INSA INSTITUT NATIONAL DES SCIENCES APPLIQUÉES HAUTS-DE-FRANCE

Services Couche 7

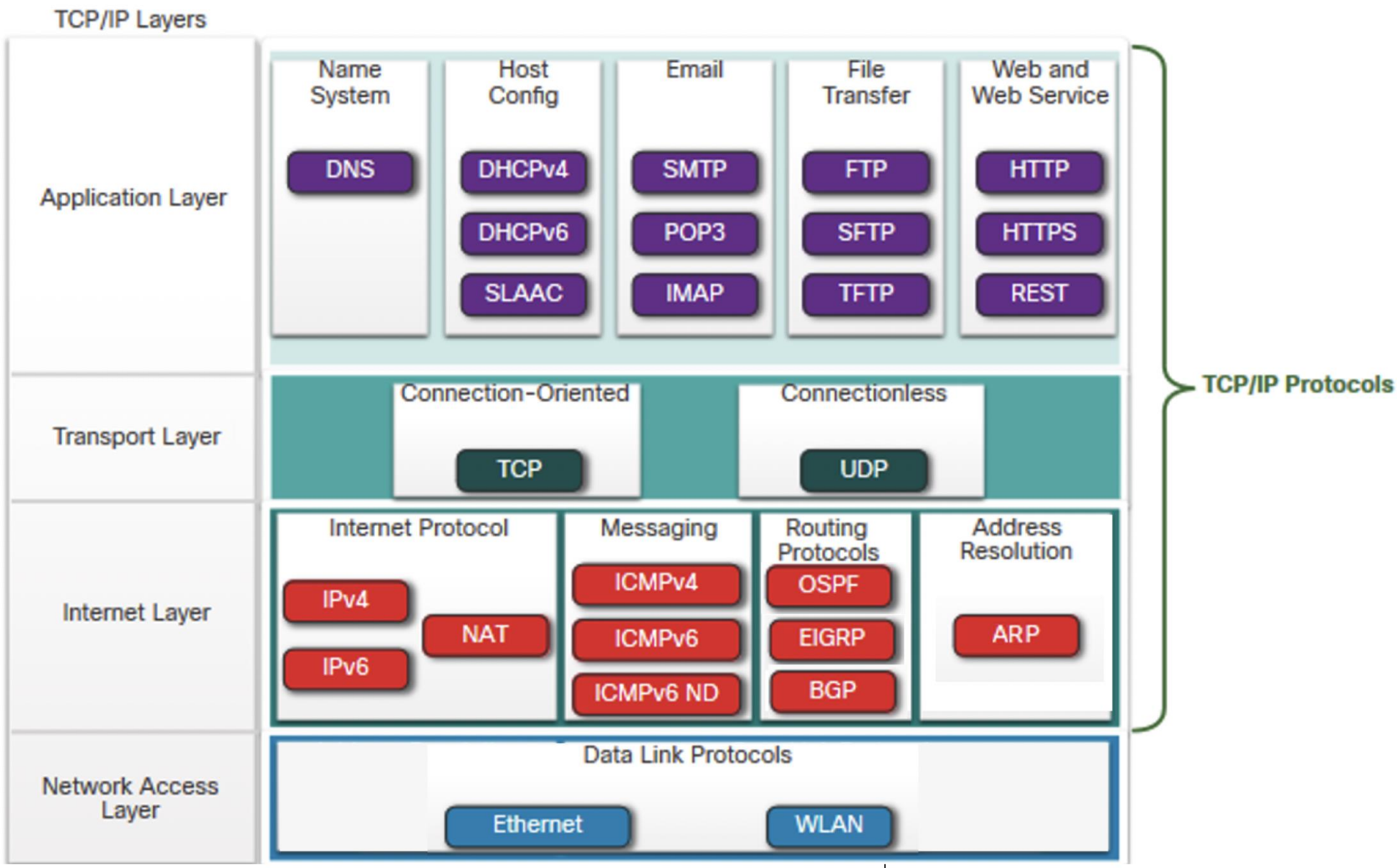


Figure 4 : les protocoles TCP-IP

La couche application

- Aussi appelée : *Services TCP/IP*

- Trois grandes familles
  - Configuration automatique
  - Service d'application
    - Nom de domaine
    - Courrier
    - Accès distant
    - Transfert de fichiers
    - Accès fichier
    - Accès web
  - Gestion de réseau TCP/IP

List non exhaustifs de services

- http
- Smtp
- Pop3
- Imap4
- Dns
- Dhcp
- telnet
- ftp
- Tftp
- R*
- Nfs
- Snmp

HTTP

HTTP transmet des données entre le client et le serveur

Le **client** envoie une **demande HTTP** au serveur

- Méthode HTTP
- En-têtes HTTP
- Corps de la demande

Le **serveur** envoie une **réponse HTTP** au client

- Code d'état HTTP
- En-têtes HTTP
- Corps de la réponse

La connexion est **standardisée** et suit un **format précis** pour les demandes et les réponses HTTP.

HTTP : Schéma de fonctionnement

Navigateur Web

Requête HTTP :

GET ....

1

3

Réponse :

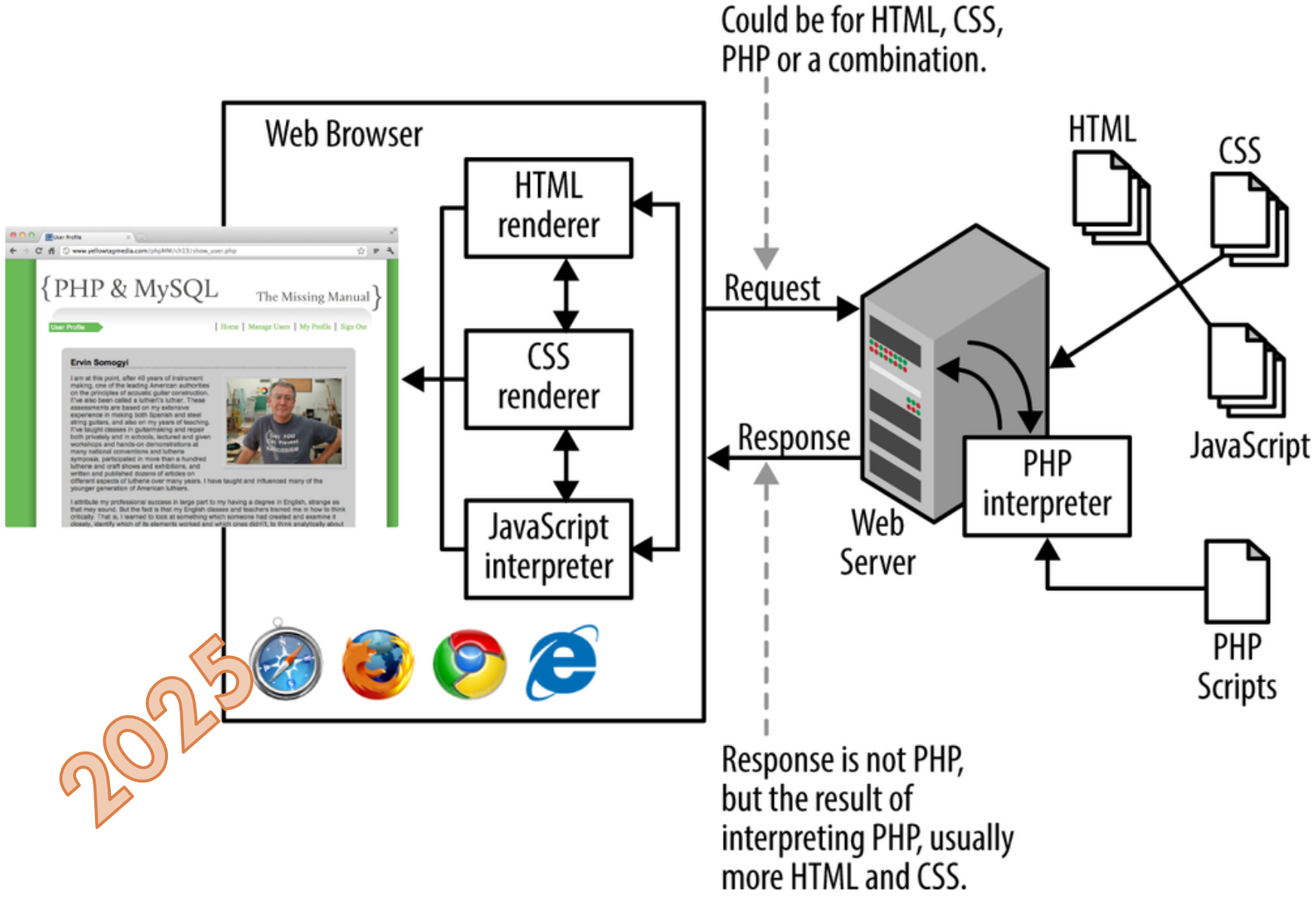
code HTML CSS ...

