| Laboratório Wireshark:  Começando v8.1  Material complementar do livro *Computer Networking: A Top-Down Approach, 8th ed.,* J.F. Kurose and K.W. Ross  *“Diga-me e eu esqueço. Mostre-me e eu me lembro. Envolva-me e eu entendo.”* Provérbio chinês  © 2005-2021, J.F Kurose and K.W. Ross, All Rights Reserved | *A picture containing outdoor, water, bridge, building  Description automatically generated* |
| --- | --- |

Tendo experimentado o sniffer de pacotes Wireshark no laboratório introdutório, agora estamos prontos para usar o Wireshark para investigar os protocolos em operação. Neste laboratório, exploraremos vários aspectos do protocolo HTTP: a interação básica GET/resposta, formatos de mensagem HTTP, recuperação de arquivos HTML grandes, recuperação de arquivos HTML com objetos incorporados e autenticação e segurança HTTP. Antes de iniciar esses laboratórios, talvez você queira revisar a Seção 2.2 do texto.[[1]](#footnote-0)

1. A interação HTTP GET/resposta básica

Vamos começar nossa exploração do HTTP baixando um arquivo HTML muito simples - um que é muito curto e não contém objetos incorporados. Faça o seguinte:

1. Inicie o seu navegador da web.
2. Inicie o sniffer de pacotes Wireshark, conforme descrito no laboratório introdutório (mas ainda não inicie a captura de pacotes). Digite “http” (apenas as letras, não as aspas, e em letras minúsculas) na janela de especificação do filtro de exibição, de modo que apenas as mensagens HTTP capturadas sejam exibidas posteriormente na janela de listagem de pacotes. (Estamos interessados apenas no protocolo HTTP aqui e não queremos ver a confusão de todos os pacotes capturados).
3. Aguarde um pouco mais de um minuto (veremos o motivo em breve) e, em seguida, inicie a captura de pacotes do Wireshark.
4. Digite o seguinte no seu navegador
5. <http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file1.html>
6. Seu navegador deve exibir o arquivo HTML de uma linha muito simples.
7. Pare a captura de pacotes do Wireshark.

Sua janela do Wireshark deve ser semelhante à janela mostrada na Figura 1. Se você não conseguir executar o Wireshark em uma conexão de rede ao vivo, você pode baixar um rastreamento de pacote que foi criado quando as etapas acima foram seguidas.[[2]](#footnote-1)

Text, application

Description automatically generated

**Figura 1:** Exibição do Wireshark após [http://gaia.cs.umass.edu/wireshark-labs/ HTTP-wireshark-file1.html](http://gaia.cs.umass.edu/wireshark-labs/%20HTTP-wireshark-file1.html) ter sido recuperado pelo seu navegador

O exemplo na Figura 1 mostra na janela de listagem de pacotes que duas mensagens HTTP foram capturadas: a mensagem GET (do seu navegador para o servidor da web gaia.cs.umass.edu) e a mensagem de resposta do servidor para o seu navegador. A janela de conteúdo do pacote mostra os detalhes da mensagem selecionada (neste caso, a mensagem HTTP OK, destacada na janela de listagem de pacotes). Lembre-se de que, como a mensagem HTTP foi transportada dentro de um segmento TCP, que foi transportado dentro de um datagrama IP, que foi transportado dentro de um quadro Ethernet, o Wireshark exibe as informações do pacote Frame, Ethernet, IP e TCP também. Queremos minimizar a quantidade de dados não-HTTP exibidos (estamos interessados em HTTP aqui e investigaremos esses outros protocolos em laboratórios posteriores). As informações TCP têm um sinal de mais ou um triângulo apontando para a direita (o que significa que há informações ocultas e não exibidas) e a linha HTTP tem um sinal de menos ou um triângulo apontando para baixo (o que significa que todas as informações sobre a mensagem HTTP são exibidas) .

(Observação: você deve ignorar qualquer HTTP GET e resposta para favicon.ico. Se você vir uma referência a este arquivo, é seu navegador perguntando automaticamente ao servidor se ele (o servidor) possui um pequeno arquivo de ícone que deve ser exibido próximo a o URL exibido em seu navegador. Ignoraremos as referências a esse arquivo incômodo neste laboratório.).

