

## Instruções para o desenvolvimento do Trabalho

Disciplina: COM241 – Administração e Gerência de Redes de Computadores

Professor: Bruno Guazzelli Batista

Email: [brunoguazzelli@unifei.edu.br](mailto:brunoguazzelli@unifei.edu.br)

Neste trabalho, vamos explorar diversos aspectos do protocolo HTTP: a interação básica do GET, formatos de mensagens HTTP, requisitar arquivos grandes em HTML, requisitar objetos incluídos em arquivos HTML. Também veremos aspectos de autenticação e segurança do HTTP. Para o desenvolvimento do trabalho, siga as instruções presentes nesse documento. O trabalho possui valor de 4 pontos e deve ser desenvolvido em duplas. As respostas devem ser enviadas por meio da tarefa cadastrada no SIGAA em um único documento, destacando as informações requeridas e analisadas.

**Atrasos não serão tolerados.** Não se esqueça de colocar os nomes e os números de matrículas de cada integrante.

### Atividade 1. O Básico do GET do HTTP/interação de resposta

Vamos começar explorando o HTTP fazendo o download de um arquivo HTML muito simples – um arquivo pequeno e que não contém referências para objetos. Faça o seguinte:

1. Inicie o navegador Web.
2. Inicie o sniffer de pacotes, Wireshark, mas não inicie a captura de pacotes. Digite “http” (somente as letras sem as aspas) na janela display-filter-specification, assim somente as mensagens HTTP serão mostradas pela ferramenta.
3. Espere um instante antes de iniciar a captura de pacotes.
4. Entre o endereço abaixo no navegador:  
<http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file1.html>  
O navegador deve mostrar uma página bem simples com uma linha.
5. Pare a captura de pacotes do Wireshark.

Se não estiver apto a rodar o Wireshark em uma conexão de rede, você pode fazer o download do pacote que foi criado, quando os passos acima foram executados. Para isso, baixe o arquivo wireshark-traces.zip disponível no SIGAA e extraia o arquivo http-ethereal-trace-1. Após baixar o arquivo, você pode abrir no Wireshark e ver as informações usando File -> Open e selecionando o arquivo http-ethereal-trace-1.

Após executar os passos anteriores, a janela de listagem mostrará mensagens HTTP capturadas: uma mensagem de GET (de um navegador para o servidor gaia.cs.umass.edu) e uma mensagem de resposta do servidor para o navegador. A janela com o conteúdo do pacotes mostra detalhes da mensagem selecionada (no caso, a mensagem HTTP OK). Lembre-se de que a mensagem HTTP é carregada num segmento TCP, o qual está num datagrama IP, o qual está num quadro Ethernet. O Wireshark mostra informações do quadro, Ethernet, IP e TCP. Queremos minimizar a quantidade de dados que não são HTTP, então mostre apenas as informações de HTTP no Wireshark e esconda o restante. (Nota: Você deve ignorar qualquer GET HTTP e resposta para favicon.ico. Se observar referência para este arquivo, é o navegador solicitando automaticamente para o servidor se o servidor possui um pequeno ícone para ser mostrado próximo à URL no navegador.)

Observando a informação do GET HTTP e a mensagem de resposta, responda as questões abaixo. Quando responder, inclua uma impressão das mensagens de GET e a resposta e indicar onde você encontrou as respostas.

1. O seu navegador está rodando a versão 1.0 ou 1.1 do HTTP? Que versão do HTTP está sendo rodada no servidor?

2. Que linguagem (se tiver) o seu navegador indica que ele pode aceitar do servidor?
3. Qual é o endereço IP do seu computador? E do servidor gaia.cs.umass.edu?
4. Qual é o código de estado retornado pelo servidor para o seu navegador?
5. Quando foi a última alteração feita no arquivo HTML?
6. Quantos bytes de conteúdo são retornados para o seu navegador?
7. Inspeccionando o dado na janela de conteúdo do pacote, você viu algum cabeçalho com dados que não mostrados na janela de listagem do Wireshark? Se sim, qual?

## **Atividade 2. O GET HTTP CONDITIONAL/interação de resposta**

Um cache Web, também denominado servidor proxy, é uma entidade da rede que atende requisições HTTP em nome de um servidor Web de origem. Para isso, ele utiliza um GET condicional, um mecanismo que permite que um cache verifique se seus objetos estão atualizados.

Antes de fazer esta atividade, certifique-se de que o cache do navegador esteja limpo (faça isso usando a opção para apagar o histórico do navegador). Agora, faça o seguinte:

1. Inicie o navegador e certifique-se de que o cache esteja limpo.
2. Inicie o Wireshark.
3. Entre com a URL no navegador  
<http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file2.html>  
Seu navegador deve mostrar um arquivo bem simples HTML com cinco linhas.
4. Rapidamente entre com a URL no navegador novamente (ou simplesmente selecione o botão para recarregar a página).
5. Pare o Wireshark e entre com “http” na janela em que se especifica filtros para que somente as mensagens HTTP possam ser mostradas na janela de listagem de pacotes.