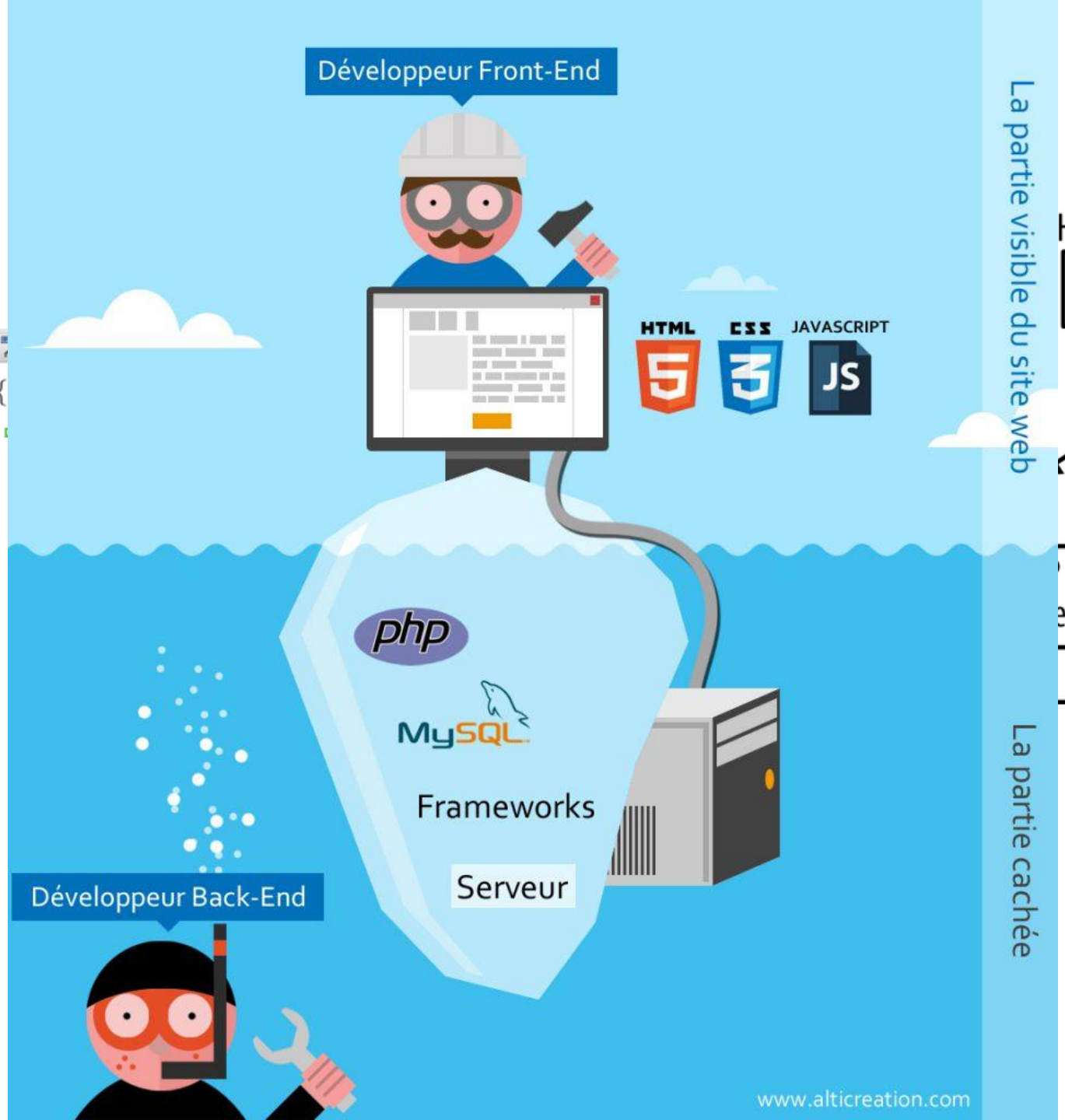
Serveur Web

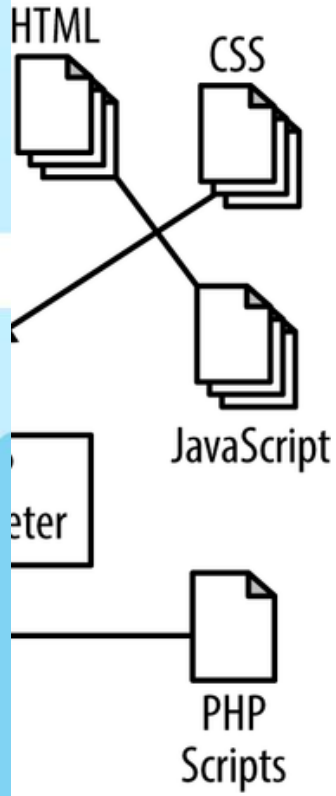
2

Lecture du fichier

2000


Could be for HTML, CSS, PHP or a combination.
Web Browser
HTML renderer
CSS renderer
JavaScript interpreter
{PHP & MySQL The Missing Manual}
Request
Response
Web Server
PHP interpreter
HTML
CSS
JavaScript
PHP Scripts
2025
Response is not PHP, but the result of interpreting PHP, usually more HTML and CSS.


Développeur Front-End
HTML
CSS
JAVASCRIPT
La partie visible du site web
php
MySQL
Frameworks
Serveur
Développeur Back-End
La partie cachée
www.alticreation.com


HTML
CSS
JavaScript
PHP
Scripts

Les méthodes HTTP (GET, POST, PUT, DELETE, etc.) et leur utilisation

- Les méthodes HTTP (GET, POST, PUT, DELETE, etc.) définissent la façon dont une demande HTTP est traitée par le serveur. Les méthodes HTTP les plus couramment utilisées sont :

  - **GET** : Utilisé pour obtenir des données à partir du serveur.
    - Par exemple, lorsque vous chargez une page web dans votre navigateur, le navigateur envoie une demande GET au serveur pour obtenir le HTML de la page.

  - **POST** : Utilisé pour envoyer des données au serveur.
    - Par exemple, lorsque vous remplissez un formulaire en ligne, le navigateur envoie une demande POST au serveur pour envoyer les données du formulaire.

Les méthodes HTTP (GET, POST, PUT, DELETE, etc.) et leur utilisation

- **PUT** : Utilisé pour mettre à jour des données sur le serveur. Par exemple, une application peut envoyer une demande PUT au serveur pour mettre à jour les informations de profil d'un utilisateur.

- **DELETE** : Utilisé pour supprimer des données du serveur. Par exemple, une application peut envoyer une demande DELETE au serveur pour supprimer un enregistrement.

- Il existe également d'autres méthodes HTTP, telles que **HEAD**, **OPTIONS**, et **CONNECT**, qui sont utilisées pour des tâches spécifiques telles que la vérification de la disponibilité d'une ressource ou la mise en place d'une connexion sécurisée.

Les codes d'état HTTP (200 OK, 404 Not Found, etc.)

- Chaque réponse HTTP inclut un code d'état qui décrit la situation de la demande
- Les plus utilisées :
  - **200 OK** : Indique que la demande a réussi et que le serveur a envoyé les données demandées.
  - **404 Not Found** : Indique que le serveur n'a pas pu trouver la ressource demandée.
  - **500 Internal Server Error** : Indique qu'une erreur interne s'est produite sur le serveur.
  - **403 Forbidden** : Indique que le serveur refuse de fournir la ressource demandée en raison de restrictions d'accès.
  - **401 Unauthorized** : Indique que l'authentification est nécessaire pour accéder à la ressource demandée.

Dans le RFC

En général, les requêtes et les réponses HTTP incluent les éléments suivants :

1. **Ligne de demande ou de réponse :**
   Comprend des informations telles que la méthode HTTP, l'URL de la ressource demandée, et la version HTTP.
   Par exemple, la ligne de demande d'une requête GET pour une page web pourrait ressembler à ceci :
   "GET /index.html HTTP/1.1".

2. **Une ligne vide**

3. **En-têtes HTTP :**
   Incluent des informations supplémentaires telles que le type de contenu souhaité, l'authentification, la date, etc.

4. **Corps de la demande ou de la réponse :**
   Peut inclure des données telles que des données de formulaire pour une demande POST ou du HTML pour une réponse.