Observando as informações nas mensagens HTTP GET e de resposta, responda às seguintes perguntas.[[3]](#footnote-2)

1. Seu navegador está executando HTTP versão 1.0, 1.1 ou 2? Qual versão do HTTP o servidor está executando?
2. Quais idiomas (se houver) seu navegador indica que pode aceitar para o servidor?
3. Qual é o endereço IP do seu computador? Qual é o endereço IP do servidor gaia.cs.umass.edu?
4. Qual é o código de status retornado do servidor para o seu navegador?
5. Quando o arquivo HTML que você está recuperando foi modificado pela última vez no servidor?
6. Quantos bytes de conteúdo estão sendo retornados ao seu navegador?
7. Ao inspecionar os dados brutos na janela de conteúdo do pacote, você vê algum cabeçalho nos dados que não são exibidos na janela de listagem de pacotes? Em caso afirmativo, cite um.

Em sua resposta à pergunta 5 acima (supondo que você esteja executando o Wireshark “ao vivo”, em vez de usar um arquivo de rastreamento gravado anteriormente), você pode ter ficado surpreso ao descobrir que o documento que você acabou de recuperar foi modificado pela última vez um minuto antes você baixou o documento. Isso ocorre porque (para este arquivo em particular), o servidor gaia.cs.umass.edu está definindo a hora da última modificação do arquivo para ser a hora atual, e está fazendo isso uma vez por minuto. Assim, se você esperar um minuto entre os acessos, o arquivo parecerá ter sido modificado recentemente e, portanto, seu navegador fará o download de uma “nova” cópia do documento.

2. A interação HTTP CONDITIONAL GET/resposta

Lembre-se da Seção 2.2.5 do texto, que a maioria dos navegadores da Web executa o cache de objetos e, portanto, geralmente executa um GET condicional ao recuperar um objeto HTTP. Antes de executar as etapas abaixo, verifique se o cache do seu navegador está vazio[[4]](#footnote-3). Agora faça o seguinte:

* Inicie seu navegador da Web e certifique-se de que o cache do navegador esteja limpo, conforme discutido acima.
* Inicie o sniffer de pacotes Wireshark
* Digite o seguinte URL em seu navegador

<http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file2.html>

Seu navegador deve exibir um arquivo HTML de cinco linhas muito simples.

* Digite rapidamente o mesmo URL em seu navegador novamente (ou simplesmente selecione o botão de atualização em seu navegador)
* Pare a captura de pacotes do Wireshark e insira “http” (novamente, em letras minúsculas sem as aspas) na janela de especificação do filtro de exibição, de modo que apenas as mensagens HTTP capturadas sejam exibidas posteriormente na janela de listagem de pacotes.

Se você não conseguir executar o Wireshark em uma conexão de rede ao vivo (ou não conseguir que seu navegador emita um campo If-Modified-Since na segunda solicitação HTTP GET), você pode baixar um rastreamento de pacote que foi criado quando as etapas acima foram seguidos. Responda as seguintes questões:

1. Inspecione o conteúdo da primeira solicitação HTTP GET de seu navegador para o servidor. Você vê uma linha “IF-MODIFIED-SINCE” no HTTP GET?
2. Inspecione o conteúdo da resposta do servidor. O servidor retornou explicitamente o conteúdo do arquivo? Como você sabe?
3. Agora inspecione o conteúdo da segunda solicitação HTTP GET do seu navegador para o servidor. Você vê uma linha “IF-MODIFIED-SINCE:” no HTTP GET? Em caso afirmativo, quais informações seguem o cabeçalho “IF-MODIFIED-SINCE:”?
4. Qual é o código de status HTTP e a frase retornada do servidor em resposta a esse segundo HTTP GET? O servidor retornou explicitamente o conteúdo do arquivo? Explicar.

3. Recuperando Documentos Longos

Em nossos exemplos até agora, os documentos recuperados foram arquivos HTML simples e curtos. Vejamos a seguir o que acontece quando baixamos um arquivo HTML longo. Faça o seguinte:

* Inicie seu navegador da Web e certifique-se de que o cache do navegador esteja limpo, conforme discutido acima.
* Inicie o sniffer de pacotes Wireshark
* Digite o seguinte URL em seu navegador

<http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file3.html>

Seu navegador deve exibir a bastante longa Declaração de Direitos dos EUA.

* Pare a captura de pacotes do Wireshark e insira “http” na janela de especificação do filtro de exibição, para que apenas as mensagens HTTP capturadas sejam exibidas.

Na janela de listagem de pacotes, você deve ver sua mensagem HTTP GET, seguida por uma resposta TCP de vários pacotes à sua solicitação HTTP GET. Certifique-se de que seu filtro de exibição do Wireshark esteja limpo para que a resposta TCP de vários pacotes seja exibida na lista de pacotes.

