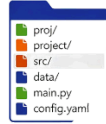


Data Storage



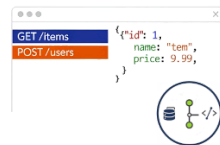
Naming Conventions



GIT Workflow



FastAPI Development



Applications et protocoles d'Internet et du Web

Fabien Baillon

02 mars 2026

Attribution - Partage dans les Mêmes Conditions : <http://creativecommons.org/licenses/by-sa/3.0/fr/>



Table des matières

- Objectifs
- Contexte
- Internet et ses protocoles
- Appliquer la notion
- SSH : Secure Shell
- Appliquer la notion
- FTP : File Transfer Protocol
- Appliquer la notion
- HTTP : HyperText Transfert Protocol
- Appliquer la notion
- Essentiel
- Exercice final
- Crédits des ressources
- Contenus annexes



1. Objectifs

- Connaître les principaux protocoles du Web ;
- Savoir se connecter en SSH et en SFTP.



2. Contexte

Internet est un réseau permettant aux ordinateurs du monde entier de communiquer. Cependant, un réseau ce n'est pas que des tuyaux, aucune communication n'est possible s'il n'existe pas des conventions sur la façon de communiquer. On parle de protocoles.

Ce module présente les principaux protocoles utilisés sur Internet, à savoir :

- SSH pour ouvrir un shell sécurisé sur un serveur distant (afin de le commander),
- HTTP pour transmettre des fichiers web (et ainsi naviguer avec son navigateur sur les serveurs du monde entier),
- SMTP, IMAP et POP pour s'échanger des mails,
- FTP pour échanger des fichiers directement entre ordinateurs.





3. Internet et ses protocoles

Objectifs

- Rappeler le fonctionnement d'Internet ;
- Découvrir la notion de protocole ;
- Découvrir les différents protocoles d'Internet.

Mise en situation

Internet est un réseau reliant des ordinateurs du monde entier. Il rend possible des applications, comme : le Web, le transfert de fichiers ou les mails. Comment ces différents ordinateurs, de marques et de types différents, peuvent-ils arriver à communiquer entre eux ?

Si des PC Windows, des téléphones Android, des tablettes Apple ou même des voitures Tesla, parviennent à communiquer avec des serveurs Linux, c'est parce que, *in fine*, tous ces ordinateurs utilisent des protocoles communs.

Modèle OSI

Rappel

Le **modèle OSI** (pour *Open Systems Interconnection*) est une norme de communications, en réseau, de tous les systèmes informatiques. C'est un modèle conceptuel qui définit le fonctionnement d'un réseau global en **7 couches**. Chaque couche réalise un **service** accessible via l'**interface** de la couche. Ce service est réalisé grâce à un **protocole**.

1. La couche « **physique** » est chargée de la transmission effective des signaux entre les interlocuteurs : elle est limitée à l'émission et la réception de bits.
2. La couche « **liaison de données** » gère les communications entre deux machines connectées entre elles.
3. La couche « **réseau** » gère les communications de proche en proche (routage et adressage des paquets).
4. La couche « **transport** » gère les communications de bout en bout entre programmes en cours d'exécution.
5. La couche « **session** » gère la synchronisation des échanges.
6. La couche « **présentation** » est chargée du codage des données applicatives (conversion entre données manipulées au niveau applicatif et chaînes d'octets transmises).
7. La couche « **application** » est le point d'accès aux services réseaux.

https://fr.wikipedia.org/wiki/Mod%C3%A8le_OSI

Protocole

Az Définition

Un protocole informatique est un **ensemble de règles** qui régissent les **échanges de données** entre plusieurs processus ou entre des ordinateurs en réseaux.



Protocole dans un bureau de vote

👁 Exemple

Les règles dans un bureau de vote peuvent être vues sous l'angle d'un protocole : le votant se présente, récupère les différents bulletins, réalise un choix tenu secret, vote en soumettant son bulletin, et signe pour attester du vote.

Applications et protocoles d'Internet

💡 Fondamental

Parmi les protocoles et applications les plus importants de la couche applicative d'Internet on retrouve :

- les **connexions à des machines distantes** avec **SSH** (*Secure Shell*),
- la consultation de documents sur le **Web** avec **HTTP** (*HyperText Transfert Protocol*),
- le **transfert de fichiers** assuré par le protocole **FTP** (*File Transfert Protocol*),
- l'**envoi et la réception d'e-mails** avec **SMTP** (*Simple Mail Transfert Protocol*), **IMAP** (*Internet Message Access Protocol*) et **POP** (*Post Office Protocol*).

💬 Remarque

Chaque protocole dispose d'un **port** par défaut qui lui est réservé.

À retenir

- Le modèle OSI est un modèle conceptuel d'Internet composé de sept couches.
- Les protocoles des applications d'Internet se trouvent sur la septième couche et disposent d'un port réservé.
- Les protocoles principaux des applications d'Internet sont SSH, HTTP, FTP et SMTP/POP/IMAP.



4. Appliquer la notion

Les protocoles principaux des applications d'Internet sont : SSH, HTTP, FTP, SMTP, IMAP et POP. Mais il existe de nombreux autres protocoles pour la couche applicative d'Internet.

Consultez la page en.wikipedia.org/wiki/Application_layer (la page en anglais est bien plus complète que celle en français).

Exercice

Parmi ces propositions, lesquelles sont des protocoles **de la couche applicative** ?

A FreeNet

B DNS

C IRCP

D Tor

E TCP

F WebDAV

G IP

H Ethernet

I SSMS



5. SSH : Secure Shell

Objectif

- Se familiariser avec le protocole SSH (Secure SHell).

Mise en situation

SSH est une application d'Internet qui n'est pas aussi connue que le Web ou le mail.

Mais elle est très utile dans le monde du développement informatique car elle permet de se connecter à un serveur à distance, puis d'exécuter des commandes sur ce serveur. C'est aujourd'hui le mode de contrôle privilégié des informaticiens sur les serveurs qui font fonctionner Internet.

Terminal et shell

Rappel

Le **terminal** (aussi appelé invite de commande ou console) est le moyen le plus naturel pour interagir avec un ordinateur. Ce programme offre un **shell** (une interface système) donnant accès aux programmes de l'ordinateur. Il suffit d'un écran et d'un clavier pour utiliser ce shell.

- *Terminal et shell* (cf. p.39)

Le programme SSH (Secure SHell)

Az Définition

Le programme SSH (*Secure Shell*) permet d'interagir de manière sécurisée avec un ordinateur distant via un shell. C'est le programme de référence pour effectuer des opérations à distance.

- Toutes les commandes tapées depuis un clavier d'ordinateur à un emplacement A sont exécutées dans le shell d'un ordinateur à un emplacement B.
- Les informations qui transitent sur Internet via SSH sont les chaînes de caractères représentant les commandes à exécuter et les chaînes de caractères représentant les résultats de ces exécutions.