Exemple : entête de la demande

Soit la requête vers le site www.tomczak.fr

i. **En-tête de la demande**

Contenu	Paramètre et signification
GET / HTTP/2	Méthode GET
Host: www.tomczak.fr	**Host** désigne le nom de domaine
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/109.0	**User-Agent**: identifie l'application qui effectue la requête
Accept: text/html, application/xhtml+xml, application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8	**Accept**: La page renvoyée doit être du texte, de HTML image ….
Accept-Language: fr,fr-FR;q=0.8,en-US;q=0.5,en;q=0.3	**Accept-Language** : les langues acceptées
Accept-Encoding: gzip, deflate, br	**Accept-Encoding**: précise les compressions que le client peut proposer au serveur
Connection: keep-alive	**Connection**: utilisation d'une seule connexion TCP/IP pour envoyer ou recevoir plusieurs demandes/réponses
	…

ii. **Corps de la requête**

Vide

Entête de la réponse

iii. Entête de la réponse :

Contenu	Paramètre et signification
HTTP/2	Version de l'HTML
200 OK	C'est le paramètre **Etat** : la requête est un succès la page demandée a été trouvée sur le serveur
date: Sun, 12 Feb 2023 10:46:02 GMT	Date et heure d'origine du message
content-type: text/html; charset=UTF-8	La page web est écrite en HTML avec l'encodage utf-8 pour les caractères
server: Apache	Le type de serveur qui est répondu est Apache
x-powered-by: PHP/7.4	Version du PHP
link: <https://www.tomczak.fr/>; rel=shortlink	Lien du site
…	

Corps de la réponse

iv. Corps de la réponse

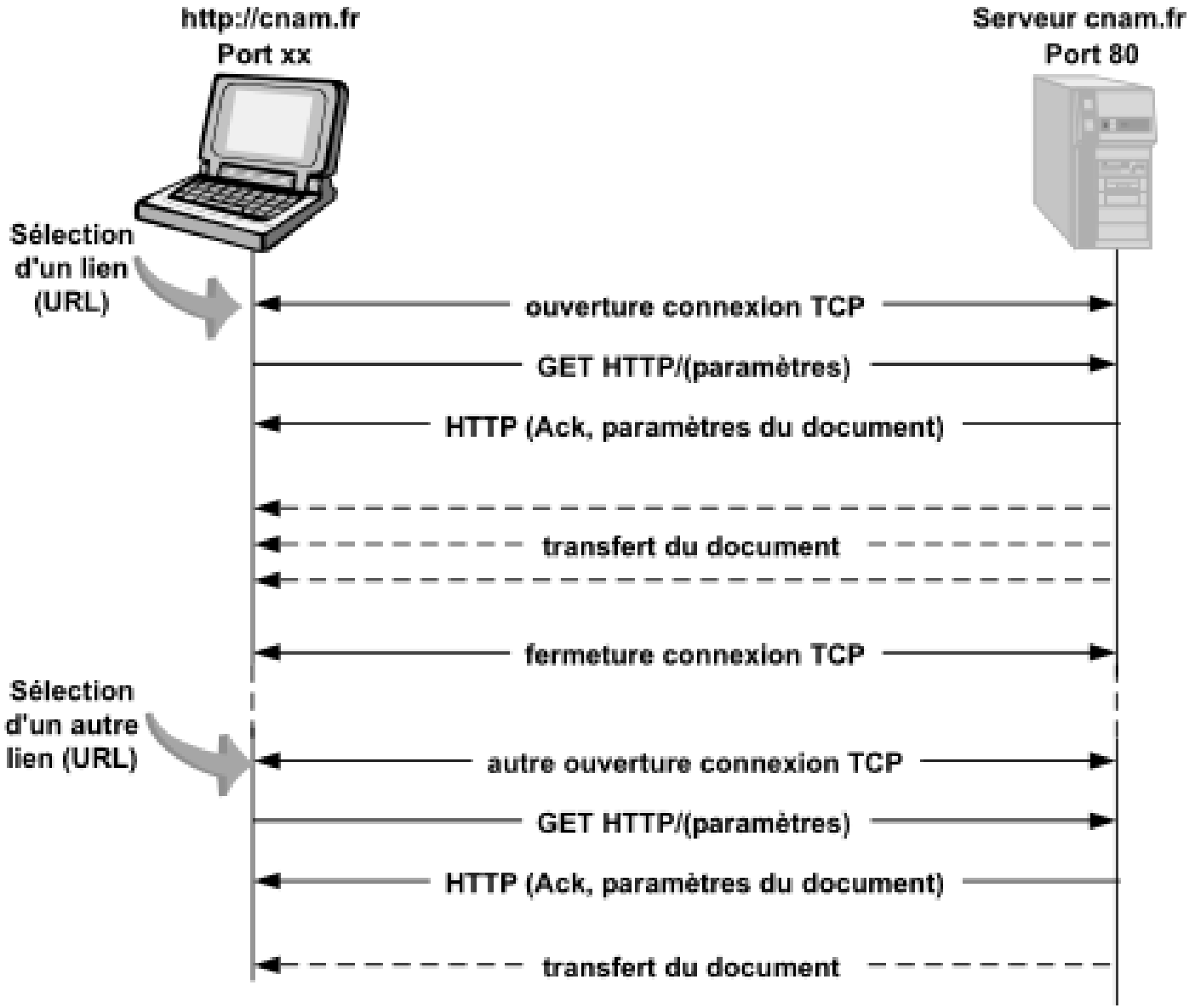
C'est tout simplement le code HTML de la page web :

```
<!DOCTYPE html> <html lang="fr-FR"><head>
            <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="profile" href="http://gmpg.org/xfn/11">
    <link rel="pingback" href="https://www.tomczak.fr/xmlrpc.php">

<title>Site professionel de Robert Tomczak &#8211; Cours/TP de BTS SN-EC et
autres</title>
<meta name='robots' content='max-image-preview:large' />
<link rel='dns-prefetch' href='//www.google.com' />
<link rel='dns-prefetch' href='//fonts.googleapis.com' />


…
```

Les échanges entre un navigateur et un serveur HTTP



La trame brute

```
0000  00 30 84 89 21 0d 00 0e  9b 8f fc b6 08 00 45 00   .0..!... ......E.
0010  01 d3 38 00 40 00 80 06  7e f4 c0 a8 6d f2 d4 1b   ..8.@... ~...m...
0020  3f 7a 05 6e 00 50 b7 48  68 48 ce 71 ca 5b 50 18   ?z.n.P.H hH.q.[P.
0030  44 70 fe 89 00 00 47 45  54 20 2f 64 68 63 70 2f   Dp....GE T /dhcp/
0040  5f 74 68 65 6d 65 73 2f  74 68 65 6d 65 2e 63 73   _themes/ theme.cs
0050  73 20 48 54 54 50 2f 31  2e 31 0d 0a 41 63 63 65   s HTTP/1 .1..Acce
0060  70 74 3a 20 2a 2f 2a 0d  0a 52 65 66 65 72 65 72   pt: */*. .Referer
0070  3a 20 68 74 74 70 3a 2f  2f 63 68 72 69 73 74 69   : http:/ /christi
0080  61 6e 2e 63 61 6c 65 63  61 2e 66 72 65 65 2e 66   an.calec a.free.f
0090  72 2f 64 68 63 70 2f 61  6e 61 6c 79 73 65 5f 64   r/dhcp/a nalyse_d
00a0  65 5f 74 72 61 6d 65 73  2e 68 74 6d 0d 0a 41 63   e_trames .htm..Ac
00b0  63 65 70 74 2d 4c 61 6e  67 75 61 67 65 3a 20 66   cept-Lan guage: f
```

Comment HTTP gère la sécurité des transactions Web

- HTTP ne fournit pas de sécurité pour les transactions Web
- Utilisation de SSL/TLS pour le chiffrement des données
- Authentification du serveur pour éviter les attaques.

Utilisation de SSL/TLS pour le chiffrement des données

- SSL/TLS : Chiffrement des données entre client et serveur
- Utilisation d'une clé publique et d'une clé privée
- Certificat digital pour l'authentification du serveur
- Vérification du certificat par le client
- Génération d'une clé secrète pour le chiffrement des données
- Les données sont chiffrées et transmises au serveur.

Conclusion

Résumé des points clés sur HTTP

- 1. **Les méthodes HTTP** : HTTP utilise plusieurs méthodes, telles que GET, POST, PUT, DELETE, etc., pour effectuer différentes opérations sur les données.
- 2. **Codes d'état** HTTP : HTTP utilise des codes d'état pour informer le client sur le statut de la requête. Les codes d'état les plus couramment utilisés sont 200 OK, 404 Not Found, etc.
- 3. **Structure des requêtes et des réponses** HTTP : Les requêtes et les réponses HTTP sont structurées de manière standardisée et incluent des en-têtes HTTP pour fournir des informations supplémentaires.

Résumé des points clés sur HTTP

- 4. **En-têtes HTTP** : Les en-têtes HTTP fournissent des informations supplémentaires sur les requêtes/réponses HTTP, telles que le type de contenu, l'autorisation, la date, le référer, etc.
- 5. **Session HTTP** : Une session HTTP est une série de requêtes HTTP et de réponses associées qui sont liées entre elles. Les sessions peuvent être utilisées pour suivre les actions d'un utilisateur sur un site web, pour stocker des informations sur l'état de la connexion, ou pour stocker des données entre plusieurs requêtes.

