Nome:____ _____ Nº:_____
Nome:_____ Nº:_____

GUIA DE RESOLUÇÃO DA **Ficha Prática Nº 1** Conceitos de Sistemas Informáticos: 2005/06

Ligue a sua máquina e introduza o login **diguest** e a senha **diguest**. Seleccione a opção **Linux fedora 4 <net>** e espere que o sistema operativo seja carregado. Entre no login **diguest** com a senha **diguest** e depois faça **startx** para activar o ambiente gráfico X Window.

A seguir são apresentadas uma série de questões. Uma vez respondidas, o aluno deverá devolver a ficha ao docente da disciplina a fim de ser avaliada.

1. Com auxilio do **man**, investigue os comandos **hostname** e /sbin/**ifconfig**. Diga qual é o nome completo da sua máquina (opção –f) e os respectivos endereços de hardware, de Internet, de broadcast, e a máscara de rede associada.

Nome da máquina obtém-se fazendo hostname –f ➔ brom161.sa.di.uminho.pt

Executando /sbin/ifconfig obtém-se a seguinte informação:
eth0 Link encap:Ethernet HWaddr 00:11:D8:34:0E:BB inet addr:192.168.100.161 Bcast:192.168.100.255 Mask:255.255.255.0

Observa-se assim que:

endereços de hardware 00:11:D8:34:0E:BB, de Internet 192.168.100.161, de broadcast 192.168.100.255,
máscara de rede 255.255.255.0

Uma rede Internet permite a troca de informação entre os computadores que a constituem. Para tal, a informação é transportada em vários pacotes contendo os dados da aplicação e os cabeçalhos adicionados pelos diversos protocolos das camadas da pilha TCP/IP. O **ethereal** é uma aplicação de captura de tráfego que permite analisar o conteúdo dos pacotes que circulam numa rede.

Faça **su –** e depois introduza diguest como password de root. Invoque o **ethereal** e capture alguns pacotes. Observe os pacotes capturados e escolha um que contenha dados de aplicação.

2. A figura que se segue representa um pacote Ethernet completo, com os vários cabeçalhos protocolares encapsulados. Complete-a indicando os protocolos (prot) que observa encapsulados no seu pacote e o tamanho (tam) respectivo, em bytes, de cada um dos cabeçalhos protocolares.

(preâmbulo)	**Lig. Lógica**	**Rede**	**Transporte**	**Aplicação**	(CRC)
8 bytes	Prot ethernet Tam 14	Prot IP Tam 20	Prot TCP Tam 20+12 (opções)	Prot NFS	4 bytes

3. Presentes no seu pacote deverão estar 4 endereços distintos: dois endereços físicos presentes no cabeçalho Ethernet e dois endereços de rede presentes no cabeçalho IP. Identifique-os e transcreva-os, e determine o tamanho (em bytes) de um endereço ethernet e de um endereço IPv4.
Diga ainda, justificando, se o pacote analisado foi recebido ou emitido pela sua máquina.

Endereços físicos observados no cabeçalho Ethernet:
Origem: 00:11:d8:34:0e:bb
Destino: 00:d0:b7:80:de:fc

Endereços de rede observados no cabeçalho IP:
Origem: 192.168.100.161
Destino: 192.168.100.250

Este pacote foi enviado da máquina local (brom161) com destino à máquina 192.168.100.250, como prova os endereços origem eth e IP observados.

4. Ao contrário de um endereço físico, um endereço IP é constituído por dois campos: um que define a (sub)rede a que diz respeito e outro que identifica a máquina nessa mesma rede. A máscara de rede permite identificar os bits respeitantes à (sub)rede.
Identifique o endereço IP da sua máquina na representação do pacote em hexadecimal que observa na janela inferior, e converta-o para binário. Aplique-lhe a máscara de rede fazendo um AND lógico bit-a-bit, e, auxiliando-se das tabelas de conversão, apresente em decimal as componentes de rede e de host obtidas.

Bin	Dec	Hex
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	4	4

Bin	Dec	Hex
0101	5	5
0110	6	6
0111	7	7
1000	8	8
1001	9	9

1010	10	a
1011	11	b
1100	12	c
1101	13	d
1110	14	e
1111	15	f

Fazendo o "AND lógico" bit-a-bit do endereço IP com a máscara (ambos em formato binário) obtém-se a componente de rede. (i.e., a componente de rede é constituídas pelo conjunto de bits cujas posições na máscara estejam a "um")

192.168.100.161 (dec) ➔ C0.A8.64.A1 (hex) ➔ 1100 0000. 1010 1000. 0110 0100. 1010 0001 (bin)
255.255.255.0 (máscara) ➔ & 1111 1111. 1111 1111. 1111 1111. 0000 0000
1100 0000. 1010 1000. 0110 0100. 0000 0000 ➔ 192.168.100.0

A componente de host é constituída pelo conjunto de bits cuja posição na máscara estejam a "zero".
Portanto no endereço IP 192.168.100.161 a componente de rede é 192.168.100 e a de host é 161.

5. As máquinas pertencentes à mesma subrede têm o mesmo prefixo de rede IP. Máquinas pertencentes a redes IP diferentes deverão comunicar através de equipamentos especiais chamados routers. Assim sendo, diga, justificando, se o pacote em análise passou por algum router.

