Informação e Comunicação 2019-20

Lab B - Experiências com Consola Remota e Introdução ao Wireshark

Notas prévias

Na realização dos trabalhos práticos de Redes, considere o documento “Introdução a Redes de Comunicação”, disponível na secção “Redes - Laboratórios” do Moodle, como de leitura obrigatória e referência primária, no caso de dúvidas. Ao longo do trabalho, identificamos este documento como IRC sempre que seja incluída uma referência a uma das suas secções, de acordo com o formato [IRC, Secção x.y].

As experiências com consola descritas nas secções 1 e 2 deste trabalho preliminar só estão disponíveis dentro da FEUPnet, podendo ser efetuadas nos computadores instalados nos laboratórios da FEUP ou remotamente através de uma sessão estabelecida para um computador ou servidor da FEUP.

Considerando a situação atual de aulas práticas efetuadas exclusivamente de forma não presencial, o estabelecimento de sessões remotas nas experiências com consola é indispensável, devendo proceder da seguinte forma:

- Crie uma ligação VPN para a FEUP.
- Abra uma janela de terminal (Terminal ou *cmd*) do sistema operativo Linux, Windows ou iOS da máquina local que está a usar.
- Crie nessa janela uma sessão por SSH (*Secure Shell*) para os servidores *gnomo.fe.up.pt* ou *yoda.fe.up.pt* usando o comando

  ```
  ssh <username@hostname>.
  ```

  Caso o cliente ssh não esteja instalado, poderá usar, por exemplo, a aplicação de código aberto PuTTY.
- Uma vez criada a ligação a um desses servidores, introduza as suas credenciais de acesso aos serviços da FEUP para abrir uma sessão em modo de consola remota.

As experiências da secção 3, em que vamos começar a utilizar a ferramenta *wireshark*, terão de ser executadas diretamente sobre a máquina local, isto é, sem ligação VPN, já que este serviço cria um túnel de dados encriptados que não poderiam ser analisados no *wireshark*. Os resultados das experiências podem variar um pouco dependendo do *browser* e hardware usado, e ainda de se está a usar uma ligação com ou sem fios.

Como boa prática de trabalho laboratorial, passe a utilizar um *logbook* no qual deverá registar os resultados intermédios, as observações e as conclusões que vai obtendo no decorrer das experiências.

1. Transferência de ficheiro por TELNET e FTP

O protocolo de aplicação TELNET [IRC, Secção 5.3] é usado para abrir uma ligação e enviar comandos para um serviço numa máquina remota. É um protocolo baseado em texto, que pode ser usado para interagir com servidores que usem protocolos de aplicação igualmente baseados em texto, como são o FTP (transferência de ficheiros) e o HTTP [IRC, Secção 5.1] (acesso a páginas web).

O FTP usa uma ligação TCP [IRC, Secção 6.3] para controlo, i.e. envio de comandos e receção de respostas curtas. Para cada ficheiro a transferir ou resposta a um pedido de listagem de conteúdos de uma pasta, abre uma nova ligação TCP. Por isso, serão necessárias duas janelas para efetuar esta experiência, a *janela_A* para comandos e a *janela_B* para dados. Estas janelas deverão ser previamente abertas para os servidores SSH da FEUP, conforme indicado na secção “Notas prévias”.

a. Na *janela_A*, escreva a seguinte sequência de comandos (note que, daqui em diante, os comandos indicados a seguir ao símbolo “>” são enviados, e os indicados a seguir ao símbolo “<” são recebidos):

```
> telnet 192.168.109.136 21
> user iclab
< 331 Password required for iclab
> pass lab1920
< 230 User iclab logged in
```

Nesta altura está concluído o login no servidor FTP.

Observe agora na janela A as mensagens que foram enviadas pelo cliente (a sua máquina) e pelo servidor (a máquina remota).

```
> pasv
```

Com o comando *pasv*, pede-se ao servidor de FTP para usar o modo passivo na transferência de dados, ou seja, será o cliente que irá estabelecer a ligação de dados.

O servidor responde que entrou no modo passivo, indicando nos parâmetros “x” devolvidos o endereço IP do servidor e a porta que deverá ser usada (ver ponto b)

```
< 227 Entering Passive Mode (x1,x2,x3,x4,x5,x6)
```

Nesta situação, o cliente ficará responsável pela abertura da ligação TCP para os dados, para esse endereço, nessa porta.

Anote no seu *logbook* o valor retornado dos parâmetros devolvidos pelo servidor.

b. Calcule o valor da porta do servidor a usar na segunda ligação como se indica a seguir.