DHCP

(Dynamic Host Configuration Protocol

- Configuration automatique dès le démarrage de la machine

- DHCP centralise et gère l'allocation des informations de configuration TCP/IP :
  - Adresse IP
  - Masque de sous-réseaux
  - Passerelle par défaut
  - Serveur de résolution de noms (DNS) : serveur de nom primaire et secondaire sous win
  - l'adresse des serveurs WINS (sous win)

Client Non DHCP

Client DHCP

@IP 1

@IP 2

Client DHCP

Base de données
DHCP
@IP 1 192.168.109.101 : attribuée
@IP 2 192.168.109.102 : attribuée
@IP 3 : 192.168.109.103 : libre

Serveur DHCP

Avantages du DHCP

- **Sa mise en place dans un réseau permet d'éliminer certains problèmes liés à la configuration manuelle :**
  -  **Interdit le conflit d'adresses**
  -  **Dès qu'un paramètre est modifié (serveur DNS), les modifications sont répercutées**
  -  **Centralise l'affectation :**
    Configuration par rapport d'@ MAC, avec possibilité d'interdire ou d'autoriser certaines machines
  -  **Économie d'@ IP :**
    Par exemple, une entreprise peut disposer de 500 machines mais n'en avoir que 150 sur le réseau en même temps
  -  **Les portables changeant de lieu** (WIFI) sont plus facilement gérable

 Définition (Bail DHCP)

- **Durée** pendant laquelle un poste peut utiliser la configuration IP
- À la fin du bail, **le renouvellement** est sollicité auprès du serveur DHCP
- Les **modifications** peuvent se faire à ce moment
- Lors du **retrait de la machine** du réseau, le bail est libéré

-  **Inconvénients**

- **DHCP utilise des trames de diffusion :** Le réseau d'une entreprise peut être surchargé en début de journée

Fonctionnement de DHCP : DORA

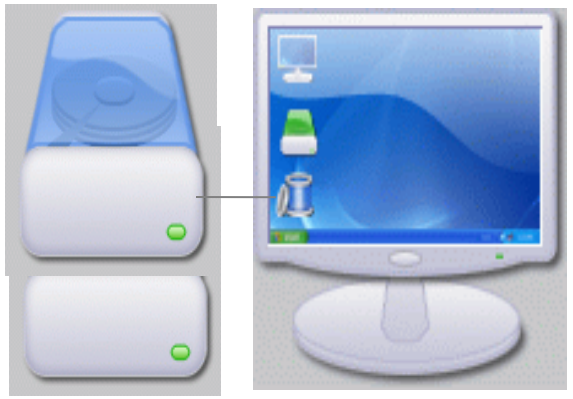
Client DHCP

1. Demande de bail

2. Offre de bail

3. Sélection de bail

4. Accusé de réception



Serveur DHCP

- DHCPDISCOVER
  - demande de bail
  - @ MAC source se trouve dans la trame
  - @ source : 0.0.0.0
  - @ destination : 255.255.255.255

- DHCPOFFER
  - offre de bail
  - @ MAC destination : celle du client
  - @ source : @ du serveur DHCP
  - @ destination : 255.255.255.255
  - @ IP offerte
  - Identificateur du serveur : @ IP du serveur
  - Durée du bail : 72 heures par ex

- DHCPREQUEST
  - sélection du bail et du serveur
  - @ MAC destination : celle du client
  - @ source : 0.0.0.0
  - @ destination : 255.255.255.255
  - @ IP acceptée
  - Identificateur du serveur : @ IP du serveur

- DHCPACK
  - Accusé de réception
  - @ MAC destination : celle du client
  - @ source : @ IP du serveur
  - @ destination : 255.255.255.255
  - Masque de sous-réseau =255.255.255.0
  - Tous les paramètres nécessaires pour la configuration IP

A retenir : DORA

- DHCP **D**ISCOVER
- DHCP **O**FFER
- DHCP **R**EQUEST
- DHCP **A**CK

- **D**ISCOVER
- **O**FFER
- **R**EQUEST
- **A**CK

DORA

🏷️ Agent de relais DHCP

- Les trames sont envoyées par **diffusion,** or un **routeur** ne retransmet pas ce type de trames.

