TP - Étude du protocole HTTP

M. Tellene

I Objectif du TP

Dans un premier temps, l'objectif de ce TP est d'analyser les échanges HTTP fait lors de demandes de pages web de différents types (classique, volumineuse, avec ressources et protégée).

II Rendu

Le TP comporte des questions. Les réponses devront être rendues dans un fichier pdf sur le moodle à la fin de la séance. Vous devrez joindre, aux moments indiqués, des captures d'écrans de Wireshark.

III Configuration de la machine virtuelle

Pour le TP, vous aurez besoin d'une machine virtuelle Windows avec *Wireshark*. Les étapes d'installation de *Wireshark* sont indiquées ci-dessous :

1. Aller à l'adresse suivante : https://www.wireshark.org/download.html
2. Télécharger la version adéquate de *Wireshark*
3. Lancer l'installer *Wireshark*
4. Faire *Next* > *Noted* > *Next* > *Next* > *Next* > *Next* > ***Si ce n'est pas coché, cocher « Install NpCap 1.78 »*** > *Next* > *Install*. Lire le point suivant pendant l'installation !
5. Pendant l'installation, une fenêtre va s'ouvrir, faire *I Agree* > ***Cocher Install NpCap in WinPcap API-compatible Mode »*** > *Install*. Une fois l'installation finie, faire *Next* > *Finish*. La fenêtre précédente se rouvre, une fois l'installation finie, faire *Next* > *Finish*
6. Changer les paramètres réseau de votre machine virtuelle si cette dernière n'est pas connectée à Internet

IV L'interaction HTTP GET/réponse de base

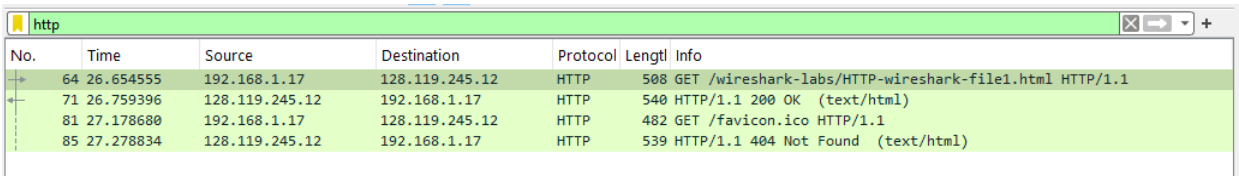
1. Commencez par démarrer votre navigateur (évitez *Edge* si vous le pouvez)

2. Démarrez *Wireshark* et sélectionnez votre carte réseau

3. Tapez l'URL suivante dans la barre de recherche :
   http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file1.html. Une fois la page chargée, arrêtez l'enregistrement sur *Wireshark*

4. **TODO** Faire une capture d'écran de la page web obtenue

5. Dans *Wireshark*, appliquez le filtre **http**. Vous devriez avoir quelque chose comme ceci :



La capture précédente montre dans la fenêtre de liste des paquets que quatre messages HTTP ont été capturés : le message GET (de votre navigateur au serveur `/wireshark-labs/HTTP-wireshark-file1.html`) et le message de réponse du serveur à votre navigateur, ainsi qu'une demande de *favicon.ico*.

La fenêtre de contenu des paquets affiche les détails du message sélectionné (panneau en bas à gauche). Rappelons que le message HTTP a été transporté :

— à l'intérieur d'un segment TCP

— qui lui-même est à l'intérieur d'un datagramme IP

— qui lui-même est à l'intérieur d'une trame Ethernet

IV/1 Questions

Pour trouver les réponses, il va falloir étendre la partie *Hypertext Transfer Protocol* (dans le panneau en bas à gauche) des différents messages HTTP envoyés et reçus.

1. Quelle est la version d'HTTP de votre navigateur? Celle du serveur?

2. Quelles langues votre navigateur indique-t-il qu'il peut accepter en réponse? Vous indiquerez après quel champ se trouve cette information.

3. Quelle est l'adresse IP de votre ordinateur? Du serveur `gaia.cs.umass.edu`?

4. Quel est le code statut renvoyé par le serveur à votre navigateur? À quoi cela correspond?

5. Quand le fichier HTML que vous récupérez a-t-il été modifié pour la dernière fois sur le serveur? Vous indiquerez après quel champ se trouve cette information.