Le protocole SSH

Az Définition

SSH est aussi le nom du protocole de communication utilisé par le programme SSH. Ce protocole fonctionne sur une **architecture client-serveur** ; l'ordinateur qui fournit les commandes est le **client** et l'ordinateur qui exécute les commandes sur son système est le **serveur**.

SSH repose généralement sur TCP pour le transport et est par défaut associé au port **22**.

L'établissement d'une connexion SSH se fait en deux étapes :

- l'établissement d'une communication sécurisée,
- l'authentification du client.



Établir une connexion SSH

Méthode

On utilise la commande `ssh` depuis un terminal :

```
1 ssh alice@adresse.ip
```

La machine se trouvant à l'adresse `serveur.exemple.com` doit avoir un utilisateur `alice`.

Sur Windows, Putty est un client SSH qui permet d'ouvrir une console sous SSH. Il est téléchargeable sur le site dédié : <https://www.putty.org/>

Attention

Si `ssh` se connecte pour la première fois au serveur, l'utilisateur devra accepter d'utiliser la clé publique du serveur pour des raisons de sécurité.

Remarque

Si un serveur est associé à un nom de domaine, on peut utiliser ce domaine à la place de l'IP.

```
1 ssh alice@serveur.exemple.com
```

Chiffrement symétrique et chiffrement asymétrique

Complément

Le **chiffrement symétrique** repose sur le chiffrement et le déchiffrement de messages par une seule clé appelée **clé symétrique**. Ainsi, si deux ordinateurs veulent protéger leurs communications, ils peuvent se mettre d'accord sur une clé symétrique à utiliser et veiller à ce qu'ils soient les seuls à la détenir. Le chiffrement symétrique est rapide mais nécessite un canal sécurisé pour échanger la clé.

Le **chiffrement asymétrique** repose sur le chiffrement et le déchiffrement de messages par deux clés différentes (l'une chiffre les messages et l'autre déchiffre). Dans ce système, une clé est connue de tous : on parle de **clé publique** ; l'autre clé est gardée secrètement : on parle de **clé privée**. Le chiffrement asymétrique est plus lent mais ne nécessite par de canal sécurisé préalable.

Étape 1 : Établissement d'une communication sécurisée

Complément

Au début de la session et avant de commencer toute interaction entre le client et le serveur, c'est-à-dire avant même que le serveur vérifie que le client est légitime, les deux ordinateurs doivent se mettre d'accord sur un moyen de sécuriser leur communication.

Pour cela, le client et le serveur se mettent d'accord via le protocole de transport (TCP) sur la version du protocole SSH à utiliser et sur une méthode de **chiffrement symétrique** de leur communication.



Les deux ordinateurs vont utiliser des méthodes de chiffrement asymétrique pour échanger une clé symétrique. Cette clé leur permettra donc de protéger toutes les communications suivantes de la session.

À la fin de cette étape, une connexion sécurisée est établie entre le client et le serveur.

Étape 2 : Authentification de l'utilisateur côté client

⊕ Complément

Une fois une connexion sécurisée établie, le serveur doit **authentifier** l'utilisateur qui cherche à se connecter. Il existe deux méthodes d'authentification :

- **Méthode 1 : Par mot de passe**

Puisque le but est ici d'ouvrir un shell à distance, le client cherche à se connecter à un utilisateur de la machine distante. Le serveur peut ainsi demander au client le mot de passe de l'utilisateur via lequel il veut se connecter.

- **Méthode 2 : Par clé**

Chaque utilisateur présent sur le serveur possède une liste de clés publiques de clients de confiance, présentes dans un fichier `authorized_keys`.

1. Le client envoie une des clés publiques de l'utilisateur avec lequel il veut se connecter.
2. Le serveur vérifie le fichier `authorized_keys` de l'utilisateur pour s'assurer que cette clé publique existe.
3. Si oui, le serveur chiffre un nombre aléatoire avec ladite clé publique et envoie le résultat au client.
4. Le client utilise sa clé privée pour déchiffrer le nombre et y appliquer un certain traitement avant de le renvoyer au serveur.
5. Le serveur applique le même traitement au nombre qu'il a envoyé et vérifie que le résultat du client est le même que le sien.
6. Si le résultat est correct, et comme l'utilisateur est le seul à posséder la clé privée et déchiffrer le nombre envoyé par le serveur, il est authentifié.

⊕ Complément

La méthode d'authentification par clé nécessite qu'une paire de clés publique/privée soit générée au préalable, et que le serveur ait stocké la clé publique du client dans le fichier `authorized_keys` de l'utilisateur.

Installer un serveur SSH sur un serveur

⊕ Complément

Pour rendre un serveur accessible à distance, il faudra installer et lancer un serveur SSH. L'implémentation de référence est OpenSSH que l'on peut installer et activer ainsi sur des serveurs GNU/Linux de la famille Debian.



```
1 sudo apt update
2 sudo apt install openssh-server
3 sudo systemctl start ssh
```

À retenir

- Le protocole SSH permet à un client d'ouvrir un shell sur un serveur distant.
- Ce protocole sécurise les communications en employant plusieurs méthodes de chiffrement, de l'authentification au transfert de données.



6. Appliquer la notion

Pré-requis

Disposer d'un compte chez <https://www.alwaysdata.com/fr/>. Alwaysdata est un hébergeur qui fournit des services gratuits sur un Cloud Privé.

Un accès SSH permet de se connecter à ses serveurs.

Inscrivez-vous ici avec votre adresse mail IMT Mines Albi : <https://www.alwaysdata.com/fr/inscription/?teacher=23040228>.

Lors de l'inscription sur le site de **Alwaysdata**, vous allez créer un utilisateur-administrateur par défaut, en lui donnant un nom que vous devez choisir.

Par exemple, votre utilisateur peut s'appeler *tartempion*.

The screenshot shows a web form for creating an Alwaysdata account. The form is titled "Bienvenue" and "Choix de l'offre". It contains several input fields and checkboxes. The "Produit*" field is set to "Cloud public gratuit (100 Mo)". The "Nom*" field contains "tartempion". The "Mot de passe*" field is masked with dots. There are three checkboxes on the right side, all of which are checked: "Conditions générales*", "Conditions particulières (hébergement mutualisé)*", and "Permissions accordées à fabien.baillon@mines-albi.fr". A pink button labeled "C'est parti !" is located at the bottom right of the form.

Création d'un compte Alwaysdata

À titre d'illustration, dans ce qui suit, le nom de l'utilisateur-administrateur sera **tartempion**.

Question 1

Configurez votre compte pour qu'il autorise l'accès à distance par **SSH**.