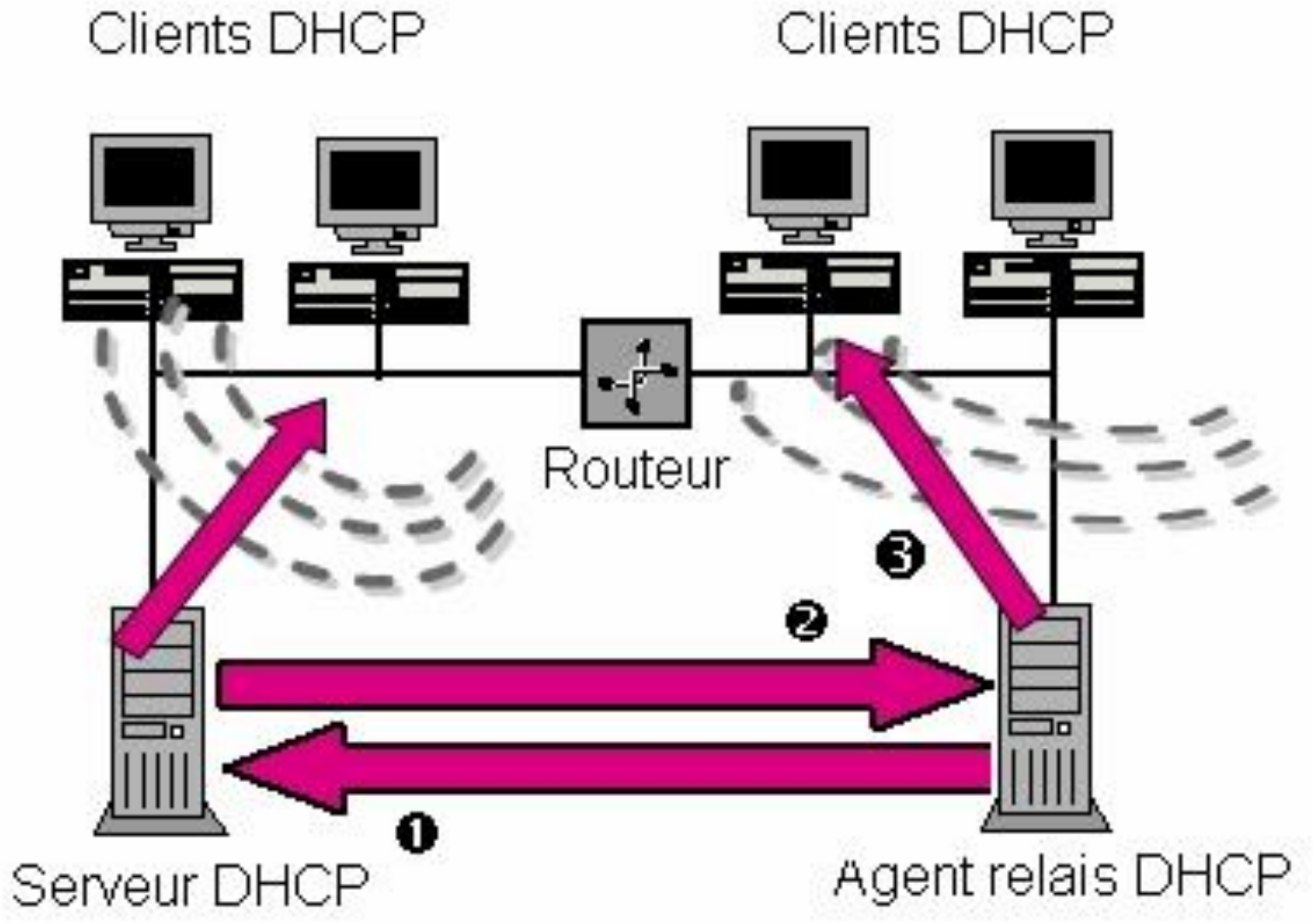
🚀 Solutions

1. **Un serveur DHCP par réseau** et sous-réseaux
2. **Agent de relais**

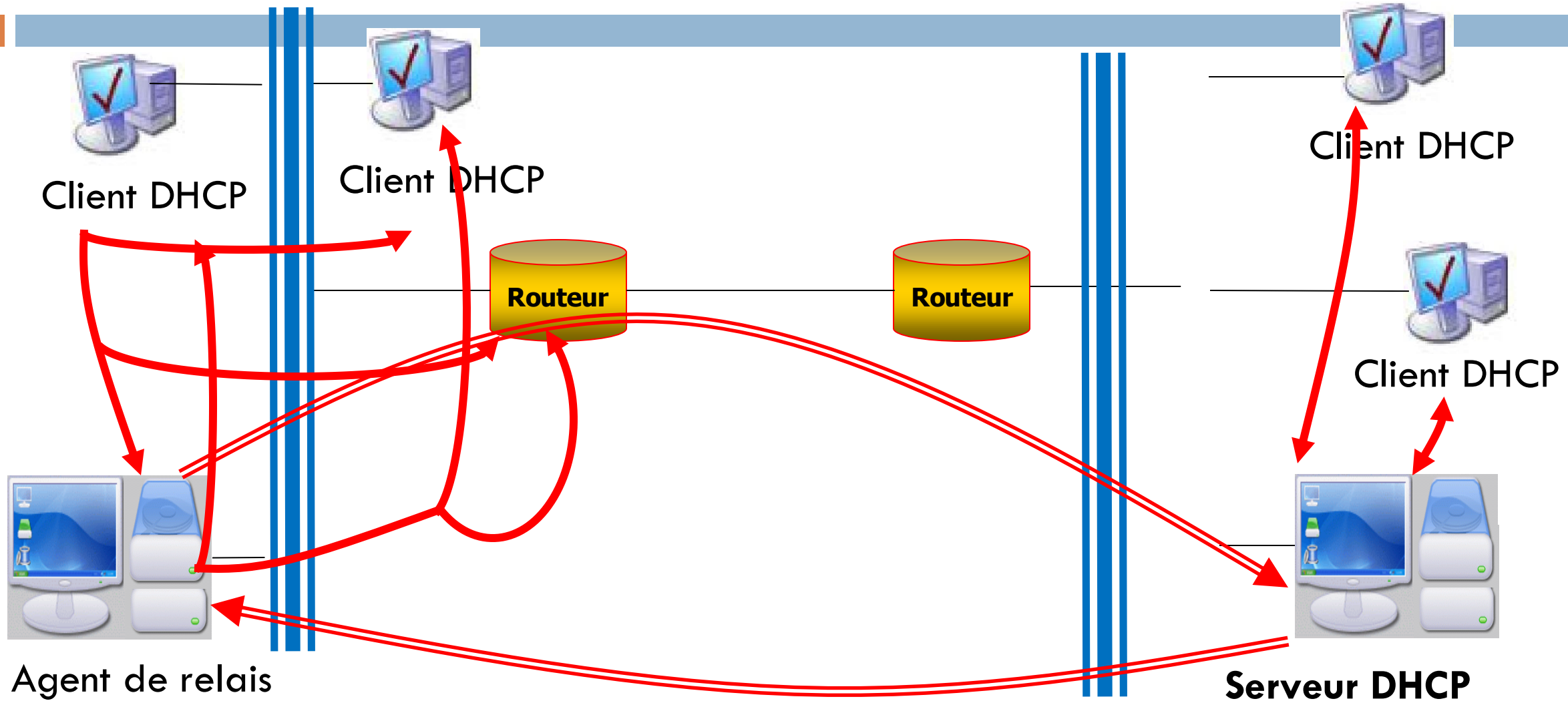
⚙️ Fonctionnement de l'agent de relais

- **Agent de relais** = Fonction activée sur votre routeur ou une machine serveur configurée
- Il va **relayer** (transmettre) les trames DHCP entre le client et le serveur dont il connaît l'@IP

C.F. animation page suivante



Relais DHCP


Client DHCP
Client DHCP
Routeur
Routeur
Client DHCP
Client DHCP
Agent de relais
Serveur DHCP

DNS

Résolution de noms de domaines

La couche application : nom de domaine

- 🌐 **DNS (Domain Name Service)**

- **Le DNS** permet d'adresser un hôte à l'aide d'un **nom** plutôt que par son **@ IP**

- Chaque fois qu'un des **services d'Internet** (HTTP, FTP, Telnet, etc.) fait référence à une machine par son nom,
  le **serveur de nom DNS** est **interrogé** (si l'enregistrement recherché n'est pas déjà dans le cache du client).

 Au début d'ARPANET

- Il n'y avait que **quelques centaines** de machines sur le réseau
- Un **fichier hosts.txt** suffisait à contenir les correspondances noms d'ordinateurs / @IP

 Centralisation

- Ce fichier était **centralisé** sur un ordinateur (SRI-NIC : *Stanford Research Institute's Network Information Center*)
  situé à Menlo Park en Californie

 Fréquence de mise à jour

- Le **téléchargement** se faisait **une ou deux fois par semaine**

```
102.54.94.97     rhino.acme.com          # serveur source
38.25.63.10      x.acme.com              # hôte client x
…
127.0.0.1        localhost
```

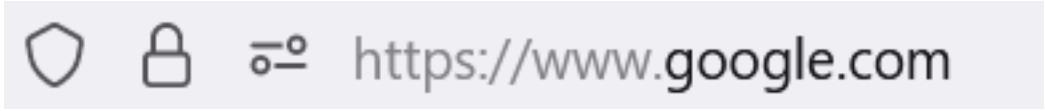
⚠ **Problèmes apparus**

- **Fichier hosts.txt** devenu trop volumineux
- Besoin de **mises à jour** plusieurs fois par jour
- Trop de **trafic**
- …

✅ **Solution**

- **Base de données** répartie et synchronisée sur plusieurs serveurs, appelés **serveurs DNS**

Exemple N°1

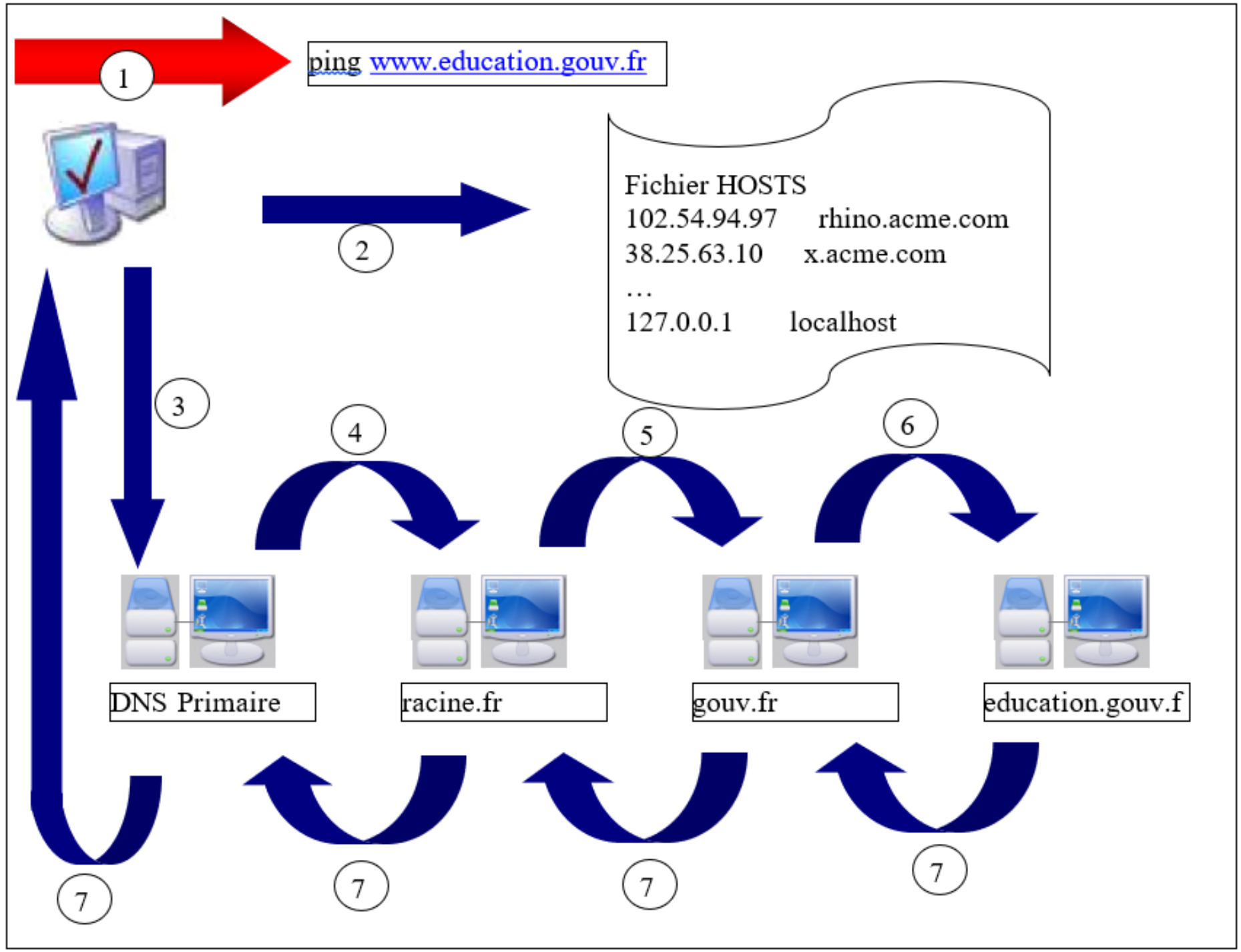


- 1. **Requête DNS** : Votre navigateur envoie une requête à un serveur DNS pour demander l'adresse IP associée au nom de domaine `www.google.com`.

- 2. **Résolution de nom** : Le serveur DNS recherche dans sa base de données pour trouver l'adresse IP correspondant à `www.google.com`. Si le serveur DNS ne connaît pas l'adresse, il demandera à d'autres serveurs DNS jusqu'à ce qu'il trouve la réponse.

- 3. **Réponse DNS** : Une fois que le serveur DNS a l'adresse IP, il la renvoie à votre navigateur.

- 4. **Connexion** : Votre navigateur utilise cette adresse IP pour établir une connexion avec le serveur web de Google et afficher le site.


