

# Le Protocole HTTP

# Plan

Généralités

Historique

Communication client / serveur HTTP

Requête HTTP

Réponse HTTP

HTTPS

# Généralités

**HTTP = Hypertext Transfer Protocol (en français  
« protocole de transfert hypertexte »)**

**Protocole de la couche « Application »**

**Utilise par défaut le port 80**

**Inventé par Tim Berners-Lee avec les adresses Web et le langage HTML pour créer le World Wide Web**

# Plan

Généralités

Historique

Communication client / serveur HTTP

Requête HTTP

Réponse HTTP

HTTPS

# Historique

## HTTP/0.9

1989, très élémentaires, pas de support des types MIME

## HTTP/1.0

Mai 1996, RFC 1945, support de serveurs HTTP virtuels, cache et identification

## HTTP/1.1

Janvier 1997, RFC 2068, standard IETF, support de transfert en pipeline (ou pipelining) et la négociation de type de contenu

## HTTP/1.1 bis

Juin 1999, RFC 2616

# Historique

## HTTP/2

Mai 2015, RFC 7230 à RFC 7237, réduction du temps de chargement des pages web, mobilité

# Plan

Généralités

Historique

**Communication client / serveur HTTP**

Requête HTTP

Réponse HTTP

HTTPS

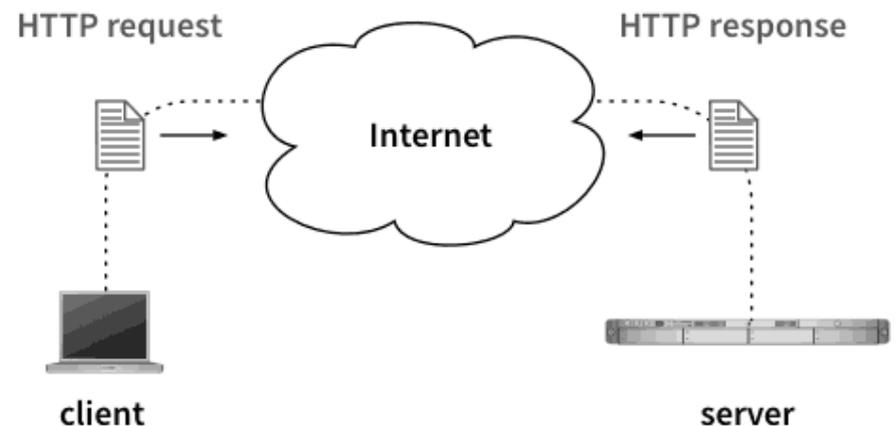
Types MIME

# Communication client / serveur HTTP

**La communication entre le navigateur et le serveur se fait en deux temps :**

Le navigateur effectue une requête HTTP

Le serveur traite la requête puis envoie une réponse HTTP



# Communication client / serveur HTTP

**La liaison entre le client et le serveur n'est pas toujours directe, il peut exister des machines intermédiaires servant de relais**

Serveur mandataire (Proxy) : peut modifier les réponses et requêtes qu'il reçoit et peut gérer un cache des ressources demandées

Passerelle (Gateway) : intermédiaire modifiant le protocole utilisé

Tunnel : transmet les requêtes et les réponses sans aucune modification, ni mise en cache

# Communication client / serveur HTTP

## Clients

Navigateurs GUI : Firefox, IE, Edge, Chrome, Safari,...

Navigateurs CLI : elinks, lynx,...

Outils GUI : Postman, RESTed,...

Outils CLI : curl, telnet,...

## Serveurs

Apache HTTP server, Zeus web server, Lighttpd, Nginx, Cherokee,...

# Plan

Généralités

Historique

Communication client / serveur HTTP

Requête HTTP

Réponse HTTP

HTTPS

Types MIME

# Requête HTTP

Ensemble de lignes envoyé  
au serveur par le client

Elle doit respecter le format  
ci-contre

```
METHODE URL VERSION<crLf>  
EN-TETE : Valeur<crLf>  
  
...  
EN-TETE : Valeur<crLf>  
Ligne vide<crLf>  
CORPS DE LA REQUETE
```

# Requête HTTP

## Exemple

```
GET http://www.commentcamarche.net HTTP/1.0
```

```
Accept : text/html
```

```
If-Modified-Since : Saturday, 15-January-2000 14:37:11 GMT
```

```
User-Agent : Mozilla/4.0 (compatible; MSIE 5.0; Windows 95)
```

# Requête HTTP

## Méthode

Commande spécifiant un type de requête, c'est-à-dire qu'elle demande au serveur d'effectuer une action

Méthode	Description
GET	Récupérer la ressource située à l'URL spécifié
HEAD	Récupérer l'en-tête de la ressource située à l'URL spécifié
POST	Envoi de données au programme situé à l'URL spécifié
PUT	Envoi de ressource au programme situé à l'URL spécifié
DELETE	Suppression de la ressource située à l'URL spécifié
OPTIONS	Obtenir les options de communication d'une ressource ou du serveur
CONNECT	Utiliser un proxy comme un tunnel de communication
TRACE	Demande au serveur de retourner ce qu'il a reçu
PATCH	Faire une modification partielle de la ressource située à l'URL spécifié

# Requête HTTP

## Entêtes

Donnent des informations / directives supplémentaires sur la requête, le client,...