Indice :

L'accès **SSH** n'est pas activé par défaut. Il faut l'activer en allant à l'onglet **Accès distant**, rubrique **SSH**, dans l'espace d'administration de votre site.

Indice :

L'activation **SSH** nécessite d'activer la **Connexion par mot de passe**, ici : <https://admin.alwaysdata.com/ssh/>

Question 2

Se connecter à ce serveur en utilisant une authentification par mot de passe. Que contient le dossier `www` sur votre serveur ?

Indice :

Il faut utiliser la commande `ssh` depuis un shell local.



Indice :

Une connexion ssh nécessite :

- L'adresse du serveur,
- Le nom d'un utilisateur,
- Le mot de passe de l'utilisateur.

Question 3

Lors de la première connexion, pourquoi le terminal a-t-il demandé si vous faisiez confiance au serveur ?

Question 4

Vérifier que la clé SSH du serveur a bien été ajoutée aux hôtes connus (`known_hosts`).

La commande suivante permet de vérifier qu'un hôte est connu sur votre client SSH :

```
1 ssh-keygen -F ip_serveur
```

La commande renvoie des informations sur le serveur, la méthode de chiffrement utilisée ainsi que la clé publique.



7. FTP : File Transfer Protocol

Objectif

- Se familiariser avec FTP.

Mise en situation

Si le mail permet l'échange de messages, il est mal adapté à l'échange de fichiers, surtout s'ils sont nombreux ou volumineux. Le protocole FTP est historiquement le premier à avoir rempli ce rôle. Un client FTP peut déposer et récupérer des fichiers sur un serveur FTP.

Le principal défaut de FTP est qu'il n'est pas chiffré. Ainsi il est remplacé aujourd'hui par ses variantes sécurisées SFTP et FTP/S. On utilise aussi parfois HTTP pour échanger des fichiers, via des sites comme *Firefox Send*.

FTP — File Transfer Protocol

Az Définition

FTP est le protocole de référence pour le **transfert de fichiers**. Il respecte une architecture client-serveur entre un **client FTP** et un **serveur FTP**.

Voici quelques fonctions permises par le protocole :

- établir une connexion
- envoyer son identifiant/nom d'utilisateur
- envoyer son mot de passe
- afficher le dossier courant
- changer de dossier courant
- récupérer une copie d'un fichier (téléchargement)
- déposer une copie d'un fichier (téléversement)
- supprimer un fichier
- ...

FTP a besoin d'un protocole de transport fiable tel que TCP (on ne peut pas se permettre de perdre des paquet quand on transfère des fichiers en général).

Il fonctionne par défaut sur le port 21 pour recevoir les commandes FTP et le port 20 pour recevoir les données.

Gérer les fichiers d'un site web

Exemple

Un développeur a besoin de transférer sur son serveur les fichiers qui constituent son site web. Il va pouvoir utiliser le protocole FTP pour transférer ses fichiers depuis son poste client jusqu'au serveur fourni par l'hébergeur.



Les méthodes sécurisées

💡 Fondamental

FTP pose un important problème de sécurité par l'absence de chiffrement des communications.

C'est pourquoi il existe aujourd'hui des protocoles alternatifs qui chiffrent les communications par défaut.

Ce sont ces protocoles qui sont recommandés.

- **SFTP** (SSH File Transfer Protocol) utilise une session SSH pour transférer les fichiers et implémentent les commandes FTP. C'est le protocole à privilégier lorsqu'on dispose d'un accès SSH à un serveur.
- **FTP/S** est une extension de FTP lui permettant de supporter le protocole TLS. TLS fournit un chiffrement des communications sur un réseau.

SFTP en ligne de commande

🔧 Méthode

```
1 sftp répertoire-d-arrivée utilisateur@adresse-ip
```

Une fois connecté, plusieurs commandes analogues aux commandes Linux sont disponibles pour interagir avec le système de fichiers.

Par exemple :

- `ls` pour lister les fichiers sur le serveur
- `lls` (local ls) pour liste les fichiers sur le client
- `pwd` pour voir le répertoire courant
- `cd` pour changer de répertoire
- `get` pour télécharger un fichier depuis le serveur
- `put` pour téléverser un fichier sur le serveur

SFTP avec FileZilla

🔧 Méthode

FileZilla est un client graphique qui permet de communiquer avec un serveur en FTP, SFTP et FTP/S.

Sur Linux (Debian et Ubuntu), on pourra installer FileZilla avec la commande suivante :

```
1 sudo apt install filezilla
```

Sinon, on pourra télécharger FileZilla depuis cette page : https://filezilla-project.org/download.php?show_all=1

Alternatives à FTP

➕ Complément

Il existe d'autres protocoles et outils pour échanger des fichiers, comme les programmes en ligne de commande `scp` et `rsync` (qui s'appuient sur le protocole SSH).

```
1 scp chemin/local/fichier user@<adresse-IP>:chemin/de/réception/
```



```
1 scp chemin/local/fichier user@<adresse-IP>:chemin/de/réception/
```

À retenir

- FTP est un protocole utilisé pour transférer des fichiers sur Internet.
- Il peut être utilisé pour envoyer des pages web sur le serveur d'un hébergeur.
- SFTP et FTP/S sont des protocoles similaires qui fonctionnent comme FTP mais avec des communications chiffrées.



8. Appliquer la notion

Pré-requis

Disposer d'un compte chez <https://www.alwaysdata.com/fr/>. Alwaysdata est un hébergeur qui fournit des services gratuits sur un Cloud Privé.

Un accès SSH permet de se connecter à ses serveurs.

Inscrivez-vous ici avec votre adresse mail IMT Mines Albi : <https://www.alwaysdata.com/fr/inscription/?teacher=23040228>.

Lors de l'inscription sur le site de **Alwaysdata**, vous allez créer un utilisateur-administrateur par défaut, en lui donnant un nom que vous devez choisir.

Par exemple, votre utilisateur peut s'appeler *tartempion*.

The screenshot shows a web form for creating an Alwaysdata account. The form is titled "Bienvenue" and "Choix de l'offre". It contains several input fields and checkboxes. The "Produit*" field is set to "Cloud public gratuit (100 Mo)". The "Nom*" field contains "tartempion". The "Mot de passe*" field is masked with dots. There are three checkboxes on the right side, all of which are checked: "Conditions générales*", "Conditions particulières (hébergement mutualisé)*", and "Permissions accordées à fabien.baillon@mines-albi.fr". A pink button labeled "C'est parti !" is located at the bottom right of the form.

Création d'un compte Alwaysdata

À titre d'illustration, dans ce qui suit, le nom de l'utilisateur-administrateur sera **tartempion**.

Question 1

Vérifier la présence du logiciel FileZilla sur l'ordinateur local (si ce n'est pas le cas, installez le !).

Ce client peut envoyer des fichiers avec FTP, SFTP et FTP/S.

