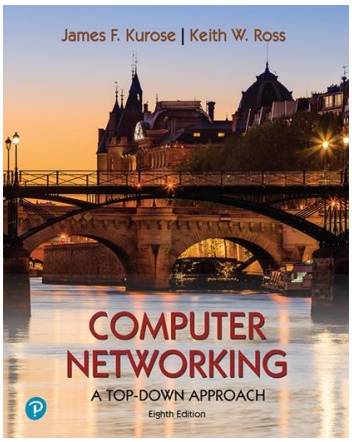
Laboratoire Wireshark : HTTP v8.0

Supplement to *Computer Networking: A Top-Down Approach, 8e éd.* , J.F. Kurose et K.W. Ross



« Dites-le moi et j'oublie. Montrez-moi et je me souviens. Impliquez-moi et je comprends. Proverbe chinois

---

Après nous être mouillés les pieds avec le renifleur de paquets Wireshark dans le laboratoire d'introduction, nous sommes maintenant prêts à utiliser Wireshark pour étudier les protocoles en fonctionnement. Dans cet atelier, nous explorerons plusieurs aspects du protocole HTTP : l'interaction GET/response de base, les formats de message HTTP, la récupération de fichiers HTML volumineux, la récupération de fichiers HTML avec des objets incorporés, ainsi que l'authentification et la sécurité HTTP. Avant de commencer ces ateliers, vous voudrez peut-être passer en revue la section 2.2 du texte.[1]

1. L'interaction HTTP GET/response de base

Commençons notre exploration de HTTP en téléchargeant un fichier HTML très simple - un fichier très court et ne contenant aucun objet incorporé. Procédez comme suit :

1. Démarrez votre navigateur Web.
2. Démarrez le renifleur de paquets Wireshark, comme décrit dans le laboratoire d'introduction (mais ne commencez pas encore la capture de paquets). Entrez « http » (seulement les lettres, pas les guillemets) dans la fenêtre display-filter-specification, de sorte que seuls les messages HTTP capturés seront affichés plus tard dans la fenêtre de liste de paquets. (Nous ne nous intéressons qu'au protocole HTTP ici, et nous ne voulons pas voir l'encombrement de tous les paquets capturés).
3. Attendez un peu plus d'une minute (nous verrons pourquoi sous peu), puis commencez la capture de paquets Wireshark.
4. Entrez ce qui suit dans votre navigateur http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file1.htmlVotre navigateur doit afficher le fichier HTML très simple d'une ligne.
5. Arrêtez la capture de paquets Wireshark.

---

[1] Les références aux figures et aux sections sont pour le 8ième édition de notre texte, *Réseaux informatiques, une approche descendante, 8h ed., J.F. Kurose et K.W. Ross, Addison-Wesley/Pearson, 2020.*

Votre fenêtre Wireshark doit ressembler à la fenêtre illustrée à la figure 1. Si vous ne parvenez pas à exécuter Wireshark sur une connexion réseau en direct, vous pouvez télécharger une trace de paquets créée lorsque les étapes ci-dessus ont été suivies.[2]

