

I. Mode client / serveur

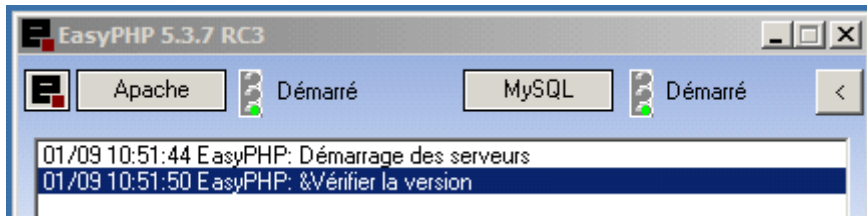
Les pages HTML que nous avons créées précédemment ne sont visibles que depuis notre PC. Pour rendre ces page visibles depuis les autres PC du réseau ou même depuis Internet, il faut utiliser une architecture client/serveur: le serveur HTML héberge la page HTML et attend une sollicitation (requête) venant d'un client HTML situé sur un PC du réseau.

II. EasyPHP:

Le serveur que nous allons utiliser s'appelle "Apache".

- Pour activer Apache cliquer sur l'icône EasyPHP ()

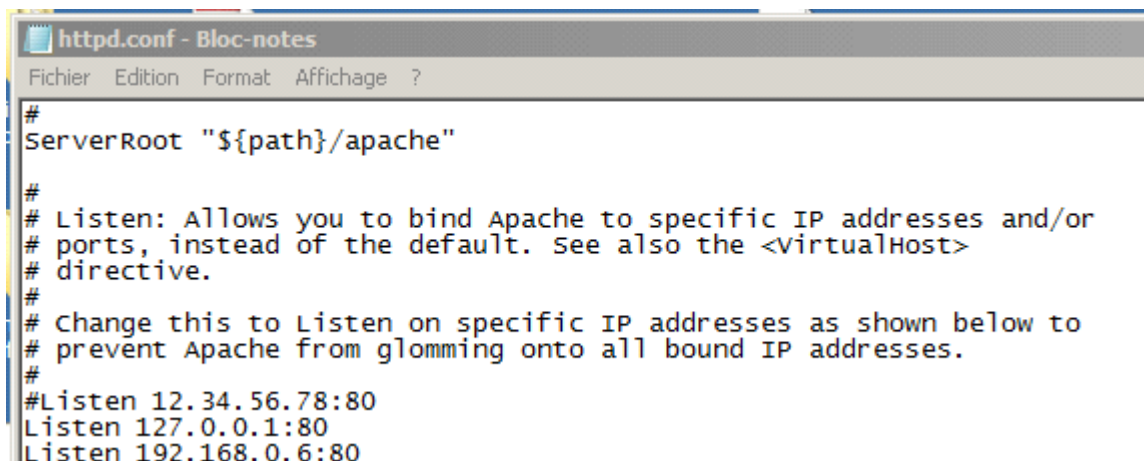
La fenêtre suivante s'ouvre indiquant qu'Apache est démarré:



Il faut maintenant indiquer à Apache qu'il doit "écouter" les requêtes arrivant sur la carte réseau de notre ordinateur et utiliser le port 80 (port par défaut des serveurs HTML) Pour cela:

- Cliquer (bouton droit) sur , puis cliquer sur puis sur

Cela a pour effet d'ouvrir le fichier "httpd.conf" (fichier de configuration de Apache):

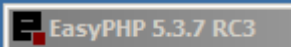


- Ajouter maintenant dans le paragraphe Listen la ligne: Listen adresse IP de la carte réseau de votre PC:80 , enregistrer le fichier et fermer la fenêtre.(utiliser ipconfig pour retrouver l'IP de votre PC)

Mode Local

Il faut maintenant placer le fichier HTML que nous voulons diffuser dans le répertoire www de Apache

Pour accéder à ce répertoire:

- Sélectionner la fenêtre  puis appuyer la touche F8

L'explorateur de Windows s'ouvre alors dans le répertoire ..EasyPHP\www

- Copier alors le fichier exemple1.html dans le répertoire www

Nous allons à présent tester le fonctionnement des serveurs Apaches des PC de la salle

- Ouvrir un client HTML (par exemple Internet Explorer), puis entrer

xxx.xxx.xxx.xxx/exemple1.html (xxx.xxx.xxx.xxx désigne l'IP du serveur que l'on veut tester).