Question 2

Ouvrir une session SFTP sur votre compte Alwaysdata depuis FileZilla.

Indice :

Pour indiquer que l'on souhaite utiliser le protocole SFTP, le nom d'hôte doit avoir le format suivant :

```
1 sftp://adresse-ip
```

Indice :

SFTP utilise SSH pour transférer les fichiers, le port à renseigner est donc celui de SSH.

Indice :

Le port par défaut de SSH est le port 22.



Question 3

Faites de même, mais maintenant dans un Terminal sur votre ordinateur local.

Indice :

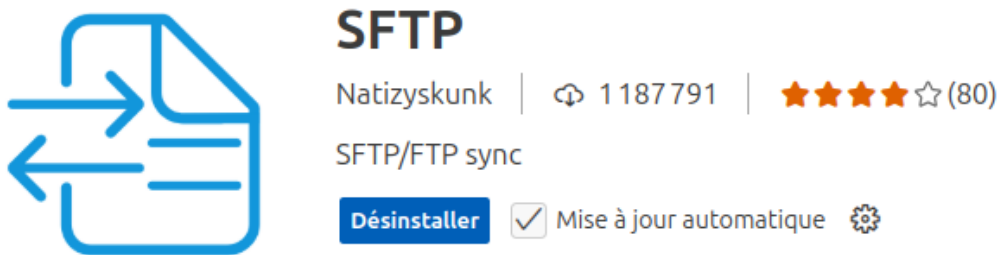
La commande linux pour utiliser **SFTP** est `sftp`.

Indice :

Il faut préciser l'utilisateur et l'adresse du serveur sur lequel on veut se connecter. Le mot de passe sera demandé à la connexion.

Différentes extensions existent dans Visual Studio Code, pour gérer des synchronisations de fichiers en **SFTP**.

On vous propose d'installer celle-ci.



Extension SFTP dans VSCode

Question 4

Créez un dossier local par exemple `www_tartempion` et synchronisez son contenu avec le contenu du dossier `www` de votre serveur Alwaysdata.

Indice :

Ouvrez le dossier `www_tartempion` depuis VSCode. Ouvrez la Palette de commandes (**F1**) pour lancer la commande SFTP : `Config`.

Indice :

Configurez le fichier `sftp.json` avec les informations de configuration de votre accès **SFTP** (adresse de l'hôte, utilisateur, et port).

Indice :

Le chemin de dossier distant `www` est `/home/tartempion/www/`.



9. HTTP : HyperText Transfert Protocol

Objectif

- Se familiariser avec HTTP.

Mise en situation

HTTP est le protocole du Web : il permet à un client (le navigateur) d'obtenir des pages et des ressources depuis un serveur web (le site).

On connaît ce protocole car les adresses web commencent par `http://`.

Par exemple, `http://en.wikipedia.org/wiki/Tim_Berners-Lee` signifie : je souhaite communiquer avec le serveur `en.wikipedia.org` en utilisant le protocole HTTP, je lui demande de me donner la page qui s'appelle `Tim_Berners-Lee`. On appelle cela une requête HTTP.

Une courte histoire du Web

Rappel

HTTP est conçu au début des années 90 par Tim Berners-Lee¹ dans le cadre de ses travaux au CERN. L'objectif est de mettre en place un système qui permette l'échange d'informations et de ressources, entre les chercheurs du CERN, en s'appuyant sur le réseau interne. Le projet prend rapidement le nom de *World Wide Web* et abouti à la création du protocole HTTP et du premier navigateur.

HTTP

Az Définition

HTTP (pour *HyperText Transfert Protocol*) est le protocole utilisé par le Web pour le transfert de ressources.

Ce protocole fonctionne sous la forme de **requête-réponse** dans une architecture **client-serveur** : le client adresse une requête HTTP à un serveur qui lui enverra une réponse appropriée.

Ce protocole repose généralement sur un protocole de transport fiable, tel que TCP.

Port 80

Fondamental

Le port **80** est le port réservé par les serveurs Web pour HTTP.

Les requêtes HTTP

Il existe différents types de requêtes HTTP, voici les plus importantes :

- **GET** : c'est la plus commune et elle demande à recevoir une certaine ressource disponible sur un serveur web. Quand un client fait une requête **GET**, c'est qu'il veut uniquement recevoir une ressource et qu'il ne cherche pas à la modifier.

¹ Tim Berners-Lee - https://fr.wikipedia.org/wiki/Tim_Berners-Lee



- **POST**: une requête **POST** est utilisée dans l'intention de modifier ou ajouter une ressource sur le serveur web.
- **DELETE** : elle est utilisée par le client lorsqu'il veut supprimer une ressource d'un serveur web.

Format des requêtes

Les requêtes HTTP suivent le format suivant :

- le type de la requête (**GET**, **POST**, etc.) avec la référence de la ressource en question sur le serveur,
- des en-têtes donnant diverses informations au serveur,
- un corps optionnel contenant des données utiles à la requête ; dans le cas de l'ajout d'une ressource, le corps contiendra la ressource en elle-même.

Un message HTTP

👁 Exemple

```
1 GET /wiki/Hypertext_Transfer_Protocol HTTP/1.1
2 Host: www.wikipedia.org
```

On demande à recevoir (**GET**) la ressource `/wiki/Hypertext_Tansfert_Protocol` se trouvant sur le serveur `www.wikipedia.org`.

La directive `Host` : est une en-tête.

Réaliser une requête GET

🔗 Méthode

- Les navigateurs web exécute une requête **GET** à chaque fois qu'un utilisateur demande à consulter une page web.
- Il existe des programmes tels que `wget` ou `curl` pour effectuer des requêtes HTTP en ligne de commande.

```
1 curl https://fr.wikipedia.org/wiki/Hypertext_Transfer_Protocol
1 wget https://fr.wikipedia.org/wiki/Hypertext_Transfer_Protocol
```

Cette commande enregistre le contenu de la page `Hypertext_Transfer_Protocol` de Wikipédia. Contrairement à un navigateur web, la page n'est pas affichée ni mise en page.

Les en-têtes HTTP

Les en-têtes (ou *headers*) HTTP permettent d'ajouter de nombreuses informations aux requêtes, ce qui facilite l'ajout de fonctionnalités au protocole : gestion du cache, de l'authentification, des formats de données à échanger, etc.