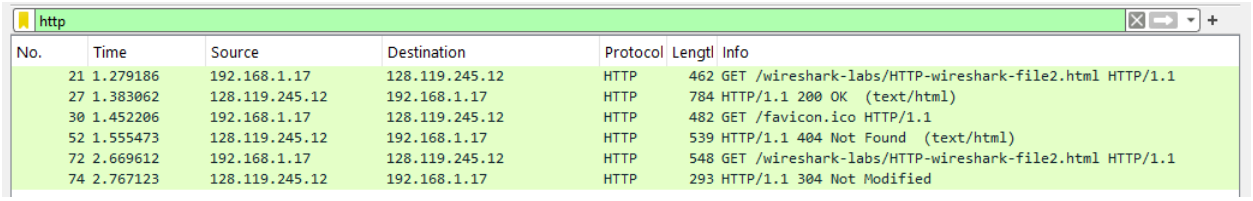
6. Combien d'octets fait le fichier renvoyé par le serveur? Vous indiquerez où se trouve cette information.

V GET conditionnel HTTP

1. Redémarrez un enregistrement (icône aileron, à côté du bouton stop). N'enregistrez pas la capture précédente

2. Toujours dans votre navigateur, saisissez l'URL suivante : http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file2.html, et rapidement après, entrer à nouveaux la même URL (ou cliquer sur le bouton de rafraîchissement)

3. Arrêtez la capture de paquet de *Wireshark*. Vous devriez avoir ceci :



4. **TODO** Faire une capture d'écran de la page Web obtenue

V/1 Questions

1. Dans la première requête `HTTP GET`, le champ *If-Modified-Since* est-il présent ?

2. Le serveur a-t-il retourné le contenu du fichier ? Comment peut-on voir cette information ?

3. Dans la seconde requête `HTTP GET`, le champ *If-Modified-Since* est-il présent ? Si oui, quelle information contient-il ?

4. Quels sont les code statut et message retournés par le serveur ? Qu'est-ce que cela veut dire ? Le serveur a-t-il retourné le contenu du fichier ?

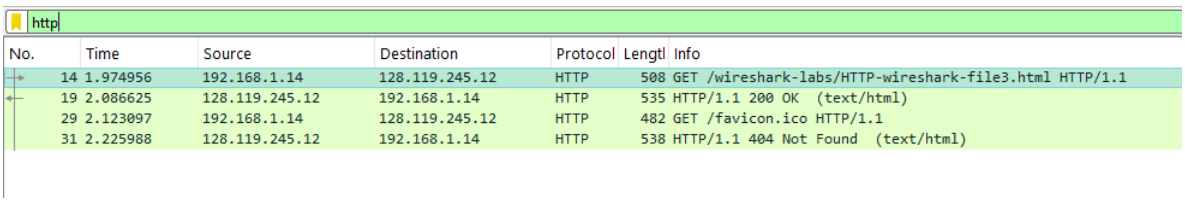
VI Récupération de longs documents

Dans l'exemple précédent, le document récupéré était un fichier `HTML` simple et court. Voyons maintenant ce qui se passe lorsque nous téléchargeons un long fichier `HTML`. Procédez comme suit :

— Redémarrez un enregistrement (icône d'un aileron, à côté du bouton stop). N'enregistrez pas la capture précédente

— Toujours dans votre navigateur, saisissez l'`URL` suivante http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file3.html. Votre navigateur devrait afficher une page ayant pour titre *THE BILL OF RIGHTS*

— Arrêtez la capture de paquets de *Wireshark*. Vous devriez avoir ceci :

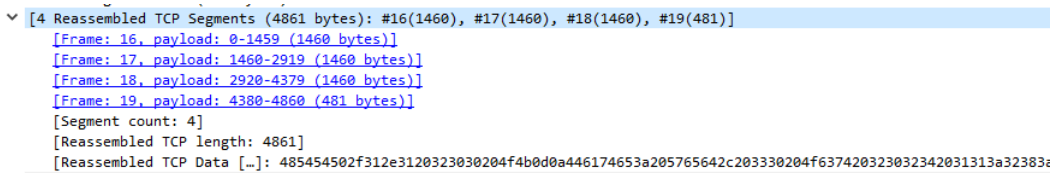


— **TODO** Faire une capture d'écran de la page Web obtenue

Dans la fenêtre de listage des paquets, vous devriez voir votre message `HTTP GET` (première ligne), suivi d'une réponse en plusieurs paquets à votre requête `HTTP GET` (seconde ligne). Cette réponse à paquets multiples mérite quelques explications.