III. La couche TCP

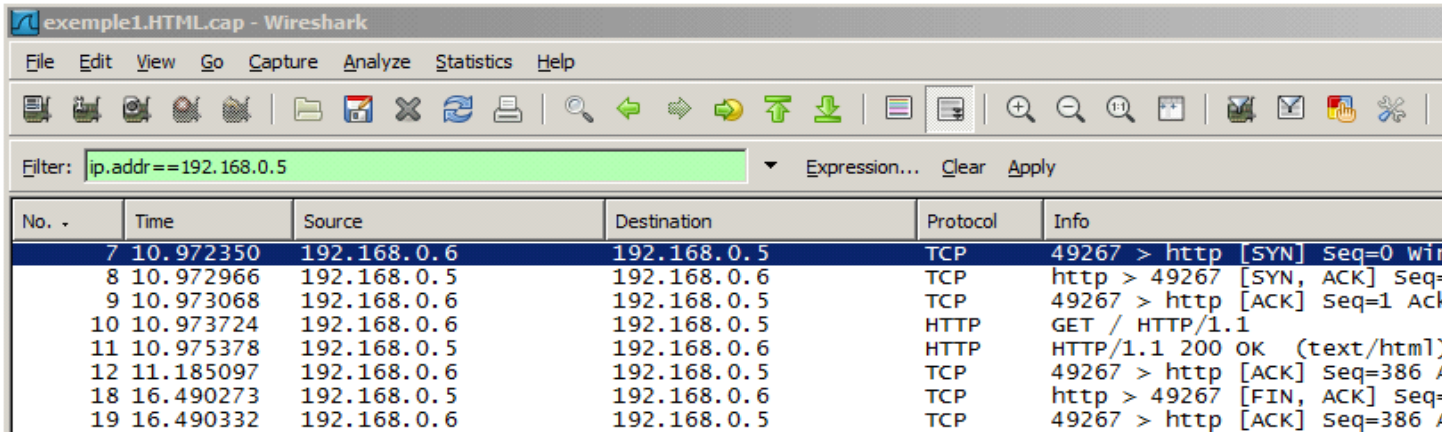
III. 1. Observation des trames:

Nous allons à présent visualiser les informations circulant sur le câble Ethernet et correspondant aux échanges entre client et serveur lors de la visualisation de la page HTML hébergée par le serveur Apache situé dans le PC "vidéo projecteur":

- Ouvrir un client HTML (par exemple Internet Explorer)
- Ouvrir Wireshark et configurer la capture avec le filtre: ip.addr == 10.139.54.199

(rappel: voir l'utilisation de Wireshark dans le document [Activité ping](#))

- Lancer la capture dans la fenêtre wireshark
- Saisir 10.139.54.199/index.html dans la fenêtre Internet Explorer et appuyer la touche Entrée
- Arrêter la capture dans la fenêtre wireshark dès que la page HTML apparaît et sauvegarder dans le dossier Mes documents sous le nom GxHTML_video.pcap (x désigne le numéro de votre binôme)