1
ping www.education.gouv.fr
Fichier HOSTS
102.54.94.97 rhino.acme.com
38.25.63.10 x.acme.com
…
127.0.0.1 localhost
2
3
4
5
6
DNS Primaire
racine.fr
gouv.fr
education.gouv.f
7
7
7
7

Etapes de la résolution de nom distante

1. Le client consulte son cache de correspondance @symbolique / @IP, si succès arrêt
2. Interrogation du serveur DNS primaire configurée chez le client. Celui-ci consulte sa table/cache, s'il ne possède pas la correspondance, il transmet la demande à un serveur racine
3. Le serveur racine.fr consulte sa table et transmet la demande si nécessaire
4. Le serveur gouv.fr est interrogé
5. Le serveur education.gouv.fr est interrogé, il est en mesure de donner @IP
6. @IP est relayée de manière récursive jusqu'au client et chaque serveur l'inscrit dans sa table

- Puisque ce système fonctionne sur le principe requête/réponse, c'est UDP qui soit le mieux adapté.
- Ainsi on ne perd pas de temps à maintenir une connexion
- TCP peut être utilisé mais le coût d'établissement et de fermeture d'une connexion ne se justifie pas pour un seul échange requête/réponse

- TCP est utilisé si UDP ne fonctionne pas

Le fichier host

- Il est quelques fois utilisé :
  - Pour les réseaux de quelques postes
  - Bloquer certains sites (publicités, sites de charmes …)
    - **127.0.0.1 ad.searchsquire.com**
- Le processus de résolution de nom consulte *host* puis interroge un serveur DNS

- Se trouve :
  - Linux : /etc/host
  - Win : C:\WINDOWS\system32\drivers\etc\host

Serveur DNS

- L'implémentation majoritaire de DNS est BIND initialement développé sous Linux
- BIND = Berkeley Internet Domain Name
- Démon appelé *named*

- WINS (Microsoft) n'est pas un serveur DNS. Il convertit une @IP en nom Netbios.
- Nom Netbios : 15 caractères maxi.

SMTP, POP3 & IMAP4

Les protocoles de courrier

2.4 La couche application : courrier

-  **SMTP (Simple Mail Transfer Protocol)** : Sert à l'envoi d'e-mails d'un client vers un serveur de messagerie, et aussi au transfert entre serveurs de messagerie.
-  **POP3 (Post Office Protocol, version 3)** : Permet à un client de récupérer les e-mails depuis le serveur. Par défaut, les e-mails sont souvent téléchargés et supprimés du serveur (selon la configuration).
-  **IMAP4 (Internet Message Access Protocol)** : Laisse les e-mails sur le serveur et propose une synchronisation en temps réel avec le client, permettant de consulter le même contenu sur plusieurs appareils.
- Simple, non ?

SMTP / ESMTP :

- **Protocole d'échange :**
  Utilisé pour l'envoi d'e-mails par les clients (mode « submission » sur le port 587) et pour l'interconnexion entre serveurs (port 25).

- **Commandes clés :**
  Les commandes **HELO/EHLO, MAIL FROM, RCPT TO** et **DATA** sont utilisées pour établir la communication et transférer le contenu du message.

- **Extensions ESMTP :**
  Apportent des fonctionnalités supplémentaires telles que l'**authentification** et le **chiffrement** avec **STARTTLS**.

- **Sécurisation :**
- **Sécurité accrue :**
  De plus en plus, les échanges entre serveurs sont sécurisés grâce à des mécanismes comme **STARTTLS**, qui chiffrent la session SMTP pour protéger les informations échangées

Spécificités de SMTP

- SMTP est un protocole textuel
- Le protocole spécifie :
  - Le format des adresses des utilisateurs suivant une notation internet classique
  - Les champs des en-têtes de courrier (from … to …)
  - Les possibilités d'envoi groupé (champs CC ou BCC)
  - La gestion des heures
  - Le codage pour les fichiers joints (attachés)
    - ASCII pur
    - Standard MIME pour un texte formaté, des images ou du son

Commandes d'envoi

- ***HELO exp*** : requête de connexion en provenance d'un expéditeur SMTP
- ***MAIL FROM ; adr_exp*** : @ de l'expéditeur, annonce le début de l'échange
- ***RCP TO :ad_dest*** : spécifie un destinataire
- ***DATA*** : les données sont reçus jusqu'à la réception de deux sauts consécutifs (CR LF CR LF)
- ***QUIT*** : demande au récepteur d'envoyer la réponse OK et de refermer la connexion

Commandes de réponse

- ***250*** : action demandée correctement effectuée (OK)
- ***354*** : le message peut être transmis
- ***451*** : action demandée annulée
- ***550*** : action non effectuée : boites aux lettres inaccessible

- Echange SMTP en 3 phases :

1. Etablissement de la connexion au niveau SMTP et identification de la source et de la destination
2. Envoi du message avec les en-têtes
3. Libération de la connexion

SPAM

- En théorie, le protocole SMTP de base ne vérifie pas systématiquement l'adresse de l'expéditeur, ce qui pourrait permettre à une personne malintentionnée d'envoyer des messages en masse.

- Cependant, dans la pratique moderne, plusieurs mesures ont été mises en place pour empêcher cela :

-  **Authentification obligatoire** : La plupart des serveurs SMTP exigent désormais une authentification (avec des identifiants) avant de permettre l'envoi d'e-mails. Cela empêche les utilisateurs non autorisés d'envoyer des messages via ces serveurs.
-  **Configuration d'open relay** : Les serveurs bien configurés désactivent les "open relays" (serveurs ouverts) qui, par défaut, pourraient accepter des e-mails de n'importe qui sans vérification.

SPAM

- .

- **Filtres anti-spam et politiques strictes** : Les serveurs intègrent des mécanismes de filtrage qui détectent et bloquent les activités suspectes, limitant ainsi l'envoi de spam.

- **Technologies complémentaires** : Des protocoles et standards complémentaires, comme le chiffrement TLS et les listes noires, contribuent à la sécurisation des échanges et à la protection contre l'usurpation d'identité.

FTP ET SES DÉRIVÉS

Protocoles de transfert de fichiers

2.4 La couche application : transfert de fichiers

- FTP (File Transfert Protocol)

- TFTP (Trivial File Transfert Protocol)
  - Utilise UDP
  - Donc :
    -
    -
    -
- Simple et compacte -> ROM d'amorçage

- FTP permet le transfert de fichiers et de répertoires entre un serveur et un client

- Ce protocole fonctionne avec 2 canaux (ports)
  - Le port TCP 20 : c'est le canal de données
  - Le port TCP 21 : le canal de commandes

- La connexion peut être anonyme (login : anonymous ou ftp, password @électronique) ou authentifiée (votre login/mot de passe sur le réseau)

Les commandes FTP

- Les commandes utilisateurs utilisées par un logiciel client sont traduites en commandes internes FTP

- user -> USER
- password -> PASS
- dir -> LIST
- get -> RETR
- …
- Elles sont suivies éventuellement d'arguments (nom d'utilisateur, mot de passe …) ou de données correspondant au fichier

Autres caractéristiques

Modes de Transfert de Données :

a) **Mode Actif :**

- Dans ce mode, le client ouvre un port et informe le serveur de se connecter à ce port pour commencer la transmission des données.

a) **Mode Passif :**

- Dans le mode passif, c'est le serveur qui ouvre un port et informe le client de s'y connecter pour le transfert de données. Ce mode est souvent utilisé pour éviter les problèmes causés par les pare-feux et les routeurs côté client.

Le mode passif

Dans le **mode passif,** c'est le **serveur** qui prend l'initiative d'**ouvrir un port** pour le transfert de fichiers.

Voici comment ça se passe :

1. Le **client** (toi) se connecte d'abord au **port 21** du serveur pour lui dire : "Je veux transférer un fichier."
2. Le **serveur** répond en disant : "Très bien ! Pour cela, connecte-toi à ce port spécifique (par exemple, 21503) que je viens d'ouvrir."
3. Le **client** se connecte alors à ce **port ouvert par le serveur,** et le transfert de données commence.

🔒 **Pourquoi c'est utile avec les pare-feux et routeurs ?**

Beaucoup de clients (ordinateurs personnels, par exemple) sont protégés par des **pare-feux** ou **routeurs NAT** qui bloquent les connexions **entrantes.**

- En **mode actif,** c'est le **serveur** qui essaie de se connecter au client. Et ça, le pare-feu peut le bloquer.
- En **mode passif,** c'est toujours le **client** qui établit la connexion vers le serveur, ce qui **passe plus facilement à travers les pare-feux.**

Autres caractéristiques

Transmission de Données :

Protocole de Contrôle :

FTP utilise un canal de contrôle pour envoyer des commandes entre le client et le serveur (comme les commandes de connexion, de déconnexion, et de navigation dans les dossiers).

Protocole de Données :

Parallèlement, un canal de données séparé est utilisé pour le transfert de fichiers proprement dit.