Na resposta acima, o servidor envia 6 bytes (exemplo: 193,168, 109, 250, 122, 50) com o seguinte significado:

- x1, x2, x3, x4: endereço IP do servidor;
- x5, x6: porta em que o servidor se encontra à espera de ligação.

Estes dois últimos bytes representam o número da porta a usar em base 256. Ou seja, em base decimal, deverão ser interpretados da seguinte forma: *porta* = x5*256 + x6.

No caso do exemplo, será *porta* = 122*256 + 50 = 31282.

c. Na *janela_B* anteriormente criada, abra uma nova ligação para o mesmo servidor e para a porta que calculou.

```
> telnet 192.168.109.136 <nova porta>
< ...
```

Após o estabelecimento da nova ligação na *janela_B*, escreva na *janela_A* o comando que lista os ficheiros disponíveis no diretório corrente da máquina remota.

```
> list
```

Interprete o que observou na *janela_B*, identificando o ficheiro de texto que está disponível no servidor.

d. Execute de novo na *janela A* o comando

```
> pasv
```

Seguidamente, determine o novo valor da porta e abra uma nova ligação na *janela B* para o mesmo servidor e para a porta que calculou.

```
> telnet 192.168.109.136 <nova porta>
< ...
```

Após o estabelecimento da nova ligação na *janela_B*, escreva na *janela_A* o comando que inicia o envio de um ficheiro.

```
> retr <nome_do_ficheiro>
```

Observe o ficheiro recebido na *janela_B* (cada janela funciona como a extremidade de uma ligação TCP). Que imagem e que mensagem viu na janela onde recebeu o ficheiro?

e. Conclua este trabalho encerrando as ligações ao servidor nas janelas A e B com o comando seguinte.

```
> quit
```

2. Acesso a uma página web por TELNET e HTTP

Usando a ferramenta TELNET e comandos GET do protocolo HTTP [IRC, Secção 5.1], iremos agora tentar ligar-nos a servidores para aceder a páginas Web.

Nestas experiências, mantenha aberta a ligação VPN à FEUPnet e estabeleça uma sessão em modo de consola remota numa das máquinas indicadas na secção “Notas prévias”.

NOTA 1: Nos comandos que se seguem, não se esqueça de incluir uma linha em branco adicional a seguir ao comando `HOST`, isto é, `<hostname>` seguido de 2 `"enter"`. Respeite escrupulosamente os espaços e maiúsculas/minúsculas de cada comando.

Para cada alínea, observe se a resposta é a esperada (listagem do código HTML de uma página web) ou se ocorreu algum erro após o comando TELNET ou após o pedido GET, ou se a página devolvida é a que tentava aceder.

NOTA 2: Quando se pretende aceder a uma página *nome.html* alojada numa determinada pasta do sistema de ficheiros do servidor, definida pelo respetivo caminho, a sintaxe genérica do pedido GET é

```
GET /caminho/nome.html HTTP/1.1
```

A omissão do nome do ficheiro com o comando

```
GET / HTTP/1.1
```

conduz a que seja acedida a página *index.html* alojada na raiz do sistema de ficheiros do servidor. Ou seja, aceder com um browser a um servidor Web através do endereço `http://<hostname>` é equivalente a aceder diretamente à página `http://<hostname>/index.html` → faça esta experiência com um qualquer servidor *web*.

Use os comandos *ping*, *nslookup* e *traceroute* [IRC, Secção 8.1] aplicados ao servidor para verificar a conetividade em cada caso, tentando igualmente diagnosticar se, nos casos em que ocorrem erros, estes resultam de algum problema de conetividade:

- `ping <hostname>` - verifica a ligação à máquina remota e devolve a estimativa de RTT (*Round-Trip Time*) a partir de um conjunto de pacotes de teste enviados e respetivas respostas recebidas (termine o teste com ctrl+c);
- `nslookup <hostname>` - permite obter informações sobre registos de DNS de um determinado domínio, máquina ou IP;

- `traceroute <hostname>` - permite seguir o caminho (a rota) percorrido pelos pacotes IP em direção a uma máquina destino, mostrando os endereços IP (e por vezes os nomes) dos interfaces de entrada dos vários *routers* por onde os pacotes passam, e o RTT até cada um (alguns *routers* não enviam os pacotes de teste esperados, ocorrendo asteriscos na listagem devolvida pelo comando traceroute).