Essa resposta de pacote múltiplo merece um pouco de explicação. Lembre-se da Seção 2.2 (veja a Figura 2.9 no texto) que a mensagem de resposta HTTP consiste em uma linha de status, seguida por linhas de cabeçalho, seguidas por uma linha em branco, seguida pelo corpo da entidade. No caso de nosso HTTP GET, o corpo da entidade na resposta é todo o arquivo HTML solicitado. No nosso caso aqui, o arquivo HTML é bastante longo e com 4500 bytes é muito grande para caber em um pacote TCP. A única mensagem de resposta HTTP é então dividida em várias partes pelo TCP, com cada parte contida em um segmento TCP separado (consulte a Figura 1.24 no texto). Em versões recentes do Wireshark, o Wireshark indica cada segmento TCP como um pacote separado, e o fato de que a única resposta HTTP foi fragmentada em vários pacotes TCP é indicado pelo “segmento TCP de uma PDU remontada” na coluna Informações da exibição do Wireshark .

Responda as seguintes questões[[5]](#footnote-4):

1. Quantas mensagens de solicitação HTTP GET seu navegador enviou? Qual número de pacote no rastreamento contém a mensagem GET para o Bill ou Rights?
2. Qual número de pacote no rastreamento contém o código de status e a frase associada à resposta à solicitação HTTP GET?
3. Qual é o código de status e a frase na resposta?
4. Quantos segmentos TCP contendo dados foram necessários para transportar a única resposta HTTP e o texto da Declaração de Direitos?

4. Documentos HTML com objetos incorporados

Agora que vimos como o Wireshark exibe o tráfego de pacotes capturados para arquivos HTML grandes, podemos ver o que acontece quando seu navegador baixa um arquivo com objetos incorporados, ou seja, um arquivo que inclui outros objetos (no exemplo abaixo, arquivos de imagem ) que estão armazenados em outro(s) servidor(es).

Faça o seguinte::

* Inicie seu navegador da Web e certifique-se de que o cache do navegador esteja limpo, conforme discutido acima.
* Inicie o sniffer de pacotes Wireshark
* Digite o seguinte URL em seu navegador

<http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file4.html>

* Seu navegador deve exibir um pequeno arquivo HTML com duas imagens. Essas duas imagens são referenciadas no arquivo HTML base. Ou seja, as próprias imagens não estão contidas no HTML; em vez disso, os URLs das imagens estão contidos no arquivo HTML baixado. Conforme discutido no livro, seu navegador terá que recuperar esses logotipos dos sites indicados. O logotipo de nosso editor é obtido do site gaia.cs.umass.edu. A imagem da capa da nossa 8ª edição (uma das nossas capas favoritas) está armazenada em um servidor na França.
* Pare a captura de pacotes do Wireshark e insira “http” na janela de especificação do filtro de exibição, para que apenas as mensagens HTTP capturadas sejam exibidas.

Responda as seguintes questões[[6]](#footnote-5):

1. Quantas mensagens de solicitação HTTP GET seu navegador enviou? Para quais endereços da Internet essas solicitações GET foram enviadas?
2. Você pode dizer se o seu navegador baixou as duas imagens em série ou se elas foram baixadas dos dois sites em paralelo? Explicar.

5. Autenticação HTTP

Por fim, vamos tentar visitar um site protegido por senha e examinar a sequência de mensagens HTTP trocadas por esse site. O URL

<http://gaia.cs.umass.edu/wireshark-labs/protected_pages/HTTP-wireshark-file5.html> é protegido por senha. O nome de usuário é “wireshark-students” (sem as aspas) e a senha é “network” (novamente, sem as aspas). Então vamos acessar este site “seguro” protegido por senha. Faça o seguinte:

* Certifique-se de que o cache do seu navegador esteja limpo, conforme discutido acima, e feche o navegador. Em seguida, inicie o seu navegador
* Inicie o sniffer de pacotes Wireshark
* Digite o seguinte URL em seu navegador

http://gaia.cs.umass.edu/wireshark-labs/protected\_pages/HTTP-wireshark-file5.html

Digite o nome de usuário e a senha solicitados na caixa pop-up.

* Pare a captura de pacotes do Wireshark e insira “http” na janela de especificação do filtro de exibição, de modo que apenas as mensagens HTTP capturadas sejam exibidas posteriormente na janela de listagem de pacotes.
* Observação: Se você não conseguir executar o Wireshark em uma conexão de rede ao vivo, poderá usar o rastreamento de pacote http-ethereal-trace-5 “clássico” ou outros rastreamentos adicionais, como notas na nota de rodapé 2, para responder às perguntas abaixo.