No.	Time	Source	Destination	Protocol	Info
7	10.972350	192.168.0.6	192.168.0.5	TCP	49267 > http [SYN] Seq=0 win=0
8	10.972966	192.168.0.5	192.168.0.6	TCP	http > 49267 [SYN, ACK] Seq=1
9	10.973068	192.168.0.6	192.168.0.5	TCP	49267 > http [ACK] Seq=1
10	10.973724	192.168.0.6	192.168.0.5	HTTP	GET / HTTP/1.1
11	10.975378	192.168.0.5	192.168.0.6	HTTP	HTTP/1.1 200 OK (text/html)
12	11.185097	192.168.0.6	192.168.0.5	TCP	49267 > http [ACK] Seq=386
18	16.490273	192.168.0.5	192.168.0.6	TCP	http > 49267 [FIN, ACK] Seq=386
19	16.490332	192.168.0.6	192.168.0.5	TCP	49267 > http [ACK] Seq=386

Mode Local

- Relever le contenu complet de la première trame (repérable par l'information [SYN] dans la colonne Info). Remplacer cette copie partielle d'écran par la copie partielle de votre écran
- Repérer sur la trame précédente les différentes couches

A partir de la capture réalisée précédemment (GxHTML_video.pcap), sélectionner la couche TCP (Transmission Control Protocol) de la première trame (repérable par l'information [SYN] dans la colonne Info). Compléter alors le tableau suivant en hexadécimal sauf pour l'octet flag à convertir en binaire:

	contenu	c0	73	00	50	e4	4f	94	15	00	00	00	00	80	00000010
	champ	Source port		Dest. port	Sequence number				Acknowledgement number				H. len	flag	

Le client: je veux communiquer sur le port (0050)₁₆, (port HTML par défaut sur lequel le serveur écoute) il faudra me répondre sur le port (c073)₁₆. mon numéro de séquence est (e44f9415)₁₆

- même travail pour la deuxième trame (repérable par l'information [SYN ,ACK] dans la colonne Info).

	contenu														
	champ	Source port		Dest. port	Sequence number				Acknowledgement number				H. len	flag	

le serveur: bien reçu votre n° de séquence, si j'ajoute 1 cela donne()₁₆ . Mon numéro de séquence est ()₁₆

- troisième trame (repérable par l'information [ACK] dans la colonne Info).

	contenu														
	champ	Source port		Dest. port	Sequence number				Acknowledgement number				H. len	flag	

Le client: bien reçu votre n° de séquence, si j'ajoute 1 cela donne()₁₆ .

- quatrième trame (GET / HTTP/1.1).

	contenu														
	champ	Source port		Dest. port	Sequence number				Acknowledgement number				H. len	flag	

Le client: voici mon message (voir couches suivante de cette trame), son n° d' accusé de réception est ()₁₆ .

- cinquième trame (HTTP/1.1 200 OK).

Mode Local

	contenu														
	champ	Source port	Dest. port	Sequence number				Acknowledgement number				H. len	flag		

Le serveur: j'ai bien reçu votre message n°()¹⁶ voici ma réponse (voir couches suivante de cette trame) son n° d' accusé de réception est ()¹⁶ .

III. 2. Interprétation

a) La couche TCP permet

de communiquer en mode connecté, c'est-à-dire avec des accusés de réception permettant à chaque machine de savoir si l'information qu'elle a envoyée a bien été reçue.

de faire transiter sur la même ligne plusieurs conversations "simultanément". Ceci est possible grâce au concept de ports (ou sockets): un port est un nombre différent associé à chaque machine et valable pendant une "conversation".

b) Établissement d'une connexion

Le processus de communication, qui se fait grâce à une émission de données et d'un accusé de réception, est basé sur un numéro d'ordre (Sequence number ou Acknowledgement number), il faut que les machines émettrices et réceptrices (client et serveur) connaissent le numéro d'ordre initial de l'autre machine.

L'établissement de la connexion entre deux applications se fait souvent selon le schéma suivant :

Les deux machines doivent donc synchroniser leurs séquences grâce à un mécanisme communément appelé three ways handshake (poignée de main en trois temps).