Começaremos por aceder por telnet a uma página Web, contendo um conhecido poema de António Gedeão, nos casos A e B, e ao servidor web da FEUP, no caso C. Utilize igualmente um *browser* para tentar visualizar as páginas indicadas e observe o resultado.

Numa janela de consola remota, aberta numa sessão SSH estabelecida com um servidor da FEUP, execute os seguintes comandos, observando e interpretando os resultados:

```
A. > telnet ark.fe.up.pt 80
   > GET /ic/testeIC1.html HTTP/1.1
   > HOST: ark.fe.up.pt
   > "enter" adicional - linha em branco
```
➔ Experimente aceder à página http://ark.fe.up.pt/ic/testeIC1.html num *browser*

```
B. > telnet netlab2.fe.up.pt 80
   > GET /ic/testeIC2.html HTTP/1.1
   > HOST: netlab2.fe.up.pt
   > "enter" adicional - linha em branco
```
➔ Experimente aceder à página http://netlab2.fe.up.pt/ic/testeIC2.html num *browser*

```
C. > telnet www.up.pt 80
   > GET / HTTP/1.1
   > HOST: www.up.pt
   > "enter" adicional - linha em branco
```
➔ Experimente aceder à página http://www.up.pt/ num *browser*

Nas alíneas A, B e C, leia com atenção o resultado da tentativa de acesso por TELNET, identificando na resposta HTTP os elementos do código HTML, através das respetivas etiquetas (tags) que são usadas em cada caso (as tags definem o tipo de elemento e servem para marcar onde este começa e termina). Use uma ferramenta de pesquisa online para perceber o significado das tags e dos atributos (campos específicos) presentes nos elementos HTML identificados. Clarifique o que é diferente em cada caso, do ponto de vista do conteúdo dos elementos HTML e do efeito para o utilizador quando acede por telnet ou através de um *browser*.

Nas tentativas seguintes vão existir diversas situações de erros, que deverá diagnosticar analisando a resposta na consola e recorrendo aos comandos *ping*, *nslookup* e *traceroute* - logo que ocorra o erro, terá de suspender a sequência.

```
D. > telnet xlab.fe.up.pt 80
   > GET / HTTP/1.1
   > HOST: xlab.fe.up.pt
   > "enter" adicional - linha em branco

E. > telnet www.fe.up.pt 21
   > GET / HTTP/1.1
   > HOST: www.fe.up.pt
   > "enter" adicional - linha em branco

F. > telnet 192.168.109.136 21
   > GET / HTTP/1.1
   > HOST: 192.168.109.250
   > "enter" adicional - linha em branco
```
(esta máquina é a que foi utilizada na experiência de transferência de ficheiro por FTP)

3. Utilização da ferramenta *wireshark*

Vamos seguidamente introduzir a ferramenta *wireshark*, que permite fazer a captura de pacotes numa interface de rede, e, deste modo, analisar a operação dos diversos protocolos de comunicação envolvidos.

Deverá, neste caso, realizar as experiências na sua máquina local, sem a ligação anteriormente estabelecida à VPN da FEUP.

É absolutamente essencial que consulte o documento "Introdução a Redes de Comunicação", onde são apresentadas diversas explicações sobre o funcionamento do *wireshark*. Assim, deverá analisar cuidadosamente a seguinte secção desse documento antes de prosseguir com este trabalho.

→ Introdução a Redes de Comunicação - Secção 8.2 - Wireshark

Na primeira vez que utilizar o *wireshark*, recomendamos que efetue duas pequenas alterações da configuração inicial que facilitarão a análise de registos, tal como descrito nas duas últimas subsecções de [IRC, Secção 8.2], intituladas:

- Configuração de colunas da lista de pacotes (acrescente colunas adicionais para o número das portas do protocolo de transporte);
- Configuração de reassemblagem de segmentos TCP;

O primeiro exemplo que iremos tratar neste trabalho corresponde a uma captura relativa a um teste de conetividade com o comando `ping`, que é exatamente o exemplo ilustrado em [IRC, Secção 8.2].

3.1 Captura de tráfego num teste de conetividade

Execute a seguinte sequência de operações:

- Feche todas os aplicativos, incluindo, se possível os aplicativos de troca de mensagens instantâneas (*facebook*, *messenger*, *hangouts*, etc.) e aplicativos de partilha de ficheiros (*dropbox*, *gdrive*, etc.).