Dans notre cas, le fichier HTML est assez long et donc trop volumineux pour tenir dans un seul paquet TCP. Le message de réponse HTTP unique est donc divisé en plusieurs morceaux par TCP, chaque morceau étant contenu dans un segment TCP distinct.

Wireshark indique chaque segment TCP comme un paquet séparé, et le fait que la réponse HTTP unique a été fragmentée en plusieurs paquets TCP est indiqué par le message « ... *Reassembled TCP Segment* ». Ce message est indiqué dans la panneau inférieur, entre le détail TCP et HTTP.



VI/1 Questions

1. Combien de messages de requête HTTP GET votre navigateur a-t-il envoyés ? (Sans compter le favicon si ce dernier apparaît dans la trace) Quel numéro de paquet dans la trace contient le message GET pour *THE BILL OF RIGHTS* ? Pour déterminer le numéro de paquet, regardez la colonne No. de la trace.

2. Quel est le numéro de paquet, dans la trace, qui contient le code statut et la phrase associés à la réponse à la requête HTTP GET ? Pour déterminer le numéro de paquet, regardez la colonne No. de la trace.

3. Combien de segments TCP contenant des données ont été nécessaires pour transporter la réponse HTTP ?

4. Quelles sont les tailles de chacun de ces paquets ? Quel est donc la taille totale des données TCP envoyées par le serveur ?

VII Documents HTML avec objets intégrés

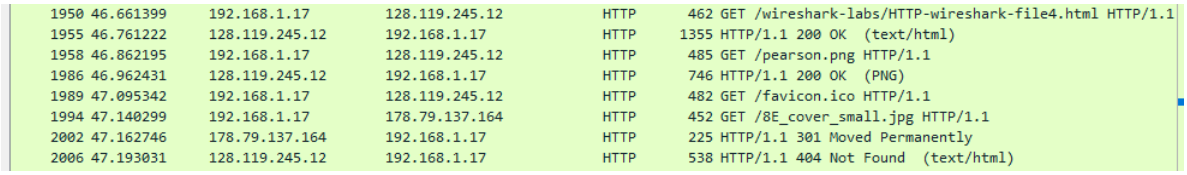
Maintenant que nous avons vu comment *Wireshark* affiche le trafic de paquets capturés pour les gros fichiers HTML, nous pouvons examiner ce qui se passe lorsque votre navigateur télécharge un fichier avec des objets intégrés, c'est-à-dire un fichier qui inclut d'autres objets (dans l'exemple ci-dessous, des fichiers images) qui sont stockés sur un ou plusieurs autres serveurs. Procédez comme suit :

— Redémarrez un enregistrement (icône d'un aileron, à côté du bouton stop). N'enregistrez pas la capture précédente

— Toujours dans votre navigateur, saisissez l'URL suivante
http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file4.htm

Votre navigateur doit afficher un court fichier HTML contenant deux images. Ces deux images sont référencées dans le fichier HTML de base. En d'autres termes, les images elles-mêmes ne sont pas contenues dans le fichier HTML, mais les URL des images sont contenues dans le fichier HTML téléchargé

— Arrêtez la capture de paquets *Wireshark*. Vous devriez avoir ceci :



1950	46.661399	192.168.1.17	128.119.245.12	HTTP	462 GET /wireshark-labs/HTTP-wireshark-file4.html HTTP/1.1
1955	46.761222	128.119.245.12	192.168.1.17	HTTP	1355 HTTP/1.1 200 OK (text/html)
1958	46.862195	192.168.1.17	128.119.245.12	HTTP	485 GET /pearson.png HTTP/1.1
1986	46.962431	128.119.245.12	192.168.1.17	HTTP	746 HTTP/1.1 200 OK (PNG)
1989	47.095342	192.168.1.17	128.119.245.12	HTTP	482 GET /favicon.ico HTTP/1.1
1994	47.140299	192.168.1.17	178.79.137.164	HTTP	452 GET /8E_cover_small.jpg HTTP/1.1
2002	47.162746	178.79.137.164	192.168.1.17	HTTP	225 HTTP/1.1 301 Moved Permanently
2006	47.193031	128.119.245.12	192.168.1.17	HTTP	538 HTTP/1.1 404 Not Found (text/html)

— **TODO** Faire une capture d'écran de la page Web obtenue

VII/1 Questions

1. Combien de requêtes `HTTP GET` votre navigateur a-t-il envoyées? À quelles adresses `IP` ces requêtes `GET` ont-elles été envoyées? Que peut-on en conclure sur l'emplacement des ressources?

