utilisée en remplacement de la requête PUT, qui devrait être utilisée pour la mise à jour de ressources.

Voici une utilisation de la méthode GET en mode console pour interroger le serveur OPENERP-WEB.



Et un autre morceau



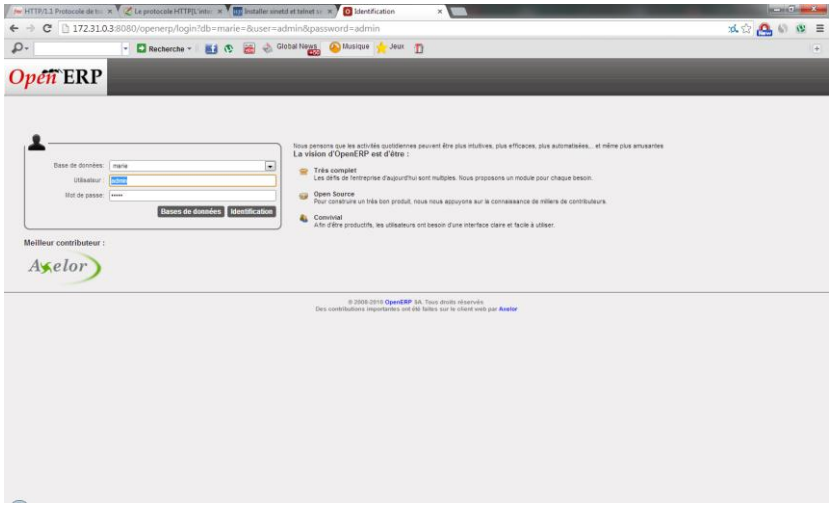
Comment travail un navigateur :

- Il appelle la page demandée GET http://172.31.0.3:8080 HTTP/1.1.
- Une fois la page reçue, il va chercher dedans tous les URI et va les appeler avec un GET. Dans le cas d'une l'image, il devra la recomposer, ce qui lui est possible parce qu'il connaît le format d'encodage gif.
- Il affiche alors la page, dans son intégralité.

La méthode GET peut transporter des paramètres (définis par l'application) pour atteindre une page précise, par exemple la page de connexion au serveur OPENERP WEB avec les paramètres "user" et "password" saisis :

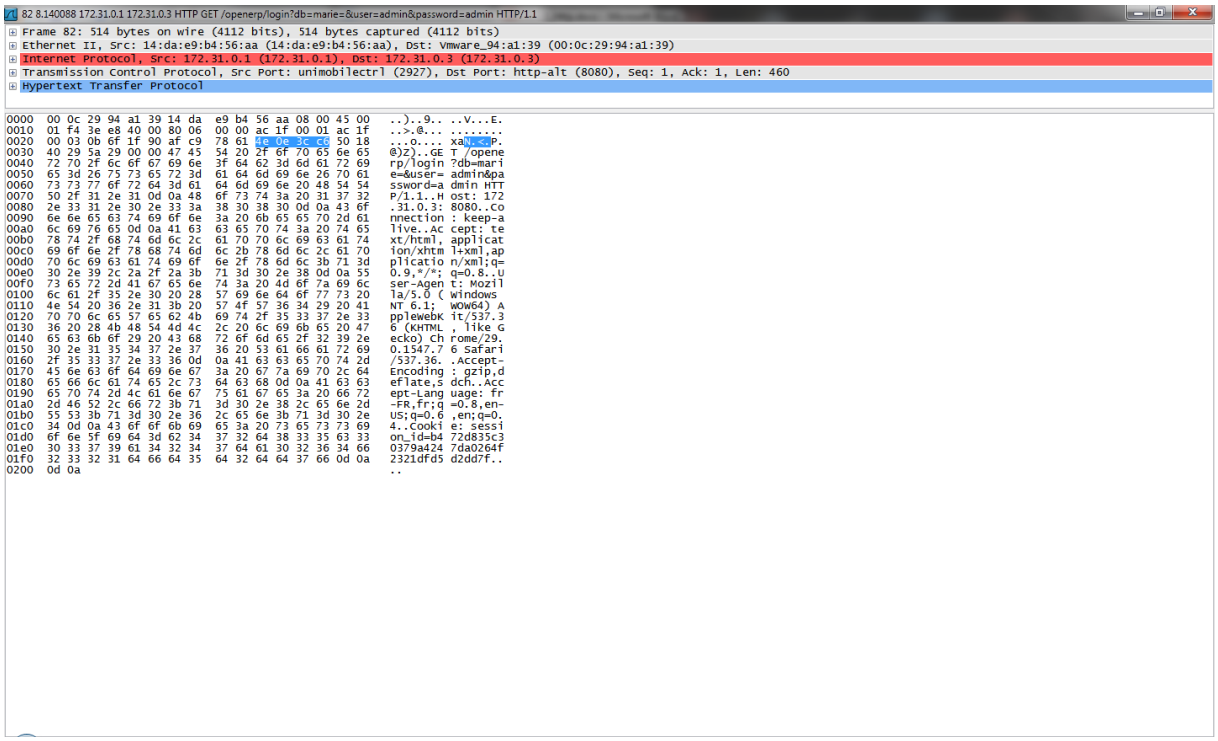
http://172.31.0.200:8080/openerp/login?db=&user=admin&password=admin

Réponse obtenue dans un navigateur:



II.1 EXERCICE 2 :

En analysant la capture de trames présentées ci-dessous :



Donner l'adresse du serveur interrogé ;

Donner le nom de la page demandée ;

Préciser si des paramètres ont été fournis ;

Conclure sur les dangers d'un tel passage de paramètres.