Se não estiver apto a rodar o Wireshark em uma conexão de rede, você pode fazer o download do pacote que foi criado, quando os passos acima foram executados. Para isso, baixe o arquivo wireshark-traces.zip disponível no SIGAA e extraia o arquivo http-ethereal-trace-2. Após baixar o arquivo, você pode abrir no Wireshark e ver as informações usando File -> Open e selecionando o arquivo http-ethereal-trace-2.

Responda às seguintes questões:

1. Inspecione o conteúdo da primeira requisição GET do HTTP de seu navegador para o servidor. Você viu uma linha com “IF-MODIFIED-SINCE” no GET do HTTP?
2. Inspecione o conteúdo da resposta do servidor. O servidor retornou o conteúdo do arquivo explicitamente? Como isso se observou?
3. Agora investigue o conteúdo da segunda requisição GET do HTTP de seu navegador para o servidor. Você viu uma linha com “IF-MODIFIED-SINCE” no GET do HTTP? Se sim, que informação segue o cabeçalho “IF-MODIFIED-SINCE:”?
4. Qual é o código de estado do HTTP e a frase retornada do servidor em resposta ao segundo GET do HTTP? O servidor explicitamente retornou o conteúdo do arquivo? Explique!

## **Atividade 3. Obtendo longos documentos**

Em nossos exemplos até aqui, os documentos HTML obtidos eram simples e curtos. Vamos analisar o que acontece quando trabalharmos com arquivos HTML longos. Faça o seguinte:

1. Inicie o seu navegador e execute a limpeza do cache, como discutido acima.
2. Inicie o Wireshark.
3. Entre com a seguinte URL no navegador  
<http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file3.html>  
Seu navegador deve mostrar o documento “US Bill of Rights”.

4. Pare a captura de pacotes pelo Wireshark e entre com “http” na janela em que se especifica filtros para que somente as mensagens HTTP possam ser mostradas na janela de listagem de pacotes.

Se não estiver apto a rodar o Wireshark em uma conexão de rede, você pode fazer o download do pacote que foi criado, quando os passos acima foram executados. Para isso, baixe o arquivo wireshark-traces.zip disponível no SIGAA e extraia o arquivo http-ethereal-trace-3. Após baixar o arquivo, você pode abrir no Wireshark e ver as informações usando File -> Open e selecionando o arquivo http-ethereal-trace-3.

Na janela de listagem de pacotes do Wireshark, você pode ver a mensagem GET do HTTP, seguida de respostas TCP em pacotes múltiplos para a requisição do GET do HTTP. A resposta em pacotes múltiplos merece uma explicação: a mensagem de resposta HTTP consiste de uma linha de estado, seguida por linhas de cabeçalho, seguida por uma linha em branco, seguida pelo corpo da entidade. No caso do nosso GET do HTTP, o corpo da entidade na resposta é todo o arquivo HTML requisitado. Neste caso, o arquivo HTML é longo e os 4500 bytes são grandes para caber no pacote TCP. A única resposta HTTP é então quebrada em diversas peças pelo TCP, com cada parte sendo contida em um segmento TCP separado (fragmento). Nas recentes versões do Wireshark, ele indica cada segmento TCP como um pacote separado e o fato de uma única resposta HTTP fragmentada em múltiplos pacotes TCP é indicada por “TCP segment of a reassembled PDU” na coluna Info da tela do Wireshark. Versões mais antigas usavam a frase “Continuation” para indicar que o conteúdo integral de uma mensagem HTTP foi quebrado em múltiplos segmentos TCP.

Responda às seguintes questões:

1. Quantas mensagens de requisição GET de HTTP o seu navegador enviou? Qual número de pacote contém a mensagem de GET do “Bill or Rights”?
2. Qual número de pacote contém o código de estado e a frase associada com a resposta do pedido de GET de HTTP?
3. Qual é o código de estado e a frase de resposta?
4. Quantos segmentos TCP contendo dados são necessários para uma simples resposta HTTP para o texto?

