Ficha do Trabalho Prático nº 5

Protocolos IP e ICMP

Descrição do trabalho: Estudo dos protocolos do nível de rede IP e ICMP. Análise do formato dos datagramas IP e das mensagens ICMP.

Entre num dos sistemas Linux com o seu username (cdr-g0<nº PC>) e password 3690147258

1. Protocolos IP

Em anexo, nas tramas 1 e 2, é apresentada a descodificação de dois datagramas IP capturados na rede local do laboratório, utilizando um analisador de rede. Analise um desses datagramas IP.

a) Refira-se aos valores observados nos campos: *Version, Header_Length, Total_Length, Time to Live, Protocol, Header Checksum, Source e Destination Address*. Qual a função e utilidade de cada um destes campos?

Version: contém a identificação da versão do protocolo IP usado no datagrama.
Header_Length: tamanho do cabeçalho, medido em palavras de 32 bits.
Total_Length: tamanho do ~~the~~ datagrama IP, medido em octetos.
Time to Live: especifica o tempo, em segundos, que um datagrama é permitido permanecer na rede, evitando que fique infinitamente na rede.
Protocol: identifica o protocolo do nível superior transportado na área Data do datagrama.
Header Checksum: a integridade dos valores do cabeçalho, pode ser formada por uma sequência de 16 bits.
Source e Destination Address: contêm os endereços IP de 32 bits do emissor e recetor do datagrama. Estes nunca mudam especificam os endereços do emissor original e do último recetor.

b) Por que é que as entidades do nível de rede às vezes fragmentam os datagramas IP? Indique vantagens ou desvantagens resultantes das operações de fragmentação.

como o emissor de um datagrama desconhece o tamanho mínimo dos MTU de cada fracção de rede, estes não for vezes enviados com um tamanho superior ao MTU de rede que liga ~~dois~~ por exemplo dois routers, por isso pode que o datagrama tenha de ser fragmentado para que possa passar nessa ~~rede~~ parte de rede. Desvantagens: o datagrama ao ser fragmentado pelo router, diminui o seu desempenho, se um fragmento possuir um tamanho inferior ao MTU de uma parte de rede, há um desaproveitamento desse mesmo. Vantagens: o emissor não tem de se preocupar com o tamanho mínimo dos datagramas que envia, ~~sendo antes~~ podendo estes serem fragmentados, ou não, por qualquer parte da rede, independente do MTU: os fragmentos só são recapturados no último destino. Outra desvantagem é o facto de que se um dos fragmentos é perdido, todo o ~~tag~~ datagrama é também perdido, aumentando a probabilidade de um datagrama se perder.

c) Poderá algum dos dois datagramas IP encapsulados nas tramas 1 e 2 ser considerado um fragmento de um outro datagrama IP que tenha sido fragmentado? Justifique.

O primeiro datagrama não é um fragmento, pois o flag, referente ao bit indicador de mais fragmentos, está a 0, ou seja, é o último fragmento, e o fragment offset está a 0, o que indica que este é o primeiro e o último fragmento do datagrama original, ou seja, este é o datagrama original.
O segundo datagrama é um fragmento, pois o flag, referente ao bit indicador de mais fragmentos está a 1, logo indica que existem mais fragmentos a seguir a este e referentes do mesmo datagrama original, logo o datagrama é um fragmento do outro datagrama.

d) Quais os campos do cabeçalho IP que têm de ser consultados para se efectuar o reagrupamento dos diversos fragmentos de um mesmo datagrama IP original? Explique as funções desses campos.

São necessários os campos Identification, Flags e Fragment Offset. O campo Identification fornece um número único como o todos os fragmentos de um datagrama, que permite identificar quais os fragmentos que pertencem a um datagrama. O campo Flags diz-nos o bit indicador de mais fragmentos, indica-nos um fragmento é o último, ou seja se existem mais fragmentos a seguir a este. O campo Fragment Offset indica a ordem pela qual os fragmentos estão ordenados.

2. Protocolo ICMP

As tramas 3 e 4, em anexo, encapsulam dois datagramas capturados imediatamente após se ter executado o comando **ping kepler** a partir da estação pc2. Nota: Sempre que necessário recorra à bibliografia de apoio a este trabalho.

a.1) O comando **ping** envolve a troca de duas mensagens ICMP distintas. Quais são elas?
Indique qual o formato das mensagens ICMP contidas nas tramas 3 e 4 bem como a função de cada campo dessas mensagens.

As mensagens são: echo request, na trama 3, e echo reply na trama 4. Cada mensagem ICMP contém um Type e um code que juntos identificam o tipo de mensagem. Um campo Checksum que verifica a integridade de toda a mensagem ICMP, um identifier que diferencia uma mensagem de outra, um campo sequence number que no caso do Ping envia com um contador por cada echo request, um campo de data para outras informações adicionais.

a.2) Em que tipo de pacotes (PDU's) são encapsuladas as mensagens ICMP? Justifique a resposta a partir do que verifica nas capturas apresentadas.

As mensagens são encapsuladas em ~~[illegible]~~ datagramas IP, porque os cabeçalhos que se encontram acima do cabeçalho ICMP são do tipo IP. Consequentemente uma mensagem ICMP é sempre encapsulada em datagrama IP

a.3) Um dos valores devolvidos pelo comando ping corresponde ao denominado *round trip time*, ou seja, o tempo que decorre entre o envio duma mensagem ICMP a uma máquina remota e a recepção da resposta desta última. Analisando o resultado de um ping verifica-se que para o primeiro pacote gerado, esse tempo é por vezes ligeiramente superior ao dos seguintes. Qual a justificação para este comportamento?