👁 Exemple

Nous avons vu plus haut une requête utilisant le *header* `Host`. Un autre très utilisé est le *header* `Authorization` qui est utilisé pour gérer l'accès avec des identifiants à une ressource. Sans rentrer dans le détail, voici un exemple :

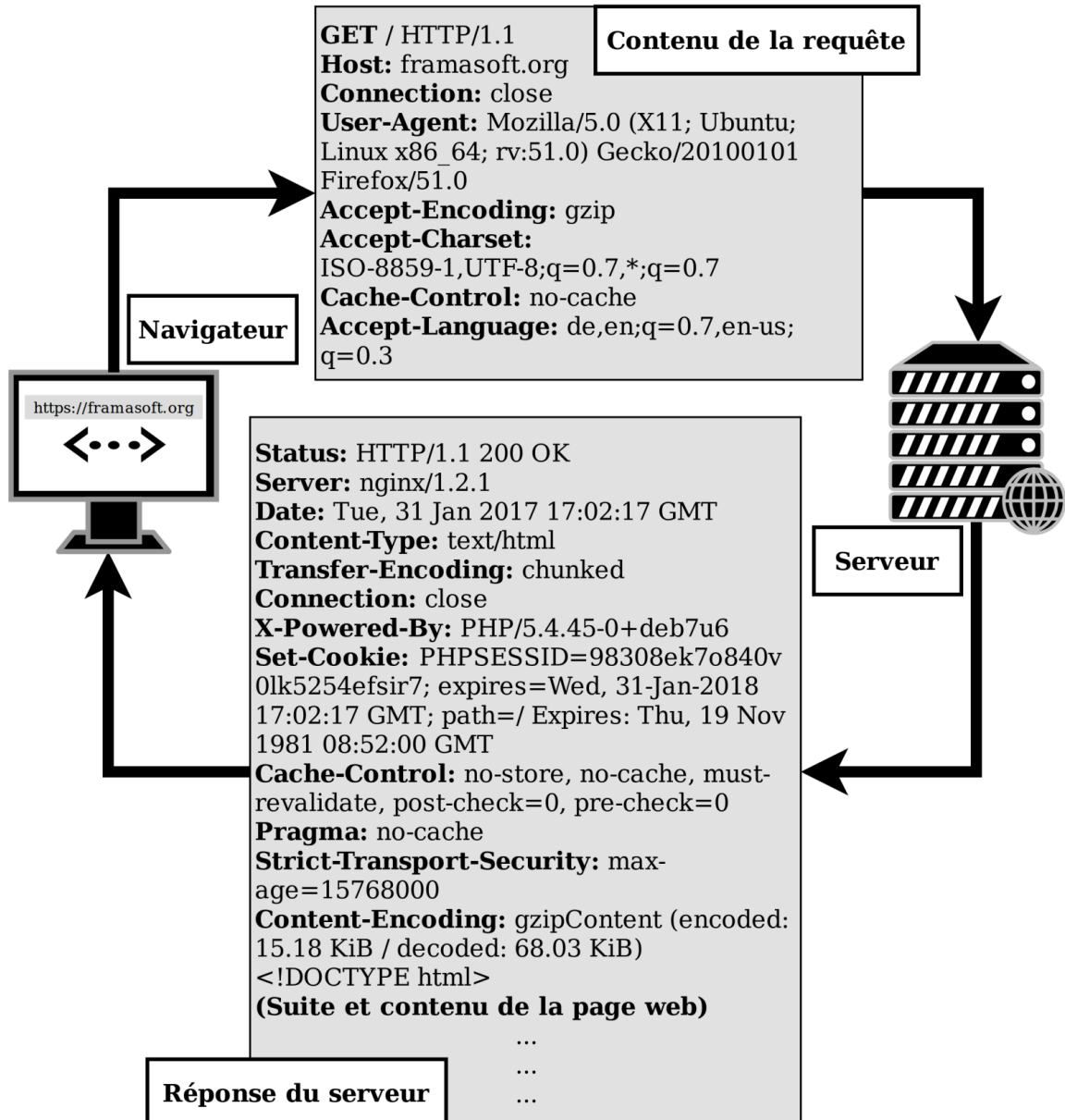
```
1 GET /internal/secret_file.txt HTTP/1.1
2 Authorization: Basic a3lhbmU6
```



Ici on demande l'accès à la ressource /internal/secret_file.txt en utilisant de l'authentification. L'en-tête `Authorization` permet de spécifier que l'on utilisera la méthode d'authentification "Basic" (il en existe plusieurs) et on passe comme valeur d'identifiant "a3lhbmU6".

Récapitulatif

Fondamental



Contenu d'une requête HTTP et de sa réponse

À retenir

- HTTP est le protocole du Web qui permet d'échanger des ressources, le plus souvent des fichiers HTML.
- Il permet de faire des requêtes pour demander à consulter, ajouter, modifier ou supprimer une ressource sur un serveur web.



⊕ Complément

- *Serveurs web* (cf. p.42)
- *Navigateurs web* (cf. p.43)



10. Appliquer la notion

Question 1

Utiliser `wget` pour télécharger la page d'accueil de Wikipédia (`www.wikipedia.org`).

Indice :

Le programme `wget` prend pour argument l'adresse de la page à télécharger.

Question 2

Afficher le contenu de la page.

Indice :

On peut utiliser les programmes `cat` ou `more`.



11. Essentiel

Internet est un réseau physique qui permet de relier les machines entre elles grâce à des câbles et des ondes. Mais c'est aussi un ensemble de protocoles, c'est à dire de langages de communication standard.

Ce sont ces protocoles qui permettent à tous les ordinateurs du monde de communiquer entre eux, indépendamment de leur architecture physique, de leur système d'exploitation ou même de leurs logiciels.

Il existe des protocoles pour échanger des mails (SMTP, IMAP et POP) ou encore pour échanger des fichiers (comme FTP).

Il existe des protocoles pour se connecter à des serveurs à distance (comme SSH).

Il existe enfin le protocole qui permet l'existence du Web : HTTP. C'est ce langage qui permet aux navigateurs web de demander des ressources à des serveurs web : page HTML, image, vidéo, etc. Les navigateurs effectuent ces requêtes grâce à la mécanique des URL qui permettent de donner une adresse unique à chaque ressource présente sur un serveur web connecté à Internet.



12. Exercice final

Quiz - Culture

Exercice 2

Quels protocoles sont utilisés pour consulter ses mails sur un serveur distant ?

A HTTP

B IMAP

C POP

D SSH

E SMTP

Exercice 3

Quel protocole peut-on utiliser pour ouvrir un shell sur un serveur hébergeant un site web ?

A URL

B SSH

C HTTP

D Aucune de ces réponses

Exercice 4

Quel protocole peut être utilisé pour envoyer des fichiers web sur son serveur web ?

A HTTP

B POP

C FTP



D Aucune de ces réponses

Exercice 5

Quels types de chiffrement utilise SSH ?

A Le chiffrement symétrique

B Le chiffrement asymétrique

C Le chiffrement elliptique

D Aucune de ces réponses

Exercice 6

Pourquoi est-il déconseillé d'utiliser FTP aujourd'hui ?