Agora vamos examinar a saída do Wireshark. Você pode querer primeiro ler sobre autenticação HTTP revisando o material fácil de ler em “HTTP Access Authentication Framework” em <http://frontier.userland.com/stories/storyReader$2159>

Responda as seguintes questões abaixo[[7]](#footnote-6):

1. Qual é a resposta do servidor (código e frase de status) em resposta à mensagem HTTP GET inicial do seu navegador?
2. Quando o seu navegador envia a mensagem HTTP GET pela segunda vez, que novo campo é incluído na mensagem HTTP GET?

O nome de usuário (wireshark-students) e a senha (rede) que você inseriu são codificados na sequência de caracteres (d2lyZXNoYXJrLXN0dWRlbnRzOm5ldHdvcms=) após o cabeçalho “***Authorization: Basic***” na mensagem HTTP GET do cliente. Embora possa parecer que seu nome de usuário e senha estão criptografados, eles estão simplesmente codificados em um formato conhecido como formato Base64. O nome de usuário e a senha não são criptografados! Para ver isso, vá para <http://www.motobit.com/util/base64-decoder-encoder.asp> e digite a string codificada em base64 d2lyZXNoYXJrLXN0dWRlbnRz e decodifique. Voilá! Você traduziu da codificação Base64 para a codificação ASCII e, portanto, deve ver seu nome de usuário! Para visualizar a senha, insira o restante da *string* Om5ldHdvcms= e pressione *decode*. Como qualquer um pode baixar uma ferramenta como o Wireshark e farejar pacotes (não apenas os seus) passando por seu adaptador de rede, e qualquer um pode traduzir de Base64 para ASCII (você acabou de fazer isso!), deve ficar claro para você que senhas simples em WWW os sites não são seguros, a menos que medidas adicionais sejam tomadas.

Não tema! Como veremos no Capítulo 8, existem maneiras de tornar o acesso à WWW mais seguro. No entanto, claramente precisaremos de algo que vá além da estrutura básica de autenticação HTTP!

1. As referências a figuras e seções referem-se à 8ª edição de nosso texto, Computer Networks, A Top-down Approach, 8ª ed., J.F. Kurose e K.W. Ross, Addison-Wesley/Pearson, 2020. O site dos autores deste livro é <http://gaia.cs.umass.edu/kurose_ross>. Você encontrará muito material aberto interessante lá. [↑](#footnote-ref-0)
2. Você pode baixar o arquivo zip <http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces-8.1.zip> e extrair o arquivo de rastreamento http-wireshark-trace1-1. Esses arquivos de rastreamento podem ser usados para responder a essas perguntas de laboratório do Wireshark sem realmente capturar pacotes por conta própria. Cada traço foi feito usando o Wireshark rodando em um dos computadores do autor, durante a execução das etapas indicadas no laboratório Wireshark. Depois de baixar um arquivo de rastreamento, você pode carregá-lo no Wireshark e visualizar o rastreamento usando o menu suspenso Arquivo, escolhendo Abrir e, em seguida, selecionando o nome do arquivo de rastreamento. A exibição resultante deve ser semelhante à Figura 1 (para o arquivo de rastreamento **http-wireshark-trace1-1** deste laboratório HTTP). A interface do usuário do Wireshark é exibida de maneira um pouco diferente em diferentes sistemas operacionais e em diferentes versões do Wireshark. [↑](#footnote-ref-1)
3. Para esta atividade, ao responder às perguntas a seguir com tarefas entregues, os alunos imprimem o GET e as mensagens de resposta (consulte o laboratório introdutório do Wireshark para obter uma explicação de como fazer isso) e indicam onde na mensagem eles encontraram o informações que respondem a uma pergunta. [↑](#footnote-ref-2)
4. See <https://www.howtogeek.com/304218/how-to-clear-your-history-in-any-browser/> for instructions on clearing your browser cache. [↑](#footnote-ref-3)
5. Se você não conseguir executar o Wireshark em uma conexão de rede ao vivo, você pode baixar o arquivo zip <http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces-8.1.zip> e extrair o arquivo de rastreamento http-wireshark-trace3-1. [↑](#footnote-ref-4)
6. Se você não conseguir executar o Wireshark em uma conexão de rede ao vivo, você pode baixar o arquivo zip <http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces-8.1.zip> e extrair o arquivo de rastreamento http- wireshark-trace4-1. [↑](#footnote-ref-5)
7. Se você não conseguir executar o Wireshark em uma conexão de rede ao vivo, você pode baixar o arquivo zip <http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces-8.1.zip> e extrair o arquivo de rastreamento http- wireshark-trace5-1. [↑](#footnote-ref-6)