O pacote contém o endereço IP origem=192.168.100.161 e destino=192.168.100.250.

Ambos pertencem à rede 192.168.100.0 e portanto não passaram por nenhum router.

6. O endereço IP de broadcast é reservado para enviar mensagens a todas as máquinas existentes na rede IP a que diz respeito. Observando o endereço de broadcast da sua máquina, deduza como se constrói um a partir do endereço de rede.

O endereço de broadcast é 192.168.100.255 e forma-se colocando todos os bits do campo do host a "um".

7. Investigue o comando **ping**. Como usaria este comando para saber rapidamente o endereço IP de todas as máquinas activas existentes na sua rede IP?

Executa-se um ping para o endereço de broadcast e observa-se no monitor os endereços IP de todas as máquinas activas na rede.

ping –b 192.168.100.255

8. O número de porta, existente nos cabeçalhos de transporte, permite identificar o protocolo de aplicação envolvido numa comunicação. Identifique as portas origem e destino presentes no pacote em análise. Com ajuda do **vi /etc/services**, determine a porta atribuída aos serviços http e smtp (email).

No cabeçalho TCP identificam-se facilmente as portas origem e destino:
Source port: 800
Destination port: 2049

O servidor http escuta na porta 80 e o smtp na porta 25.

9. Pretende-se aceder por http à máquina servidora **marco** existente no domínio **uminho.pt** para obter a página **index.html** existente na directoria <directoria-base das pág. html>/disciplinas/CSI.
Indique a URL respeitante a este recurso, e confirme-a com o seu *browser*.

http://marco.uminho.pt/disciplinas/CSI/index.html

Os filtros do ethereal permitem capturar (*capture filter*) ou mostrar (*display filter*) apenas pacotes com características específicas. Apesar de ambos filtros produzirem o mesmo *output* final, existem diferenças essenciais nos fins a que se destinam. A captura de tráfego com visualização em tempo real é o exemplo típico da utilização do *capture filter*.
Na sua máquina arranque um *browser* (e.g. netscape, mozilla) e configure-o para utilizar o *proxy* HTTP **proxy.di.uminho.pt** (193.136.19.6, 193.136.19.8) na porta **3128** (Edit > Preferences > Proxies > …).

10. Coloque o ethereal a capturar tráfego. Com o browser da sua máquina aceda à página indicada na alínea anterior, e depois pare a captura.

a) Porque razão os pacotes que trazem a página pretendida não provêm nem da máquina marco.uminho.pt, (193.136.9.240) nem da porta 80 (atribuída ao serviço http)? Qual a máquina e a porta de que é proveniente a página? A que máquina o seu browser efectuou o pedido http?

Provêm da máquina 193.136.19.8, porta 3128, que são precisamente o endereço IP da máquina proxy proxy.di.uminho.pt e a porta com os quais se configurou o browser. Foi precisamente a esta máquina (e também a esta porta) que o browser efectuou o pedido.

b) Transcreva a regra que permite ao ***display filter*** (Analyze>Display Filter) mostrar unicamente pacotes cujo endereço IP destino seja o da sua máquina, e cujo nº de porta origem seja o da proxy. Experimente-a com os pacotes capturados na alínea anterior.

ip.dst==192.168.100.161&&tcp.srcport==3128

11. Coloque novamente o ethereal a capturar tráfego, mas agora com o ***capture filter*** especificado para a regra da alínea anterior. Com o *browser* da sua máquina volte a aceder à mesma página (com Shift+Reload), e depois pare a captura.

a) Transcreva a especificação usada no filtro, e diga, observando a captura, quantos ficheiros foram transferidos para a sua máquina ao aceder à referida página?

dst host 192.168.100.161 && src port 3128 (Nota: A sintaxe do capture filter é idêntica à do tcpdump)

Para saber a quantidade de ficheiros transferidos deve-se contar o número de ficheiros trazidos com sucesso em resposta aos vários pedidos GET efectuados pelo *browser* (e.g. GET http://marco.uminho.pt/disciplinas/CSI/index.html HTTP/1.1\r\n)

Identificam-se assim 3 ficheiros transferidos: um ficheiro html e duas imagens.

b) Quantos pacotes foram precisos para transferir o código html da página, e qual o tamanho total deste? (sugestão: procurar esta informação no campo Content-Length presente no cabeçalho da resposta HTTP)

Foram precisos 11 pacotes.

Observa-se no cabeçalho http do primeiro pacote da resposta que:
Content-Length: 11544 (bytes)

c) Qual a percentagem de bytes imposto pelo protocolo de comunicação TCP/IP relativamente à quantidade de bytes do código html transferido, ou seja, qual é o *overhead* imposto pelo TCP/IP nessa transferência?
Nota: adicione 12 bytes a cada cabeçalho ethernet, por forma a contabilizar o preâmbulo e o campo de verificação de erros CRC.

O numero total de bytes de cabeçalho transportado pela rede é (ignorando o cabeçalho http)
11 pacotes*[(14+12)(eth)+20(IP)+32(TCP)]=858 bytes

Portanto o *overhead* imposto pelo TCP/IP na transferência da referida página html é 858/11544=7,4%
(na realidade o *overhead* real é um pouco mais alto porque não se contabilizaram os pacotes de estabelecimento e encerramento da ligação TCP)