2. Combien de segments `TCP` ont été utilisés pour chacune des réponses?

VIII Authentification `HTTP`

Enfin, essayons de visiter un site web protégé par un mot de passe et examinons la séquence de messages `HTTP` échangés pour un tel site.

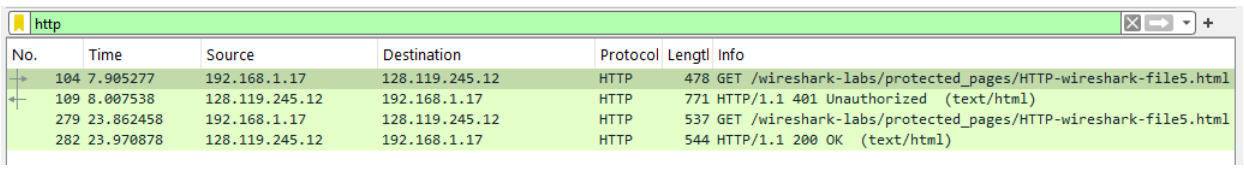
L'`URL`
http://gaia.cs.umass.edu/wireshark-labs/protected_pages/HTTP-wireshark-file5.html est protégée par un mot de passe. Le nom d'utilisateur est "wireshark-students" (sans les guillemets) et le mot de passe est "network" (toujours sans les guillemets). Accédons donc à ce site sécurisé protégé par un mot de passe. Procédez comme suit :

— Redémarrez un enregistrement (icône d'un aileron, à côté du bouton stop). N'enregistrez pas la capture précédente

— Toujours dans votre navigateur, saisissez l'`URL` suivante
http://gaia.cs.umass.edu/wireshark-labs/protected_pages/HTTP-wireshark-file5.html

— Tapez le nom d'utilisateur et le mot de passe demandés dans la fenêtre contextuelle

— Arrêtez la capture de paquets *Wireshark*. Vous devriez avoir ceci :



http

No.	Time	Source	Destination	Protocol	Lengtl	Info
104	7.905277	192.168.1.17	128.119.245.12	HTTP	478	GET /wireshark-labs/protected_pages/HTTP-wireshark-file5.html
109	8.007538	128.119.245.12	192.168.1.17	HTTP	771	HTTP/1.1 401 Unauthorized (text/html)
279	23.862458	192.168.1.17	128.119.245.12	HTTP	537	GET /wireshark-labs/protected_pages/HTTP-wireshark-file5.html
282	23.970878	128.119.245.12	192.168.1.17	HTTP	544	HTTP/1.1 200 OK (text/html)

— **TODO** Faire une capture d'écran de la page Web obtenue

VIII/1 Questions

1. Quelle est la réponse du serveur (code statut et phrase) au premier message `HTTP GET` envoyé par votre navigateur? Qu'est-ce que cela veut dire?

2. Lorsque votre navigateur envoie le message `HTTP GET` pour la deuxième fois, quel nouveau champ est inclus dans le message `HTTP GET` ? Que contient-il de manière générale (la réponse attendue n'est pas la valeur du champ) ?

Le nom d'utilisateur (*wireshark-students*) et le mot de passe (*network*) que vous avez saisis sont encodés dans la chaîne de caractères (*d2lyZXNoYXJrLXN0dWRlbnRzOm5ldHdvcms=*) qui suit l'en-tête « *Authorization : Basic* » dans le message `HTTP GET` du client. Bien qu'il puisse sembler que votre nom d'utilisateur et votre mot de passe soient chiffrés, ils sont simplement encodés dans un format connu sous le nom de format *Base64*.

Avec cette méthode de chiffrement, il est possible de retrouver la valeur en clair ! Pour le constater, rendez-vous sur le site https://www.base64decode.org/ et entrez la chaîne chiffrée *d2lyZXNoYXJrLXN0dWRlbnRzOm5ldHdvcms=* et déchiffrez-la. Vous devriez voir votre nom d'utilisateur et votre mot de passe.

Étant donné que n'importe qui peut télécharger un outil comme *Wireshark* et « renifler » les paquets (pas seulement les siens) qui passent par son adaptateur réseau, et que n'importe qui peut traduire le code *Base64* en `ASCII` (vous venez de le faire), il devrait être clair pour vous que les mots de passe simples sur les sites `HTTP` ne sont pas sûrs.