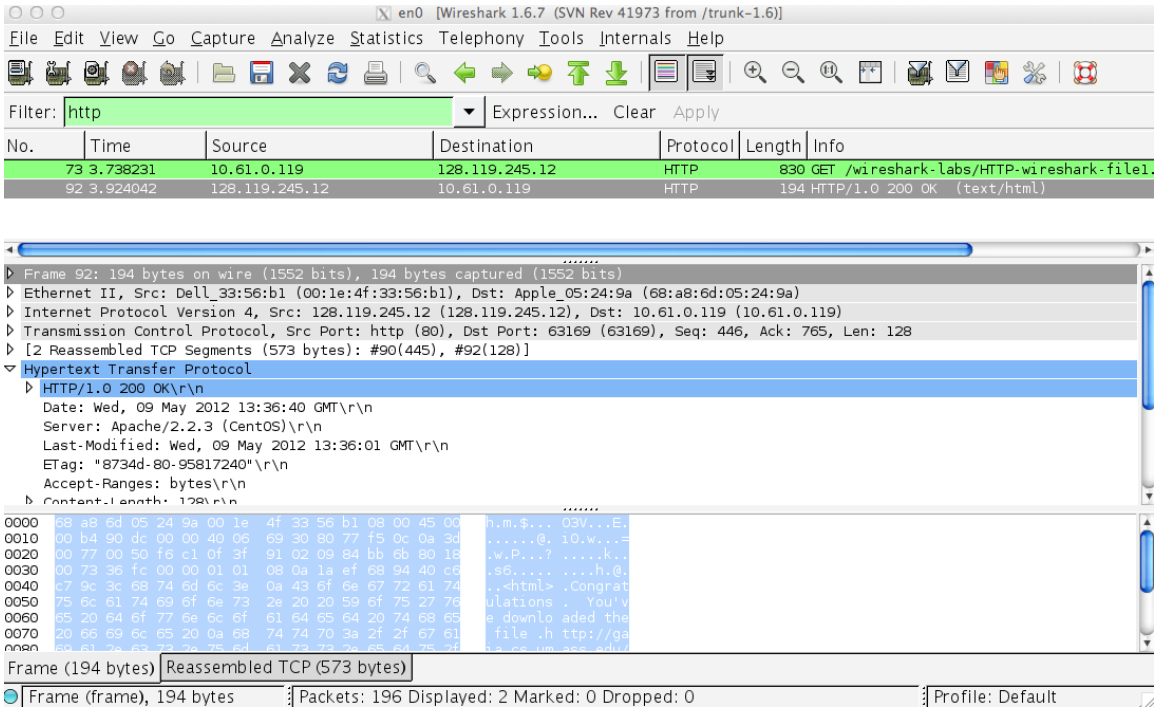


Figure 1 : Affichage Wireshark après http://gaia.cs.umass.edu/wireshark-labs/ fichier HTTP-wireshark1.html a été récupéré par votre navigateur

L'exemple de la figure 1 montre dans la fenêtre de liste de paquets que deux messages HTTP ont été capturés : le message GET (de votre navigateur au serveur Web gaia.cs.umass.edu) et le message de réponse du serveur à votre navigateur. La fenêtre du contenu des paquets affiche les détails du message sélectionné (dans ce cas, le message HTTP OK, qui est mis en surbrillance dans la fenêtre de liste des paquets). Rappelez-vous que puisque le message HTTP était transporté à l'intérieur d'un segment TCP, qui était transporté à l'intérieur d'un datagramme IP, qui était transporté dans une trame Ethernet, Wireshark affiche également les informations sur les paquets Frame, Ethernet, IP et TCP. Nous voulons minimiser la quantité de données non-HTTP affichées (nous nous intéressons à HTTP ici, et nous étudierons ces autres protocoles ultérieurement dans les laboratoires), alors assurez-vous que les boîtes à l'extrême gauche des informations

[2] Télécharger le fichier zip http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip et extrayez le fichier **http-ethereal-trace-1**. Les traces dans ce fichier zip ont été collectées par Wireshark en cours d'exécution sur l'un des ordinateurs de l'auteur, tout en effectuant les étapes indiquées dans le laboratoire Wireshark. Une fois que vous avez téléchargé la trace, vous pouvez la charger dans Wireshark et afficher la trace à l'aide de l'option *Lime* menu déroulant, sélection *Ouvrir*, puis en sélectionnant l'application **Fichier de trace http-ethereal-trace-1**. L'affichage résultant doit ressembler à la figure 1. (L'interface utilisateur de Wireshark s'affiche un peu différemment sur différents systèmes d'exploitation et dans différentes versions de Wireshark).

Frame, Ethernet, IP et TCP ont un signe plus ou un triangle pointant vers la droite (ce qui signifie qu'il y a des informations cachées et non affichées), et la ligne HTTP a un signe moins ou un triangle pointant vers le bas (ce qui signifie que toutes les informations sur le message HTTP sont affichées).

> (*Remarque:* Vous devez ignorer tout HTTP GET et réponse pour favicon.ico. Si vous voyez une référence à ce fichier, c'est votre navigateur qui demande automatiquement au serveur s'il (le serveur) a un petit fichier d'icône qui doit être affiché à côté de l'URL affichée dans votre navigateur. Nous ignorerons les références à ce fichier embêtant dans cet atelier.).

En examinant les informations contenues dans les messages HTTP GET et de réponse, répondez aux questions suivantes. Lorsque vous répondez aux questions suivantes, vous devez imprimer les messages GET et de réponse (voir le laboratoire Wireshark d'introduction pour une explication de la procédure à suivre) et indiquer où vous avez trouvé dans le message les informations qui répondent aux questions suivantes. Lorsque vous revoyez votre devoir, annotez la sortie afin qu'il soit clair où dans la sortie vous obtenez les informations pour votre réponse (par exemple, pour nos cours, nous demandons aux étudiants de baliser les copies papier avec un stylo ou d'annoter des copies électroniques avec du texte dans une police colorée).