A La façon utilisée pour transférer les fichiers peut saturer la bande passante

B Les mots de passe sont transmis en clair sur le réseau

C TCP ne supporte pas FTP

D Aucune de ces réponses

Quiz - Méthode

Exercice 8

Comment peut-on s'authentifier en SSH ?

A Via le mot de passe de l'utilisateur du serveur distant

B Via un mot de passe global choisi par le serveur

C Grâce à une paire de clés publique/privée dont la publique se trouve sur le serveur distant

D En appelant l'administrateur système du serveur distant pour lui demander un accès unique



Exercice 9

Sur les navigateurs Mozilla Firefox et Google Chrome, l'appui sur la touche F12 ouvre un ensemble d'outils destiné aux développeurs web.

Quel onglet permet d'inspecter les couples requêtes-réponses HTTP ?

A Inspecteur

B Performance

C Console

D Réseau

Quiz - Code

Exercice 11

À quoi sert un serveur web ?

A À consulter sur un site web

B À servir des fichiers web

C À traiter des requêtes HTTP

D À interpréter du JavaScript

Exercice 12

À quoi sert un navigateur web ?

A À consulter sur un site web

B À envoyer des requêtes HTTP

C À exécuter du code JavaScript

D À mettre en forme les fichiers web reçus.

Exercice 13

À quoi sert une URL ?



A À transformer un nom de domaine en adresse IP

B À identifier une ressource sur un serveur web

C À passer des paramètres à un serveur web

Exercice 14

On se donne l'URL :

<http://une.bellepage.org?hash=sha256>

Quel est le nom du paramètre de cette URL ?

A hash

B une

C sha256

D http

E Aucune de ces réponses



Crédits des ressources

p. 7

Licence Freepik — https://fr.freepik.com/vecteurs-libre/communication-internet_5188745.htm

Création d'un compte Alwaysdata p. 17, 23

Attribution - Partage dans les Mêmes Conditions - Fabien Baillon

Extension SFTP dans VSCode p. 24

Attribution - Partage dans les Mêmes Conditions - Fabien Baillon

Contenu d'une requête HTTP et de sa réponse p. 27

Attribution - Partage dans les Mêmes Conditions - framasoftware.org

Outils de développement Firefox (onglet requête) p. 46

Universel - Transfert dans le Domaine Public



Contenus annexes

1. À la découverte du terminal

Objectif

- Savoir ouvrir un terminal et exécuter une commande.

Mise en situation

On a l'habitude d'interagir avec les ordinateurs en mode graphique, c'est-à-dire en utilisant essentiellement la souris pour cliquer sur des éléments d'interface pour effectuer des actions. Mais il existe une autre manière d'interagir avec un ordinateur, très utilisée en informatique : le mode texte ou **interface en ligne de commande** (CLI en anglais).

Ce mode d'interaction est très utile pour pouvoir **utiliser un ordinateur à distance**, ce qui est généralement le cas lorsque l'on souhaite administrer un serveur web.

À travers ce module vous allez découvrir le **shell** qui permet de dialoguer avec le système d'exploitation d'une machine. Il existe beaucoup de shells, le plus connu vient du monde Linux et se nomme bash. Vous allez découvrir les commandes de base pour parcourir des répertoires ou éditer des fichiers.

Terminal ou CLI

Az Définition

Un **terminal**, ou **interface en ligne de commande** (CLI en anglais), est une interface homme-machine dans laquelle l'utilisateur interagit avec la machine en mode **texte**. L'utilisateur écrit des **lignes de commande**, la machine les exécute et affiche le résultat des commandes.

Le terme terminal est très général : un terminal peut servir à dialoguer avec un programme informatique, à donner des ordres à un ordinateur, ou à exécuter d'autres programmes.

Shell

Az Définition

Un **shell** est une interface en ligne de commande permettant de dialoguer avec le système d'exploitation de la machine. On dit qu'il **interprète** les commandes.

Langages CLI

👁 Exemple

Il existe au moins autant de shells que de systèmes d'exploitation. Chaque shell propose des commandes spécifiques. Le choix d'un shell se fait surtout par rapport à des critères pratiques (de quoi ai-je besoin ?) et des critères subjectifs (quel shell me semble le plus ergonomique ?).



- Sur Linux, le shell le plus connu et installé par défaut sur la plupart des systèmes s'appelle bash (pour "Bourne Again Shell").
- Sur Windows, il existe trois shells :
 - cmd, le shell historique,
 - powershell, une version plus moderne,
 - et il est possible d'utiliser bash avec le sous-système Linux.
- Sur macOS, le shell installé par défaut s'appelle zsh. Il partage de grandes similarités avec bash.

Shells et Windows

Remarque

Le shell historique cmd de Windows n'est presque plus utilisé. powershell utilise une syntaxe assez différente de la plupart des autres shells. Il est possible d'installer bash sous Windows depuis peu, mais il n'est pas inclus par défaut dans les installations.

Bash et Windows

Conseil

La plupart des shells adhèrent à des standards communs, mais ce n'est pas le cas des shells disponibles sur Windows. Pour suivre les exemples et exercices qui suivent, il est recommandé **d'activer le sous-système Linux pour Windows 10** et d'utiliser bash, en suivant la partie "Installer le sous-système Windows pour Linux" de cette documentation : docs.microsoft.com/fr-fr/windows/wsl/install-win10

Commandes et Bash

Exemple

```
Terminal - stc@hal9017: /tmp/test
Fichier  Édition  Affichage  Terminal  Onglets  Aide
stc@hal9017:~$ cd /tmp/test/
stc@hal9017:/tmp/test$ pwd
/tmp/test
stc@hal9017:/tmp/test$ ls -al
total 60
drwxrwxr-x  2 stc  stc   4096 mai   2 14:09 .
drwxrwxrwt 21 root root 49152 mai   2 14:10 ..
-rw-rw-r--  1 stc  stc    97 mai   2 14:09 memo
stc@hal9017:/tmp/test$ cat memo
### Documentation
man

### Système de fichier
pwd
cd
ls
mkdir
rm
cat


### Éditeurs
nano
stc@hal9017:/tmp/test$
```

Cette image montre quelques commandes de base, exécutées par un shell bash sur un système Linux.



- `cd /tmp/test` se rend dans le dossier `test` qui se trouve dans le dossier racine `/tmp`.
- `pwd` affiche le dossier où on est situé dans le système de fichier.
- `ls -al` affiche la liste des fichiers dans le dossier courant.
- `cat memo` affiche le contenu du fichier `memo`.

Terminal et environnement graphique

 Remarque

- Un ordinateur personnel moderne PC dispose d'une interface graphique et d'une interface terminal : les deux permettent d'effectuer à peu près les mêmes opérations : visualiser des fichiers, les supprimer, ouvrir des applications, etc.
- Un serveur n'offre en général qu'une possibilité d'accès à distance via un terminal. C'est une des raisons pour lesquelles savoir utiliser le terminal est utile.