1. ***Types de Fichiers et Mode de Transfert :***

Mode ASCII : ou Binaire

- Utilisé pour les fichiers texte, où les données sont converties (si nécessaire) entre les systèmes avec différents formats de fin de ligne.**:**
- Utilisé pour les fichiers non textuels (comme les images, les vidéos, les applications), où les données sont transférées byte par byte sans conversion**.**

Autres services de transfert de fichiers SFTP (SSH File Transfer Protocol)

- **FTPS (FTP Secure) :**
- Extension sécurisée du protocole FTP.
- Utilise SSL/TLS pour crypter les données de transfert.
- Deux modes : explicite sur TLS et implicite sur TLS.
- Sécurise les données transmises, y compris les identifiants d'accès.
- Idéal pour les situations où la sécurité des données est cruciale.

Autres services de transfert de fichiers FTPS (FTP Secure)

- **SFTP (SSH File Transfer Protocol) :**

- Partie de la suite de protocoles SSH.
- Fournit un transfert de fichiers sécurisé ainsi que des opérations sur les systèmes de fichiers.
- Utilise le cryptage SSH pour toutes les communications.
- Utilise un seul canal pour les commandes et les transferts, simplifiant la configuration.
- Permet des méthodes d'authentification avancées, y compris l'authentification par clé publique.

LES BASES DE DONNÉES

Interconnexion entre Bases de Données et Réseaux ?

- Il est facile de croire que les bases de données n'ont pas grand rapport avec les réseaux or :

  - Les bases de données et les réseaux sont interconnectés.
  - Toutes les bases de données sont accessibles via le réseau.
  - Les bases de données utilisent des ports spécifiques pour la connexion.
  - Exemple : Oracle utilise fréquemment le port 5500.

LES PROTOCOLES D'ACCÈS DISTANT

telnet, ssh …

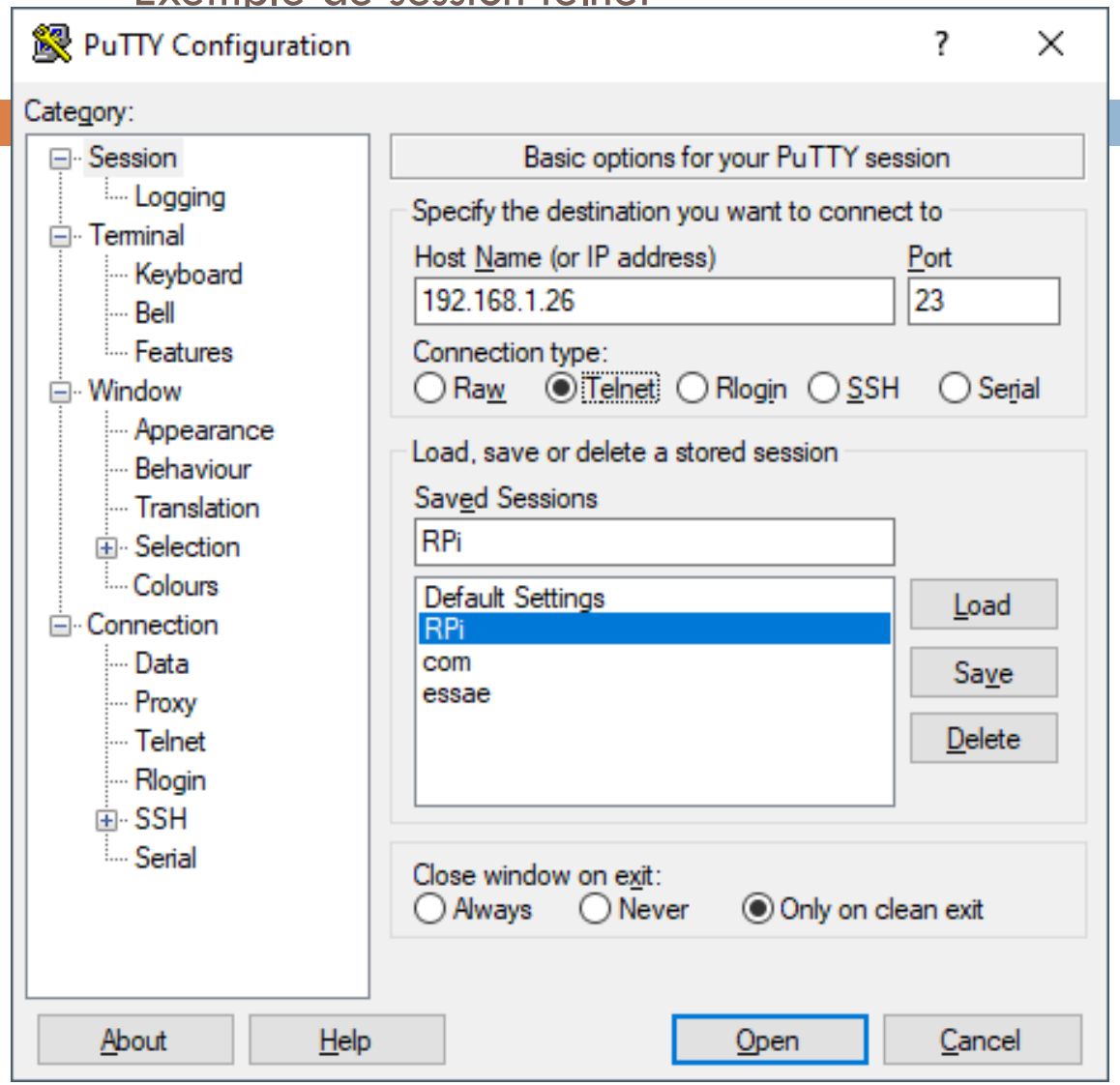
2.4 La couche application : accès distant

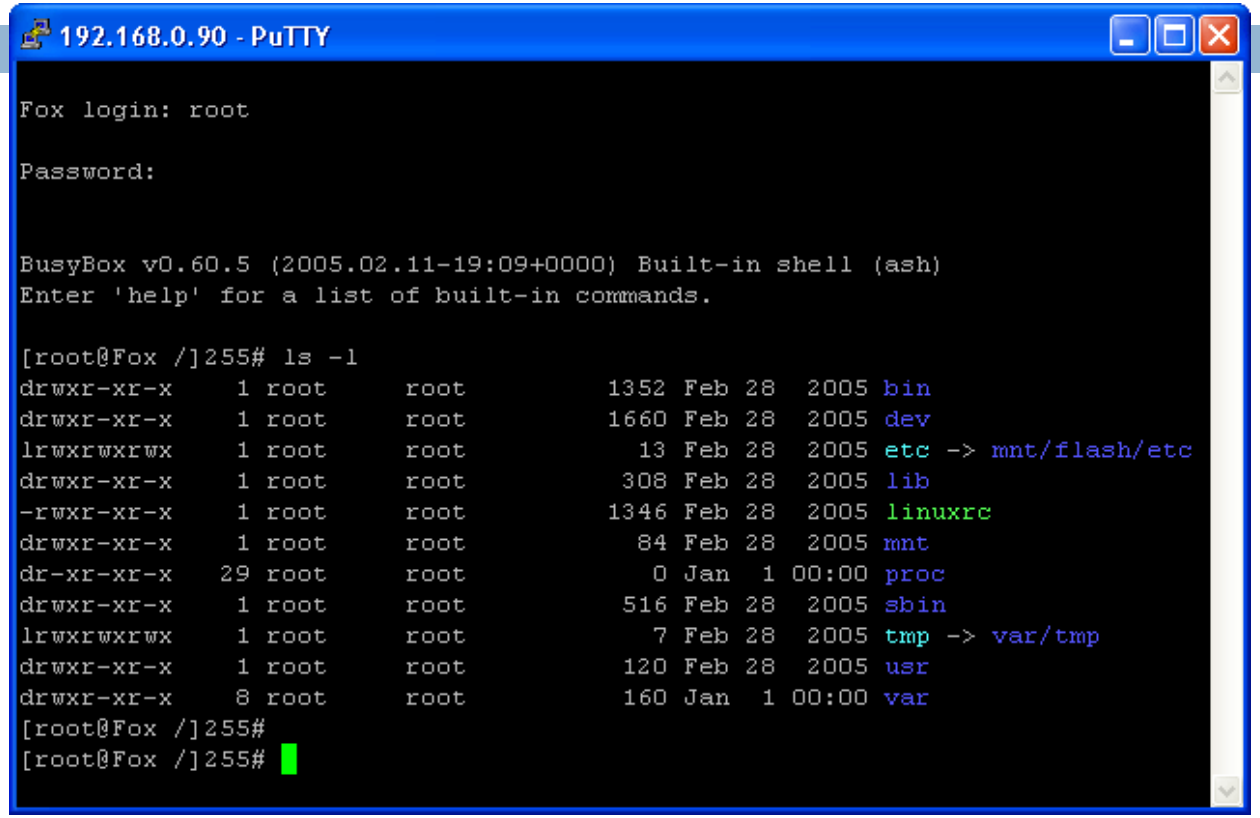
- Telnet est un protocole réseau utilisé pour effectuer une communication bidirectionnelle interactive avec un autre serveur ou appareil en mode texte sur un réseau, principalement pour l'administration à distance de serveurs.