#### **Atividade 4. Documentos HTML com objetos incluídos**

Agora que você viu com o Wireshark mostra o tráfego de pacotes capturados para arquivos HTML grandes, nós podemos ver o que acontece quando um navegador baixa um arquivo com objetos incluídos, isto é, um arquivo que inclui outros objetos (no exemplo abaixo, arquivos de imagens) que estão armazenados em outros servidores. Faça o seguinte:

1. Inicie o seu navegador e execute a limpeza do cache, como discutido acima.
2. Inicie o Wireshark.
3. Entre com a seguinte URL no navegador

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

Seu navegador deve mostrar um pequeno arquivo HTML com duas imagens. Essas imagens são referenciadas no arquivo HTML base. Isto é, as imagens propriamente não estão contidas no HTML, ao invés disso, as URLs para as imagens estão contidas no arquivo HTML baixado.

4. Pare a captura de pacotes pelo Wireshark e entre com “http” na janela em que se especifica filtros para que somente as mensagens HTTP possam ser mostradas na janela de listagem de pacotes.

Se não estiver apto a rodar o Wireshark em uma conexão de rede, você pode fazer o download do pacote que foi criado, quando os passos acima foram executados. Para isso, baixe o

arquivo wireshark-traces.zip disponível no SIGAA e extraia o arquivo http-ethereal-trace-4. Após baixar o arquivo, você pode abrir no Wireshark e ver as informações usando File -> Open e selecionando o arquivo http-ethereal-trace-4.

Responda às seguintes questões:

1. Quantas mensagens de requisição HTTP o seu navegador enviou? Para quais endereços da Internet foram enviados esses pedidos GET?
2. Você pode falar se o navegador baixou as duas imagens serialmente ou se elas foram baixadas dos dois sites paralelamente? Explique!

### Atividade 5. Autenticação HTTP

Finalmente, vamos visitar um site web protegido por senha e examinar a sequência de mensagens HTTP trocadas com esse site. A URL [http://gaia.cs.umass.edu/wireshark-labs/protected\\_pages/HTTP-wireshark-file5.html](http://gaia.cs.umass.edu/wireshark-labs/protected_pages/HTTP-wireshark-file5.html) tem proteção por senha. O 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:

1. Inicie o seu navegador e execute a limpeza do cache, como discutido acima.
2. Inicie o Wireshark.
3. Entre com a seguinte URL no navegador  
[http://gaia.cs.umass.edu/wireshark-labs/protected\\_pages/HTTP-wireshark-file5.html](http://gaia.cs.umass.edu/wireshark-labs/protected_pages/HTTP-wireshark-file5.html)  
Digite o usuário e a senha nos campos apropriados.
4. Pare a captura de pacotes pelo Wireshark e entre com “http” na janela em que se especifica filtros para que somente as mensagens HTTP possam ser mostradas na janela de listagem de pacotes.

Se não estiver apto a rodar o Wireshark em uma conexão de rede, você pode fazer o download do pacote que foi criado, quando os passos acima foram executados. Para isso, baixe o arquivo wireshark-traces.zip disponível no SIGAA e extraia o arquivo http-ethereal-trace-5. Após baixar o arquivo, você pode abrir no Wireshark e ver as informações usando File -> Open e selecionando o arquivo http-ethereal-trace-5.

Agora, vamos examinar a saída do Wireshark. Você deve primeiramente querer entender melhor sobre autenticação HTTP. Você pode ver o material “HTTP Access Authentication Framework” em [http://frontier.userland.com/stories/storyReader\\$2159](http://frontier.userland.com/stories/storyReader$2159)

Responda às seguintes questões:

1. Qual é a resposta do servidor (código de estado e frase) na resposta ao GET de HTTP inicial do navegador?
2. Quando seu navegador envia mensagem GET de HTTP pela segunda vez, que novos campos são incluídos na mensagem GET de HTTP?

O usuário (wireshark-students) e a senha (network) que você entrou são codificados para a cadeia de caracteres (d2lyZXNoYXJrLXN0dWRLbnRzOm5ldHdvcm5l) seguidos do cabeçalho “Authorization: Basic” na mensagem de GET do HTTP do cliente. Enquanto isso parece que o seu usuário e a sua senha estão criptografados, eles, na verdade, estão codificados no formato conhecido como formato Base64. O usuário e a senha não estão criptografados! Para ver isso vá para <http://www.motobit.com/util/base64-decoder-encoder.asp> e entre com a cadeia de caracteres codificadas na base64 d2lyZXNoYXJrLXN0dWRLbnRz e decodifique. Voila! Você acabou de traduzir uma codificação Base64 para codificação ASCII e o resultado deve ter sido o usuário! Para ver a senha, entre com o restante da cadeia Om5ldHdvcm5l e pressione decodificar. Desde que qualquer um pode baixar uma ferramenta como o Wireshark e capturar pacotes (não somente os

seus próprios pacotes) passando pela rede e qualquer um pode traduzir da Base64 para ASCII (você acabou de fazer isso!), está claro que senhas simples na WWW não são seguras a menos que medidas adicionais sejam tomadas.