Ouvrir un terminal sous Linux

 Méthode

En général (et en particulier sur les systèmes Ubuntu), le raccourci `Ctrl+Alt+T` ouvre un terminal. Une alternative consiste à chercher `Terminal` dans la liste des applications.

Ouvrir un terminal sous Windows 10

 Méthode

Ouvrir la fenêtre `Exécuter` à l'aide du raccourci `Super+R` (la touche `Super` est en général représentée par un logo Windows sur le clavier). Entrer :


- `cmd` ou `powershell` dans la fenêtre qui s'est ouverte pour démarrer un terminal Windows
- `bash` pour ouvrir un shell Bash s'il a été installé

Ouvrir un terminal sous macOS

 Méthode

Depuis le `Launchpad`, chercher `Terminal` et cliquer sur l'icône qui s'affiche.

Quelques commandes de base

 Syntaxe

Ces commandes sont des commandes Bash qui fonctionnent également sous macOS et avec la plupart des autres shells Linux.

- `ls -al` : lister les fichiers dans le répertoire courant
- `pwd` : afficher le répertoire courant
- `cat fichier` : afficher le contenu d'un fichier
- `cd dossier` : se rendre dans un dossier fils du dossier courant
- `cd ..` : se rendre dans le dossier parent



- `echo message` : afficher un message
- `man commande` : afficher la documentation détaillée d'une commande

Copier/coller dans un terminal

 Méthode

Les actions « copier » et « coller » sont accessibles en effectuant un clic droit sur la fenêtre d'un terminal. Il est plus rapide d'utiliser des raccourcis clavier pour copier et coller, respectivement :

- Sur Bash et la plupart des shells Linux : `Ctrl+Shift+C` et `Ctrl+Shift+V`
- Sur macOS : `Command+C` et `Command+V`
- Sur Windows (powershell) : `Ctrl+C` et `Ctrl+V`

À retenir

- Un shell, ou par abus de langage un terminal, permet de dialoguer avec le système d'exploitation d'une machine.
- Il existe beaucoup de shells. Le plus connu vient du monde Linux et se nomme `bash`.
- Un shell permet de se passer du mode graphique, ce qui est souvent indispensable pour travailler sur une machine à distance.

2. Serveurs web

Serveur web


 Définition

Un serveur web est un logiciel qui **traite les requêtes HTTP** de clients.

Il est installé sur une machine hébergeant des documents (HTML, CSS, JavaScript, etc.) et est accessible depuis Internet ou un réseau local (un intranet).

Les ressources servies par le serveur peuvent être **statiques**, c'est-à-dire sans qu'elles existent préalablement à la requête (un fichier HTML, une image PNG, etc.) ou **dynamiques**, c'est-à-dire construites à chaque requête faite au serveur.

Serveur Web et serveur HTTP

 Remarque

Les serveurs web utilisent presque exclusivement le protocole HTTP, créé spécifiquement pour le Web. On parle par abus de langage de serveur web, mais le nom **serveur HTTP** est aussi possible.

Serveurs HTTP

 Exemple

- **Apache** : le serveur web le plus utilisé avec 44,3% des parts de marché en février 2019.
- **Nginx** : le deuxième serveur web le plus utilisé mais est le premier parmi les 1000 sites les plus actifs.



- **Node.js** : un environnement d'exécution JavaScript en dehors du navigateur qui fait aussi office de serveur HTTP.

À retenir

- Un serveur hébergeant des fichiers web a besoin d'un serveur web pour les rendre disponibles à travers le protocole HTTP.
- Apache et Nginx sont les serveurs web les plus utilisés.

3. Navigateurs web

Navigateurs web

Az Définition

Un navigateur web est client HTTP qui a trois rôles.

1. Effectuer des requêtes HTTP (GET, POST, etc.).
2. Comprendre les réponses à ces requêtes.
3. Interpréter les fichiers web (HTML, CSS, JavaScript) retournés.

Remarque

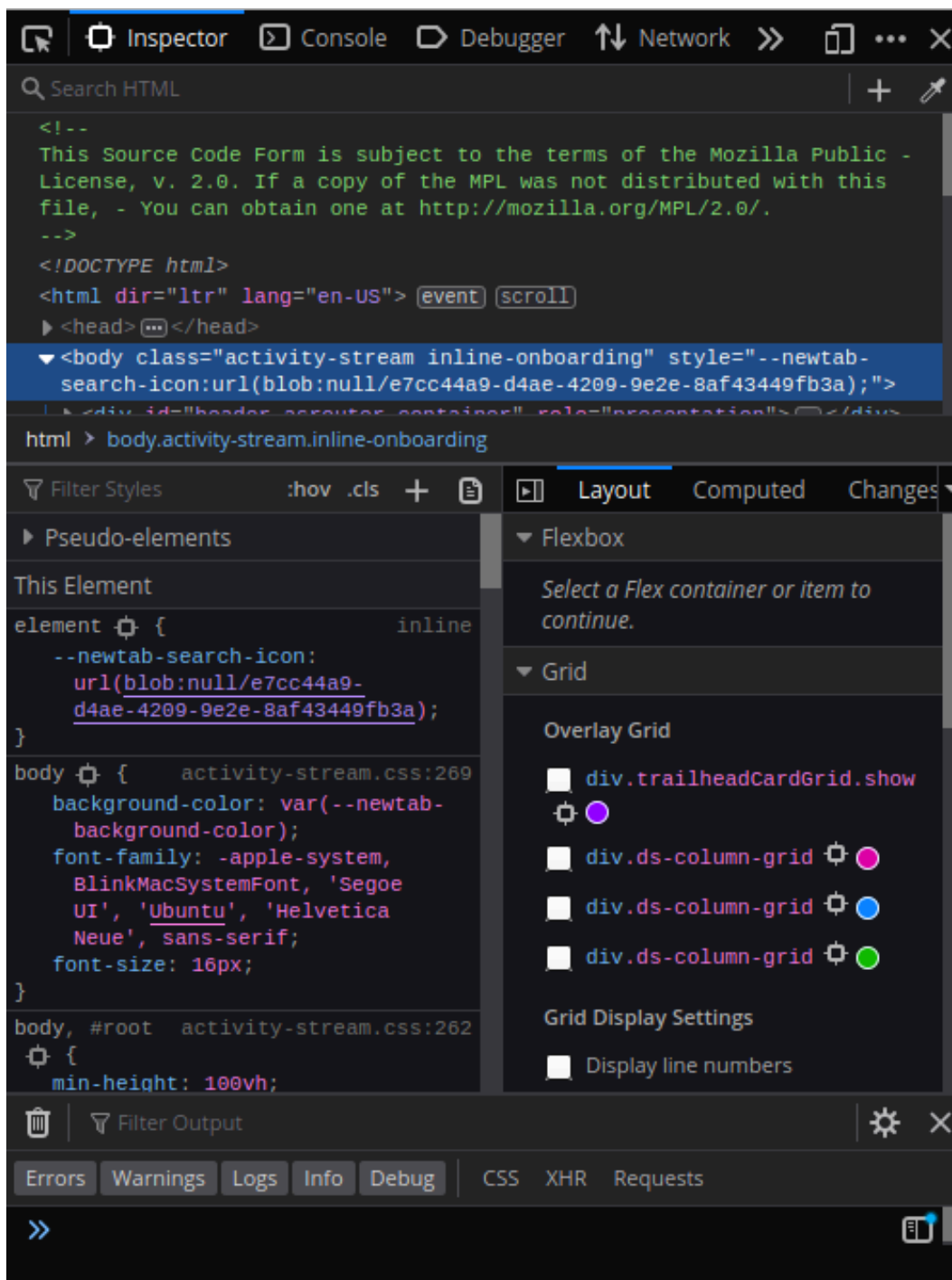
Tous les navigateurs web possèdent un interpréteur JavaScript permettant d'exécuter ce langage sur l'ordinateur client.