1. Votre navigateur exécute-t-il HTTP version 1.0 ou 1.1 ? Quelle version de HTTP le serveur exécute-t-il ?
2. Quelles langues (le cas échéant) votre navigateur indique-t-il qu'il peut accepter pour le serveur?
3. Quelle est l'adresse IP de votre ordinateur ? Du serveur gaia.cs.umass.edu ?
4. Quel est le code d'état renvoyé par le serveur à votre navigateur ?
5. Quand le fichier HTML que vous récupérez a-t-il été modifié pour la dernière fois sur le serveur ?
6. Combien d'octets de contenu sont renvoyés à votre navigateur ?

Dans votre réponse à la question 5 ci-dessus, vous avez peut-être été surpris de constater que le document que vous venez de récupérer a été modifié pour la dernière fois dans la minute qui a précédé le téléchargement du document. En effet, (pour ce fichier particulier), le serveur gaia.cs.umass.edu définit l'heure de la dernière modification du fichier sur l'heure actuelle, et le fait une fois par minute. Ainsi, si vous attendez une minute entre les accès, le fichier semblera avoir été récemment modifié, et donc votre navigateur téléchargera une « nouvelle » copie du document.

2. L'interaction HTTP CONDITIONAL GET/response

Rappelez-vous de la section 2.2.5 du texte, que la plupart des navigateurs Web effectuent la mise en cache d'objets et effectuent donc un GET conditionnel lors de la récupération d'un objet HTTP. Avant d'effectuer les étapes ci-dessous, assurez-vous que le cache de votre navigateur est vide. (Pour ce faire sous Firefox, sélectionnez *Outils-> Historique récent* et cochez la case Cache,ou pour Internet Explorer, sélectionnez *Outils->Optoires Internet->Supprimer le fichier;* ces actions supprimeront les fichiers mis en cache du cache de votre navigateur.) Maintenant, procédez comme suit:

- Démarrez votre navigateur Web et assurez-vous que le cache de votre navigateur est effacé, comme indiqué ci-dessus.
- Démarrez le renifleur de paquets Wireshark
- Entrez l'URL suivante dans votre navigateur http://gaia.cs. umass.edu/wireshark-labs/HTTP-wireshark-file2.htmlVotre navigateur doit afficher un fichier HTML très simple de cinq lignes.
- Entrez rapidement la même URL dans votre navigateur (ou sélectionnez simplement le bouton d'actualisation de votre navigateur)
- Arrêtez la capture de paquets Wireshark et entrez « http » dans la fenêtre display-filter-specification, de sorte que seuls les messages HTTP capturés seront affichés ultérieurement dans la fenêtre de liste de paquets.
- (*Remarque:* Si vous ne parvenez pas à exécuter Wireshark sur une connexion réseau en direct, vous pouvez utiliser la trace de **paquets http-ethereal-trace-2** pour répondre aux questions ci-dessous; voir note de bas de page 1. Ce fichier de trace a été collecté lors de l'exécution des étapes ci-dessus sur l'un des ordinateurs de l'auteur.)

Répondez aux questions suivantes :

7. Inspectez le contenu de la première requête HTTP GET de votre navigateur vers le serveur. Voyez-vous une ligne « IF-MODIFIED-SINCE » dans le HTTP GET ?
8. Inspectez le contenu de la réponse du serveur. Le serveur a-t-il renvoyé explicitement le contenu du fichier ? Comment pouvez-vous le dire?
9. Maintenant, inspectez le contenu de la deuxième requête HTTP GET de votre navigateur au serveur. Voyez-vous une ligne « IF-MODIFIED-SINCE: » dans le HTTP GET ? Si oui, quelles informations suivent l'en-tête « IF-MODIFIED-SINCE: »?
10. Quel est le code d'état HTTP et la phrase renvoyée par le serveur en réponse à ce deuxième HTTP GET ? Le serveur a-t-il renvoyé explicitement le contenu du fichier ? Expliquer.