- Etapes

1. Au fur et à mesure que l'utilisateur tape sur son clavier, les caractères sont reçus par le serveur telnet
2. Ce serveur les transmet au système d'exploitation sur lequel il s'exécute comme si les caractères étaient saisies depuis un terminal local.
3. Le résultat des commandes est retournés au client par le serveur
4. Le client affiche les résultats ainsi reçus sur le périphérique d'affichage de la station de travail

Exemple de session telnet





Exemple d'utilisation de telnet

- Ex :

telnet 192.168.109.10

Login user

*Password ******

$ls –la

…

- il faut donc un compte sur la machine
- Du point de vue de l'utilisateur, les réponses semblent venir localement de la machine sur laquelle nous sommes connectée

- TELecommunications NETwork

Fonctionnement C/S de telnet

- La commande *telnet* établie une connexion TCP sur le port 23 du serveur
- TCP est un protocole connecté : ouverture de la connexion, échange de données, puis fermeture
- Une fois la connexion établie : phase de négociation

Sécurité et Authentification SSH

- **Sécurité renforcée** : SSH crypte les communications entre votre ordinateur et un serveur distant, prévenant l'interception et la lecture des données par des tiers.
- **Authentification nécessaire** : Pour établir une connexion SSH, une preuve d'identité est requise, habituellement via un nom d'utilisateur et un mot de passe, ou par l'usage de clés SSH plus sécurisées.

- Les utilitaires Berkeley r*
  - rlogin : telnet
  - rexec : exécution d'une commande sur un PC distant
  - rsh :lancement d'un shell
  - rcp : copie
  - rwho :affiche la liste des utilisateurs

2.4 La couche application : transfert de fichiers

- FTP (File Transfert Protocol)

- TFTP (Trivial File Transfert Protocol)
  - Utilise UDP
  - Donc :
    -
    -
    -
- Simple et compacte -> ROM d'amorçage

NFS (NETWORK FILE SYSTEM) ET SMB (SERVER MESSAGE BLOCK)

deux protocoles de partage de fichiers en réseau, mais ils sont conçus pour des écosystèmes différents et ont des caractéristiques distinctes

NFS (Network File System)

- **Orienté UNIX/Linux :** NFS a été conçu pour les systèmes Unix et est devenu le standard de partage de fichiers pour les systèmes Linux/Unix.
- **Protocole réseau :** Utilise le RPC (Remote Procedure Call) pour la communication.
- **Sécurité :** Les contrôles d'accès sont basés sur les permissions de fichiers Unix et les adresses IP des clients, avec la possibilité d'utiliser Kerberos pour l'authentification et le chiffrement à partir de NFSv4.
- **Performance :** Souvent considéré comme ayant de meilleures performances dans un environnement Unix/Linux pur, en particulier pour les systèmes qui nécessitent un accès aux fichiers à haute performance et à faible latence.
- **Complexité de configuration :** La configuration peut être plus complexe, en particulier en matière de sécurité et de permissions.

SMB (Server Message Block)

- **Interopérabilité avec Windows :** Samba a été conçu pour permettre le partage de fichiers entre Unix/Linux et Windows, en implémentant le protocole SMB/CIFS de Microsoft.
- **Protocole réseau :** Utilise SMB (Server Message Block) et CIFS (Common Internet File System).
- **Sécurité :** Supporte divers niveaux de sécurité, y compris l'authentification basée sur des comptes utilisateurs, des groupes, et l'intégration avec Active Directory.
- **Flexibilité :** Peut servir de contrôleur de domaine et offrir des services d'impression en plus du partage de fichiers.
- **Facilité d'utilisation :** Souvent perçu comme plus facile à configurer et à gérer pour les utilisateurs finaux, en particulier dans un environnement mixte avec des clients Windows.

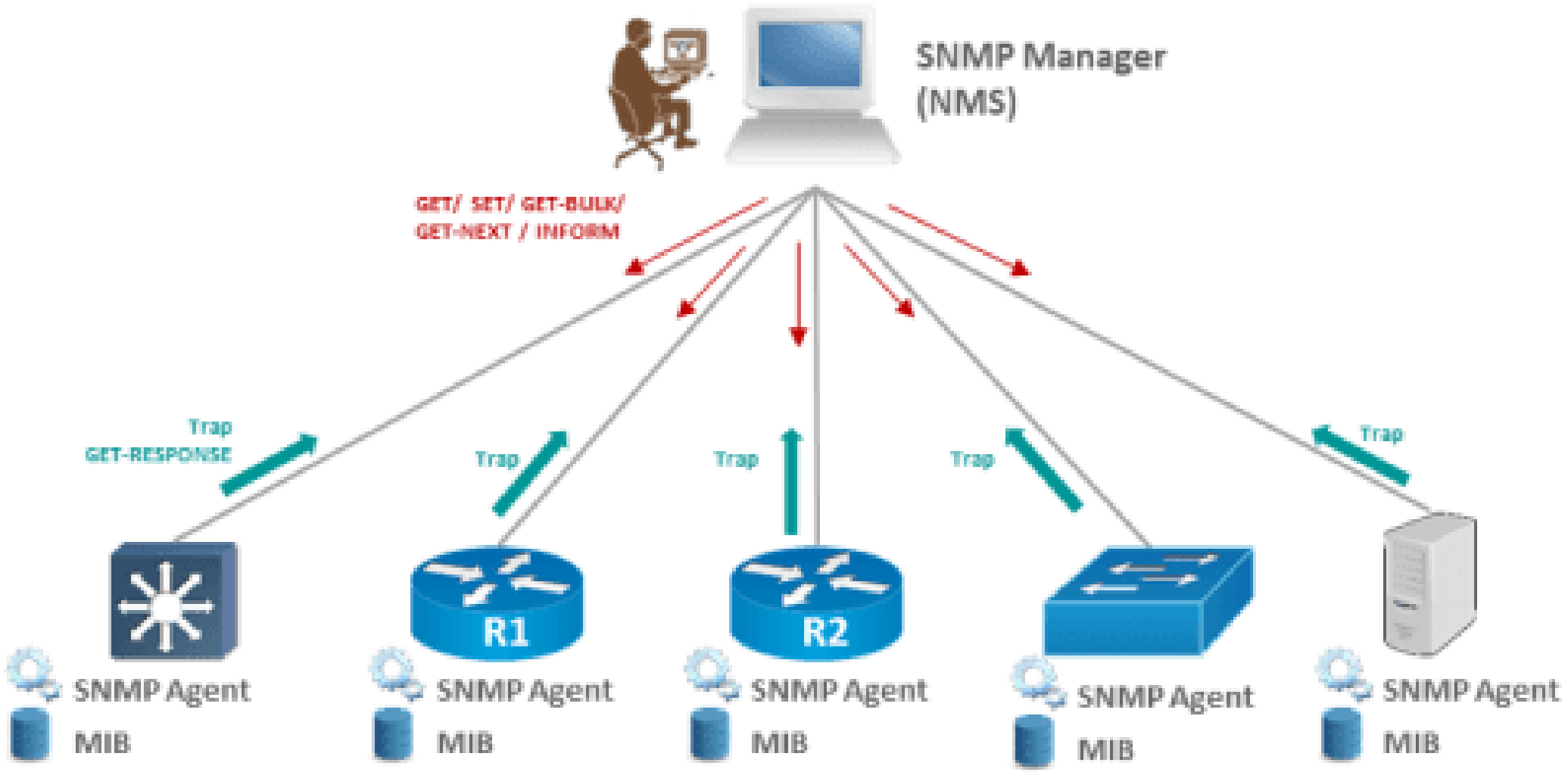
NFS ou SMB ?

- NFS est généralement préféré pour les réseaux Unix/Linux, tandis que Samba est la solution de choix pour permettre la compatibilité entre Linux et Windows :

- **Utilisations complémentaires et Partage simultané :** Il est possible de configurer un même serveur pour offrir des partages NFS pour les clients Linux/Unix et des partages Samba pour les clients Windows en même temps

SNMP (Simple Network Management Protocol)

- SNMP : Simple Network Management Protocol
- D'un côté : les agents SNMP qui s'exécute sur les périphériques
- De l'autre les stations NMS (Network Management Station) qui gèrent les agents.
- La station envoie des requêtes à un agent afin d'obtenir des informations sur son état, son paramétrage …


SNMP Manager
(NMS)
GET/ SET/ GET-BULK/
GET-NEXT / INFORM
Trap
GET-RESPONSE
Trap
Trap
Trap
Trap
R1
R2
SNMP Agent
MIB
SNMP Agent
MIB
SNMP Agent
MIB
SNMP Agent
MIB
SNMP Agent
MIB


SNMP
manager
LoriotPro V4.00, [-bv] - Extended Edition
File View Directory Routers MIB Services Monitor Tools Configure Window Help
Directory
World
LoriotPro
MyOrganisation
ULYSSE
YuMeter [1.3.6.1.2.1
(0002) mgmt
(0001) mib-2
(0001) ifentry
(0001) IfIndex
(0002) ifdescr
(0003) iftype
Agent Name
MyOrganisation
>12.1.1.252
TITAN
ULYSSE
YuMeter [ifinocte...
File Setup
3666
0
11916
Events
TimeStamp
Global Events
Syslogs
Traps
Filters

FIN