Les outils de développement sous Firefox

Fondamental

Les navigateurs possèdent aussi une partie « Outils de développement » destinée à aider les développeurs web dans leur activité.

Sous Firefox, ces outils sont disponibles en appuyant sur F12 ou clic droit > Inspecter l'élément.



Outils de développement Firefox (F12)

Voilà à quoi ressemble l'inspecteur depuis le site mozilla.org.

On retrouve plusieurs onglets en haut de la fenêtre. En voici les plus importants :

- L'inspecteur :
C'est celui que l'on voit sur l'image ci-dessus, on y retrouve l'HTML de la page et son CSS. Ils sont modifiables depuis cet onglet.
- La console :
C'est une console JavaScript. Un utilisateur peut donc y écrire des lignes de JavaScript qui seront interprétées à la volée.
- Le déboguer :
Cet outil permet un suivi très fin de ce qu'il se passe pendant l'interprétation de la page et l'exécution de JavaScript.

- Le moniteur réseau :

Cet outil permet d'analyser avec une précision fine toutes les requêtes envoyées et leur réponse associée.

St...	M...	Domain	File	Cause	Ty...	Transfer...	Size
304	GET	www....	/en-US/	document	ht...	cached	17...
200	GET	www....	site.133e404d326d.js	script	js	cached	0 B
200	GET	www....	protocol-core.9e72b5c2515...	stylesheet	css	cached	57...
200	GET	www....	home-2018.00d7b2b54843....	stylesheet	css	cached	29...
200	GET	www....	gtm-snippet.9f9cf2026c5f.js	script	js	cached	51...
200	GET	www....	common-protocol.0cfc444f...	script	js	cached	0 B
200	GET	www....	stub-attribution.4e24eb9b8...	script	js	cached	0 B
200	GET	www....	home.c5483f81f59a.js	script	js	cached	0 B
304	GET	www....	gtm.js?id=GTM-MW3R8V&l=...	script	js	cached	0 B
200	GET	www....	logo-sm.f2523d97cbe0.png	img	png	cached	2....
200	GET	www....	placeholder.71a50dbba44c...	img	png	cached	95...
200	GET	www....	logo-sm.d3157a6ac671.png	img	png	cached	2....
200	GET	www....	logo-sm.d1b49e50ffb7.png	img	png	cached	2....
200	GET	www....	logo-sm.751c5555e455.png	img	png	cached	2....
200	GET	www....	logo-word-hor.96f28a0f9ae...	img	svg	cached	9....
200	GET	www....	logo-word-hor-sm.5622edb...	imageset	png	cached	5....
200	GET	www....	billboard-more-power.f83d...	imageset	png	cached	25...
200	GET	www....	billboard-healthy-internet.4...	imageset	png	cached	25...

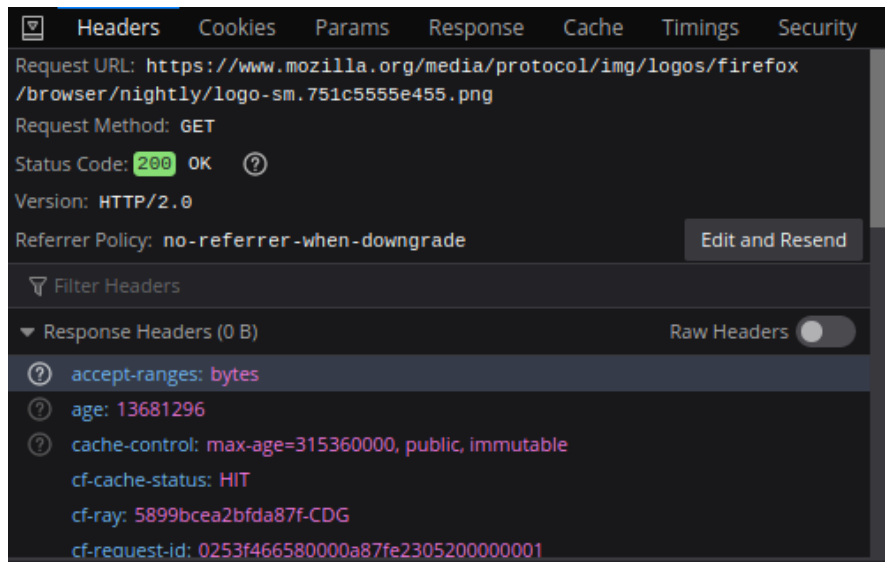
33 requests | 492.21 KB / 867 B transferred | Finish: 1.48 s | DOMContentLoaded: 69...

Extrait du moniteur réseau quand on accède à www.mozilla.org

Il y a pour chaque requête :

- le code HTTP de la réponse (304 pour signifier que le document demandé n'a pas été modifié depuis la dernière demande et 200 pour dire que tout va bien),
- la requête complète, ici il s'agit d'accéder à la page d'accueil, il n'y a donc que des GET pour récupérer le HTML, le CSS, les script JavaScript et les images,
- l'adresse du serveur web (www.mozilla.org),
- la ressource demandée,
- le type du fichier (HTML, CSS, PNG, etc),
- la quantité transférée, cette valeur est à cached lorsque que le navigateur l'avait déjà en mémoire,
- la taille de la ressource.

En cliquant sur une requête, toutes ses informations apparaissent.



Outils de développement Firefox (onglet requête)

À retenir

- Les navigateurs web sont des clients HTTP.
- Ils interprètent les langages du Web (HTML, CSS, JS).
- Ils disposent d'outils de développement permettant d'inspecter les fichiers des pages web et les requêtes et réponses HTTP.