  A manutenção de aplicativos abertos não impede a realização dos trabalhos, mas poderá criar tráfego desnecessário para o objetivo de análise. Embora sendo boa prática estabelecer filtros que eliminam tráfego não relevante (ver subsecção "Filtragem da lista de pacotes" em [IRC, Secção 8.2]), a sua construção poderá ser mais complexa se ocorrer uma grande variedade de tráfego não relevante para a captura.

- Abra uma janela de terminal (Terminal ou *cmd*) do sistema operativo Linux, Windows ou iOS da máquina local que está a usar.
- Execute nessa janela o comando do seu sistema operativo que limpa os dados na cache do DNS:

Linux (Ubuntu):	`> sudo systemd-resolve --flush-caches`
Windows:	`> ipconfig /flushdns`
iOS (2 comandos):	`> dscacheutil -flushcache` `> sudo killall -HUP mDNSResponder`

  Como pretendemos obter na captura o tráfego correspondente à resolução do endereço de um servidor, é essencial que essa informação não esteja em cache, caso contrário a consulta não seria efetuada. Este comando terá de ser repetido sempre que efetue novas tentativas de captura.

- Inicie a ferramenta *wireshark*, selecione a interface de rede (Ethernet ou WiFi), inicie uma captura e acompanhe na janela do *wireshark*, em tempo real, o efeito das ações seguintes.
- Execute na janela *cmd* o comando `ping www.google.pt` e aguarde até obter a

estatística final de conetividade (em Windows10, tal ocorre automaticamente ao fim de 4 envios de pedidos de `ping`; em Linux, terá de terminar o procedimento manualmente com ctrl+c).

- Termine a captura.
- Identifique o endereço IP do servidor na captura e no teste de conetividade - deverá ser um endereço IPV6, mas em certas configurações mais antigas de hardware e software poderá ser um endereço IPv4.
- Aplique no campo do *wireshark* do filtro de visualização (linha acima da listagem de pacotes) a expressão seguinte, em que deverá substituir <ipv6 address> pelo endereço que identificou no passo anterior (não esquecer de fazer "enter" após colocar nessa linha a expressão do filtro):

  dns.qry.name == "www.google.pt" or ipv6.addr==<ipv6 address>

  Com este filtro, restringimos a listagem de pacotes correspondentes aos pedidos e respostas de resolução do endereço "www.google.pt", trocados com o servidor DNS, e aos pacotes enviados PING Request e recebidos PING Reply do servidor que contêm o endereço IPv6 identificado, nos campos de destino ou origem.

  Caso o acesso tenha utilizado endereços IPv4, substitua ipv6.addr==<ipv6 address> por ip.addr==<ipv4 address>.
- Guarde a captura que fez num ficheiro para a analisar em próximas oportunidades (mantenha a extensão .pcapng definida pelo Wireshark, por omissão).
- Analise o registo, quer através da listagem de pacotes exibida na vista geral, quer acedendo aos detalhes de cada pacote nas janelas inferiores ou fazendo duplo clique na linha correspondente da vista geral.

Responda seguidamente às seguintes questões escrevendo a resposta no seu *logbook*:

a. Qual a função dos pacotes identificados como associados ao protocolo DNS [IRC, Secção 5.2]?
b. Quais os endereços IP da máquina local e do servidor usados no pedido de resolução de endereço? São endereços IPv4 ou IPv6?

   Na configuração da interface de rede que está a usar, o DNS poderá estar definido com configuração automática ou manual. No caso de estar ativada a configuração automática, o IP para o qual é enviado o pedido de DNS deverá corresponder à *gateway* da sua rede local. No caso de a configuração ser manual, por exemplo, indicando um servidor público de DNS da Google, então o IP que identificará no registo será o correspondente a esse servidor.
c. Qual o protocolo de transporte usado pelo protocolo DNS e quais as portas utilizadas?

   Note que o valor da porta que identificou do lado do servidor DNS deverá coincidir com a porta reservada para este protocolo, referida em [IRC, Secção 6.1].
d. Qual o protocolo usado pela ferramenta `ping` e por que razão na listagem de pacotes não encontra nenhuma referência às portas de origem e de destino?
e. Quantos pares de pedidos e respostas identifica no registo, resultantes do comando `ping`? Confronte esse resultado com o que observou na janela de terminal.
f. Qual o endereço IP da máquina local e o do servidor *web* usados no teste de conetividade com `ping`? São endereços IPv4 ou IPv6?