3. Récupération de documents longs

Dans nos exemples jusqu'à présent, les documents récupérés ont été des fichiers HTML simples et courts. Voyons maintenant ce qui se passe lorsque nous téléchargeons un long fichier HTML. Procédez comme suit :

- Démarrez votre navigateur Web et assurez-vous que le cache de votre navigateur est effacé, comme indiqué ci-dessus.
- Démarrez le renifleur de paquets Wireshark
- Entrez l'URL suivante dans votre navigateur http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file3.html
  Votre navigateur devrait afficher la déclaration des droits des États-Unis, plutôt longue.

- Arrêtez la capture de paquets Wireshark et entrez « http » dans la fenêtre display-filter-specification, afin que seuls les messages HTTP capturés soient affichés.
- (*Remarque:* Si vous ne parvenez pas à exécuter Wireshark sur une connexion réseau en direct, vous pouvez utiliser la trace de **paquets http-ethereal-trace-3** pour répondre aux questions ci-dessous; voir note de bas de page 1. Ce fichier de trace a été collecté lors de l'exécution des étapes ci-dessus sur l'un des ordinateurs de l'auteur.)

Dans la fenêtre de liste de paquets, vous devriez voir votre message HTTP GET, suivi d'une réponse TCP à plusieurs paquets à votre requête HTTP GET. Cette réponse à paquets multiples mérite un peu d'explication. Rappelez-vous de la Section 2.2 (voir Figure 2.9 dans le texte) que le message de réponse HTTP se compose d'une ligne d'état, suivie de lignes d'en-tête, suivies d'une ligne vide, suivie du corps de l'entité. Dans le cas de notre HTTP GET, le corps de l'entité dans la réponse est *l'intégralité du* fichier HTML demandé. Dans notre cas ici, le fichier HTML est plutôt long et, à 4500 octets, il est trop volumineux pour tenir dans un paquet TCP. Le message de réponse HTTP unique est donc divisé en plusieurs parties par TCP, chaque élément étant contenu dans un segment TCP distinct (voir figure 1.24 dans le texte). Dans les versions récentes de Wireshark, Wireshark indique chaque segment TCP comme un paquet distinct, et le fait que la réponse HTTP unique ait été fragmentée sur plusieurs paquets TCP est indiqué par le « segment TCP d'une PDU réassemblée » dans la colonne Info de l'affichage Wireshark. Les versions antérieures de Wireshark utilisaient la phrase « Continuation » pour indiquer que tout le contenu d'un message HTTP était divisé sur plusieurs segments TCP. Nous soulignons ici qu'il n'y a pas de message « Continuation » en HTTP!

Répondez aux questions suivantes :

11. Combien de messages de requête HTTP GET votre navigateur a-t-il envoyés ? Quel numéro de paquet dans la trace contient le message GET pour la facture ou les droits ?
12. Quel numéro de paquet dans la trace contient le code d'état et la phrase associés à la réponse à la requête HTTP GET ?
13. Quel est le code d'état et la phrase dans la réponse ?
14. Combien de segments TCP contenant des données ont été nécessaires pour transporter la réponse HTTP unique et le texte de la Déclaration des droits ?

4.HTML Documents avec objets incorporés

Maintenant que nous avons vu comment Wireshark affiche le trafic de paquets capturé pour les fichiers HTML volumineux, nous pouvons regarder 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 image) qui sont stockés sur un ou plusieurs autres serveurs.

Procédez comme suit :

- Démarrez votre navigateur Web et assurez-vous que le cache de votre navigateur est effacé, comme indiqué ci-dessus.
- Démarrez le renifleur de paquets Wireshark
- Entrez l'URL suivante dans votre navigateur http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file4.htmlVotre navigateur doit afficher un court fichier HTML avecdeux images. Ces deux images sont référencées dans le fichier HTML de base. C'est-à-dire que les images elles-mêmes ne sont pas contenues dans le code HTML; au lieu de cela, les URL des images sont contenues dans le fichier HTML téléchargé. Comme indiqué dans le manuel, votre navigateur devra récupérer ces logos à partir des sites Web indiqués. Le logo de notre éditeur est extrait du site web gaia.cs.umass.edu. L'image de la couverture de notre 5ème édition (l'une de nos couvertures préférées) est stockée sur le serveur caite.cs.umass.edu. (Il s'agit de deux serveurs Web différents à l'intérieur de cs.umass.edu).
- Arrêtez la capture de paquets Wireshark et entrez « http » dans la fenêtre display-filter-specification, afin que seuls les messages HTTP capturés soient affichés.
- (*Remarque:* Si vous ne parvenez pas à exécuter Wireshark sur une connexion réseau en direct, vous pouvez utiliser la trace de paquet **http-ethereal-trace-4** pour répondre aux questions ci-dessous; voir note de bas de page 1. Ce fichier de trace a été collecté lors de l'exécution des étapes ci-dessus sur l'un des ordinateurs de l'auteur.)