Ci-contre une liste non exhaustive

Entête	Description
Accept	Type de contenu (MIME) accepté par le client
Accept-Charset	Jeu de caractères attendu par le client
Accept-Encoding	
Accept-Language	
Authorization	
Content-Length	
Content-Type	Type de contenu (MIME) du corps de la requête
Date	
Forwarded	
Referer	
User-Agent	Informations sur le client
Keep-Alive	Spécifie si on garde la connexion ouverte après la requête

# Plan

Généralités

Historique

Communication client / serveur HTTP

Requête HTTP

Réponse HTTP

HTTPS

Types MIME

# Réponse HTTP

Ensemble de lignes  
envoyé au client par le  
serveur

Elle doit respecter le  
format ci-contre

```
VERSION-HTTP CODE EXPLICATION<crLf>  
EN-TETE : Valeur<crLf>  
...  
EN-TETE : Valeur<crLf>  
Ligne vide<crLf>  
CORPS DE LA REPONSE
```

# Réponse HTTP

## Exemple

HTTP/1.0 200 OK

Date : Sat, 15 Jan 2000 14:37:12 GMT

Server : Microsoft-IIS/2.0

Content-Type : text/HTML

Content-Length : 1245

Last-Modified : Fri, 14 Jan 2000 08:25:13 GMT

# Réponse HTTP

## Codes de réponse

1xx : Message d'information

2xx : Réussite

3xx : Redirection

4xx : Erreur due au client

5xx : Erreur due au serveur

# Réponse HTTP

## Entêtes

Donnent des informations / directives supplémentaires sur la requête, le client,...

Ci-contre une liste non exhaustive

En-tête	Description
Content-Encoding	Type de codage du corps de la réponse
Content-Type	Type de contenu du corps de la réponse (par exemple text/html)
Date	Date de début de transfert des données
Expires	Date limite de consommation des données
Forwarded	Utilisé par les machines intermédiaires entre le client et le serveur
Location	Redirection vers une nouvelle URL associée au document
Server	Caractéristiques du serveur ayant envoyé la réponse

# Plan

Généralités

Historique

Communication client / serveur HTTP

Requête HTTP

Réponse HTTP

**HTTPS**

Types MIME

# HTTPS

**Variante sécurisée de HTTP par l'usage des protocoles TLS**

**En fonctionnement, il est identique à HTTP, avec intégration du TLS**

**Besoin de certificats numériques pour les mettre en œuvre côté serveur**

Il faut s'adresser à une autorité de certification : Verisign, Comodo, Let's Encrypt,...

# Plan

Généralités

Historique

Communication client / serveur HTTP

Requête HTTP

Réponse HTTP

HTTPS

Types MIME

# Types MIME

**MIME = Multipurpose Internet Mail Extensions  
(Extensions multifonctions du courrier Internet)**

**Standard internet qui étend le format de données pour supporter des textes en différents codage des caractères autres que l'ASCII (non textuels, multiples, autres codages,...)**

**Initialement défini pour les courriels à travers SMTP, ils sont utilisés aujourd'hui dans beaucoup d'autres protocoles comme HTTP**

# Types MIME

**Un type MIME est écrit sous la forme suivante :**

type/sous-type

## Exemples

text/tml

text/json

multipart/digest

multipart/mixed

image/jpeg

image/png

# Merci

## Références

RFC 2068 et 2616, HTTP 1.1

<https://tools.ietf.org/html/rfc2068>

<https://tools.ietf.org/html/rfc2616>

Codes de réponses HTTP

[https://fr.wikipedia.org/wiki/Liste\\_des\\_codes\\_HTTP](https://fr.wikipedia.org/wiki/Liste_des_codes_HTTP)

En-têtes HTTP

<https://developer.mozilla.org/fr/docs/Web/HTTP/Headers>

Types MIME

[https://fr.wikipedia.org/wiki/Type\\_de\\_m%C3%A9dias](https://fr.wikipedia.org/wiki/Type_de_m%C3%A9dias)

<http://www.iana.org/assignments/media-types/media-types.xhtml>