3.2 Captura de tráfego no acesso a uma página *web* através de um *browser*

Vamos agora efetuar uma captura no acesso à página *web* http://info.cern.ch através de um *browser*. Este servidor é particularmente interessante porque corresponde ao primeiro endereço

da história de *web*, e contém a primeira página que foi criada nos laboratórios do CERN, com uma versão com o mesmo aspeto gráfico da versão original criada em 1989 (imperdível!) e ainda uma breve referência ao nascimento da *web* e ao seu inventor, Tim Berners-Lee.

Execute a seguinte sequência de operações:

- Feche todas os aplicativos, incluindo, se possível, os aplicativos de troca de mensagens instantâneas (*facebook*, *messenger*, *hangouts*, etc) e aplicativos de partilha de ficheiros (*dropbox*, *gdrive*, etc.)
- Abra uma janela de terminal do sistema operativo Linux ou Windows.
- Execute nessa janela o comando de limpeza da cache de DNS (ver secção anterior).

  Mais uma vez, convém não esquecer de limpar a cache de DNS, caso execute este procedimento mais do que uma vez ou tenha acedido à página *web*.

- Abra o *browser* numa janela anónima, para evitar que seja acedida a página armazenada na cache do *browser*, em vez da página do servidor *web.*

  Embora esta experiência possa ser feita com qualquer *browser*, a utilização do Chrome mostrou conduzir a capturas mais simples de interpretar, por isso, será aconselhável utilizá-lo.

- Abra a ferramenta *wireshark*, selecione a interface de rede (Ethernet ou WiFi), inicie uma captura e acompanhe na janela do *wireshark*, em tempo real, o efeito das ações seguintes.
- Aceda ao endereço “http://info.cern.ch” na janela anónima aberta no *browser*.
- Aguarde cerca de 10 segundos para que a ligação de dados seja encerrada e pare a captura que tinha iniciado (este tempo de espera no final depende da configuração, o encerramento da ligação pode ser imediato, após o envio da resposta, ou após um “timeout” de alguns segundos).
- Efetue um teste de conetividade, executando na janela de terminal o comando `ping info.cern.ch`
- Identifique o endereço IP do servidor na captura e no teste de conetividade - deverá ser um endereço IPV6, mas em certas configurações de hardware e software mais antigas poderá ser um endereço IPv4.
- Aplique no campo do *wireshark* do filtro de visualização (linha acima da listagem de pacotes) a expressão seguinte, em que deverá substituir <ipv6 address> pelo endereço que identificou no passo anterior (não esquecer de fazer “enter” após colocar a expressão do filtro nessa linha):

  dns.qry.name == "info.cern.ch" or ipv6.addr==<ipv6 address>

  Caso o acesso tenha utilizado endereços IPv4, substitua ipv6.addr==<ipv6 address> por ip.addr==<ipv4 address>.

- Guarde a captura que fez num ficheiro para a analisar em próximas oportunidades (mantenha a extensão .pcapng definida pelo Wireshark, por omissão).
- Analise o registo, quer através da listagem de pacotes exibida na vista geral, quer acedendo aos detalhes de cada pacote nas janelas inferiores ou fazendo duplo clique na linha correspondente da vista geral.

  Tenha em conta que, quando se acede a uma página Web através de um determinado endereço, todos os *browsers* enviam um pedido adicional de acesso a um ficheiro com o nome favicon.ico, localizado na raiz do servidor. Este ficheiro corresponde a uma pequena imagem (ícone personalizado) que é exibido no canto superior esquerdo da página do *browser* quando mostra essa página. Neste caso, poderá verificar que, se aceder ao endereço http://info.cern.ch/favicon.ico, aparece a imagem minúscula do ícone deste servidor na janela principal do *browser*.

Responda seguidamente às seguintes questões escrevendo a resposta no seu logbook:

a. Quantas ligações TCP foram estabelecidas e que par de portas foram utilizadas em cada uma?

   Comece por reconhecer que cada ligação TCP entre duas máquinas está associada a um determinado par de portas, uma utilizada no cliente e outra no servidor. A porta do servidor reservada para o protocolo HTTP é a 80 (cf. [IRC, Secção 6.1]), pelo que as diferentes ligações TCP que possam ser estabelecidas entre o cliente e o servidor corresponderão a portas diferentes estabelecidas dinamicamente do lado do cliente.