Ce dialogue permet d'initier la communication, il se déroule en trois temps, comme sa dénomination l'indique :

* Dans un premier temps la machine émettrice (le client) transmet une trame dont le drapeau SYN est à 1 (pour signaler qu'il s'agit d' une trame de synchronisation), avec un numéro de séquence C

* Dans un second temps la machine réceptrice (le serveur) reçoit la trame initiale provenant du client, puis lui envoie un accusé de réception, c'est-à-dire une trame dont le drapeau ACK est à 1 et le drapeau SYN est à 1 (car il s'agit là encore d'une synchronisation). Ce segment contient un numéro de séquence S et un numéro d'accusé de réception C+1,

* Enfin, le client transmet au serveur un accusé de réception, c'est-à-dire un segment dont le drapeau ACK est à 1, dont le drapeau SYN est à zéro (il ne s'agit plus d' une trame de synchronisation). Son numéro de séquence est C+1 et son numéro d'accusé de réception est S+1.

Suite à cette séquence comportant trois échanges les deux machines sont synchronisées et la communication peut commencer!

Lors de l'émission d'une trame, un numéro d'accusé de réception et un numéro de séquence sont associés. A réception de cette trame, la machine réceptrice va retourner une trame dont le drapeau ACK est à 1 (afin de signaler qu'il s'agit d'un accusé de réception) accompagné d'un numéro de séquence égal au numéro d'accusé de réception précédent et d'un nouveau numéro d'accusé de réception,

De plus, grâce à une minuterie déclenchée dès l'émission d'une trame au niveau de la machine émettrice, la trame est réexpédiée dès que le temps imparti est écoulé, car dans ce cas la machine émettrice considère que la trame est perdue...

Toutefois, si la trame n'est pas perdue et qu'elle arrive tout de même à destination, la machine réceptrice saura grâce au numéro d'ordre qu'il s'agit d'un doublon et ne conservera que la dernière trame arrivée à destination...

IV. La couche HTTP

IV. 1. Observation des trames

- Combien de couche comporte la trame "GET /index.html HTTP/1.1"
- Quelle est la machine qui a émis cette trame et quel est le statut (client ou serveur) de cette machine?
- A partir de la capture réalisée précédemment (GxHTML_video.pcap), sélectionner la couche HTTP (Hypertext Transfer Protocol) de la trame "GET /index.html HTTP/1.1"

Remarque: la couche http est un texte codé en ASCII, les "\r\n" correspondent aux octets 0d (Carriage Return) et 0a (Line Feed).

- Combien de lignes cette couche http comporte-t-elle?

La couche étudiée est une demande émise par une machine ayant un statut de client. On l'appelle une requête http.

a) structure d'une requête http:

- ligne de requête:

	Method	space	Request URI*	space	HTTP-version	CR LF
<i>exemple:</i>	GET		/exemple1.html		HTTP1.1	

*Uniform Resource Identifier

- lignes suivantes: (facultatives): ces lignes fournissent des informations sur le client HTML :

exemples

User-Agent	indique le client HTML utilisé
Accept-Language	indique que langue doit être utilisée pour la réponse
Accept	indique le type de donnée à utiliser pour la réponse

Ligne vide:

IV. 2. structure d'une réponse http:

- ligne de réponse:

Compléter le tableau dans le cas d'une première requête (exemple1) puis dans le cas d'une deuxième requête, la page n'ayant pas été modifiée entre temps et dans

	HTTP-version	space	Status-Code	space	Reason-Phrase	CR LF
<i>exemple 1</i>						
<i>exemple 2</i>						

- lignes suivantes: (facultatives): ces lignes fournissent des informations fournies par le serveur HTML :

exemples

Server	indique le serveur HTML utilisé
Date	indique que date de la réponse
Keep-Alive	indique la durée pendant laquelle le serveur restif actif en cas de nom réponse du client

- Ligne vide:

V. La couche HTML

Elle contient en mode texte la page HTML. Dans wireshark, les caractères non imprimables sont noté:

\t pour la tabulation , \n pour les saut de ligne.