Répondez aux questions suivantes :

15. Combien de messages de requête HTTP GET votre navigateur a-t-il envoyés ? À quelles adresses Internet ces requêtes GET ont-elles été envoyées ?
16. Pouvez-vous savoir si votre navigateur a téléchargé les deux images en série ou si elles ont été téléchargées à partir des deux sites Web en parallèle? Expliquer.

5 Authentification HTTP

Enfin, essayons de visiter un site Web protégé par mot de passe et examinons la séquence de messages HTTP échangés contre un tel site. L'URL http://gaia.cs.umass.edu/wireshark-labs/protected_pages/HTTP-wireshark-file5.html est protégé par mot de passe. Le nom d'utilisateur est « wireshark-students » (sans les guillemets), et le mot de passe est « network » (encore une fois, sans les guillemets). Accédons donc à ce site « sécurisé » protégé par mot de passe. Procédez comme suit :

- Assurez-vous que le cache de votre navigateur est effacé, comme indiqué ci-dessus, et fermez votre navigateur. Ensuite, démarrez votre navigateur
- Démarrez le renifleur de paquets Wireshark
- Entrez l'URL suivante dans votre navigateur http://gaia.cs.umass.edu/wireshark-labs/protected_pages/HTTP-wireshark-file5.htmlTapez le nom d'utilisateur et le mot de passe demandés dans la fenêtre contextuelle.
- Arrêtez la capture de paquets Wireshark et entrez « http » dans la fenêtre display-filter-specification, de sorte que seuls les messages HTTP capturés seront affichés ultérieurement dans la fenêtre de liste de paquets.

- (*Remarque:* Si vous ne parvenez pas à exécuter Wireshark sur une connexion réseau en direct, vous pouvez utiliser la trace de **paquets http-ethereal-trace-5** pour répondre aux questions ci-dessous; voir note de bas de page 2. Ce fichier de trace a été collecté lors de l'exécution des étapes ci-dessus sur l'un des ordinateurs de l'auteur.)

Examinons maintenant la sortie Wireshark. Vous voudrez peut-être d'abord lire sur l'authentification HTTP en consultant le matériel facile à lire sur "HTTP Access Authentication Framework"à http://frontier.userland. com/stories/storyReader$2159

Répondez aux questions suivantes :

17. Quelle est la réponse du serveur (code d'état et phrase) en réponse au message HTTP GET initial de votre navigateur ?
18. Lorsque votre navigateur envoie le message HTTP GET pour la deuxième fois, quel nouveau champ est inclus dans le message HTTP GET ?

Le nom d'utilisateur (wireshark-students) et le mot de passe (réseau) que vous avez entrés sont codés dans la chaîne de caractères (d2lyZXNoYXJrLXN0dWRlbnRzOm5ldHdvcms=) suivant 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 cryptés, ils sont simplement codés dans un format connu sous le nom de format Base64. Le nom d'utilisateur et le mot de passe ne sont *pas* cryptés! Pour voir cela, accédez à http://www.motobit.com/util/base64-decoder-encoder.asp et entrez la chaîne codée en base64 d2lyZXNoYXJrLXN0dWRlbnRz et décodez. *Voilà!* Vous avez traduit de l'encodage Base64 en codage ASCII, et vous devriez donc voir votre nom d'utilisateur! Pour afficher le mot de passe, entrez le reste de la chaîne Om5ldHdvcms= et appuyez sur décoder. Étant donné que n'importe qui peut télécharger un outil comme Wireshark et renifler des paquets (pas seulement le leur) passant par sa carte réseau, et n'importe qui peut traduire de Base64 en ASCII (vous venez de le faire!), Il devrait être clair pour vous que les mots de passe simples sur les sites WWW ne sont pas sécurisés à moins que des mesures supplémentaires ne soient prises.

N'ayez crainte! Comme nous le verrons au chapitre 8, il existe des moyens de rendre l'accès WWW plus sécurisé. Cependant, nous aurons clairement besoin de quelque chose qui va au-delà du cadre d'authentification HTTP de base!