   Para responder a esta alínea e às duas seguintes, o *wireshark* dispõe de uma funcionalidade muito útil que é a atribuição de cores diferentes escolhidas pelo utilizador, a cada uma das ligações TCP. Para isso, poderá proceder da seguinte forma, usando nesta explicação as designações das opções da versão em inglês do *wireshark*:

   - Selecione um qualquer pacote (um clique) de uma ligação TCP, correspondente a um dado par de portas (neste caso, a porta 80 do lado do servidor e uma porta específica do lado do cliente).
   - Aceda ao menu View → Colorize Conversation e selecione uma das cores disponíveis; verá que todos os pacotes dessa ligação ficam com essa cor atribuída.
   - Proceda da mesma forma, atribuindo cores diferentes para as outras ligações TCP; será agora muito mais fácil analisar e perceber a utilização de cada ligação.

b. Como foi utilizada (ou não) cada uma das ligações estabelecidas.

c. Identifique os pacotes que iniciam as fases de estabelecimento e de terminação da ligação que suportou a transferência do conteúdo da página *web* (identifique-os no conteúdo da coluna Info através dos padrões [SYN] e [FIN], respetivamente). [IRC, Secção 6.3].

d. Qual a função dos pacotes ACK (identifique-os no conteúdo da coluna Info através do padrão [ACK]) [IRC, Secção 6.3]?

e. Aplique agora o filtro "*http*" aos pacotes e identifique o conteúdo do cabeçalho e do campo de dados (corpo da mensagem) da camada correspondente ao protocolo HTTP, em cada uma das mensagens visualizadas.

f. Considere agora o segundo pacote identificado na alínea anterior, corresponde à resposta do servidor ao pedido inicial. Para este pacote, identifique os endereços de origem e destino da interface de rede (cartas Ethernet), os endereços IP de origem e destino e os endereços das portas TCP envolvidas.

g. No referido pacote, calcule o número de bytes correspondentes às várias camadas da pilha protocolar correspondentes aos protocolos Ethernet (trama / *frame*), IP (datagrama / *datagram*), TCP (segmento / *segment*) e HTTP (mensagem / *message*). Indique para cada camada, o tamanho do cabeçalho / *header*, o tamanho do campo de dados (carga / payload) e o tamanho total, assegurando que os valores indicados estão em conformidade com o princípio do encapsulamento [IRC, Secção 3.1], ou seja, o comprimento do campo de dados em cada camada corresponde ao comprimento total do pacote da camada superior.

   Para visualizar melhor o princípio do encapsulamento, preencha a seguinte tabela no seu logbook, relativa ao tamanho, em bytes, do referido pacote nas diversas camadas da pilha protocolar:

Protocolo	Designação do pacote encapsulado	Comprimento (bytes)		
		Cabeçalho	Dados	Total
Ethernet	Trama			
IP	Datagrama			
TCP	Segmento			
HTTP	Mensagem			

Tenha em conta as seguintes observações relativas aos dados que o wireshark fornece, se abrirem o pacote selecionado:

- Na camada Ethernet, se selecionarem o bloco “Ethernet”, poderão contabilizar o número de bytes do cabeçalho que ficam em destaque na listagem hexadecimal. O bloco “Frame” (trama) indica o comprimento total do pacote.
- No protocolo IP, tal como na camada inferior, se selecionarem o bloco “Internet Protocol” poderão contabilizar o número de bytes do cabeçalho que ficam em destaque na listagem hexadecimal. Se abrirem esse bloco poderão encontrar o comprimento total do datagrama (caso de IPv4) ou o comprimento dos dados/payload (caso de IPv6).
- No protocolo TCP, aplica-se o mesmo procedimento, se selecionarem o bloco “Transmission Control Protocol” poderão contabilizar o número de bytes do cabeçalho do segmento que ficam em destaque na listagem hexadecimal. Se abrirem o bloco, encontram o comprimento dos dados na informação “TCP payload”, que consta igualmente no campo *Len* deste protocolo.
- No protocolo HTTP, a situação é idêntica às camadas inferiores, se selecionarem o bloco “Hypertext Transfer Protocol” poderão contabilizar o número de bytes do cabeçalho da mensagem que ficam em destaque na listagem hexadecimal. Notem que o cabeçalho inclui a linha inicial de pedido ou resultado, os campos de cabeçalho, se existirem, e a linha em branco de terminação (cf. [IRC, Secção 5.1]). Se abrirem este bloco, o tamanho do campo de dados (corpo da mensagem) é explicitado na informação de “File Data” (se esta informação for omissa, significa que o segmento não tem dados).
- Poderão validar os valores obtidos verificando a aplicação do princípio do encapsulamento. Algum valor em falta na tabela que não possa ser calculado com os dados da respetiva camada (Cabeçalho + Dados = Total) também poderá ser obtido pela aplicação do mesmo princípio.