Isso acontece quando o endereço de hardware do destino não se encontra na cache ARP do emissor.

c) Explore o comando traceroute e descreva o processo utilizado por este programa para determinar a rota que os datagramas seguem até um dado destino.

Em primeiro lugar o traceroute envia um datagrama com o TTL=1. Quando este datagrama chega ao primeiro router, o datagrama é destruído e uma mensagem de erro ICMP, indicando que o tempo de trama acabou, e que o mesmo foi destruído, é enviada para o emissor do datagrama. Sabendo que como esta mensagem de erro contém informação sobre o router onde o datagrama chegou, o traceroute consegue assim saber e identificar o primeiro router por onde a mensagem passa, ou chega.
Quando o TTL = N, o traceroute identifica o nome do N router por onde ou até onde o datagrama chega, e assim após enviar N ~~[illegible]~~ datagramas, sendo o TTL de cada um [illegible] de 1 até N, e sem repetição, o traceroute identifica todos os routers, ou seja, o percurso que o datagrama percorre até ao seu destino final.

```
- - - - - - - - - - - - - Frame 1 - - - - - - - - - - - - -
Network Analyzer data from 06-Apr-2004 at 10:43:30

-------- IP HEADER --------

Version = 4, Header length = 20 bytes
Diff Serv Field = 0x00 (DSCP 0x00: Default; ECN: 0x00)
 0 0 0 0  0 0 . .  = DSCP: Default (0x00)
 . . . . . . 0 .  = ECN-Capable Transport (ECT): 0
 . . . . . . . 0  = ECN-CE: 0
Total length = 68 bytes
Identification = 19641
Flags = 0x0
 . 0 . . . . . .  = may fragment
 . . 0 . . . . .  = last fragment
Fragment offset = 0 bytes
Time to live = 30
Protocol = 0x11 (UDP)
Header checksum = 0xD8EC (correct)
Source address = [192.168.90.15]
Destination address = [192.168.89.11]
No options




- - - - - - - - - - - - - Frame 2 - - - - - - - - - - - - -
Network Analyzer data from 06-Apr-2004 at 10:50:12

-------- IP HEADER --------

Version = 4, Header length = 20 bytes
Diff Serv Field = 0x00 (DSCP 0x00: Default; ECN: 0x00)
 0 0 0 0  0 0 . .  = DSCP: Default (0x00)
 . . . . . . 0 .  = ECN-Capable Transport (ECT): 0
 . . . . . . . 0  = ECN-CE: 0
Total length = 1500 bytes
Identification = 57157
Flags = 0x2
 . 0 . . . . . .  = may fragment
 . . 1 . . . . .  = more fragments
Fragment offset = 1480 bytes
Time to live = 255
Protocol = 0x11 (UDP)
Header checksum = 0x3F0E (correct)
Source address = [192.168.90.12]
Destination address = [192.168.89.14]
No options
```

```
- - - - - - - - - - - - Frame 3 - - - - - - - - - - - -
Network Analyzer data from 06-Apr-2004 at 11:08:47

-------- MAC HEADER --------
Frame size is 98 bytes

Destination = station 00:00:21:80:C4:62, (kepler)
Source = station 00:50:FC:5C:E9:B0, (pc2)
Ethertype = 0x0800 (IP)

-------- IP HEADER --------
Version = 4, Header length = 20 bytes
Diff Serv Field = 0xC0 (DSCP 0x30: Class Selector 6; ECN: 0x00)
 1 1 0 0  0 0 . .  = DSCP: Class Selector 6 (0x30)
 . . . .  . . 0 .  = ECN-Capable Transport (ECT): 0
 . . . .  . . . 0  = ECN-CE: 0
Total length = 84 bytes
Identification = 19256
Flags = 0x0
 . 0 . .  . . . .  = may fragment
 . . 0 .  . . . .  = last fragment
Fragment offset = 0 bytes
Time to live = 64
Protocol = 0x01 (ICMP)
Header checksum = 0xB86D (correct)
Source address = [192.168.89.12]  (pc2.labcom.uminho.pt)
Destination address = [192.168.89.89]  (kepler.labcom.uminho.pt)
No options

-------- ICMP HEADER --------
Type = 8 (Echo)
Code = 0
Checksum = 0x7D9E
Identifier = 57887
Sequence number = 0
[56 bytes of data]

[Normal end of "ICMP header"]
```

```
- - - - - - - - - - - - - Frame 4 - - - - - - - - - - - - -
Network Analyzer data from 06-Apr-2004 at 11:18:07

-------- MAC HEADER --------
Frame size is 98 bytes

Destination = station 00:50:FC:5C:E9:B0, (pc2)
Source = station 00:00:21:80:C4:62, (kepler)
Ethertype = 0x0800 (IP)

-------- IP HEADER --------
Version = 4, Header length = 20 bytes
Diff Serv Field = 0xC0 (DSCP 0x30: Class Selector 6; ECN: 0x00)
 1 1 0 0  0 0 . .  = DSCP: Class Selector 6 (0x30)
 . . . .  . . 0 .  = ECN-Capable Transport (ECT): 0
 . . . .  . . . 0  = ECN-CE: 0
Total length = 84 bytes
Identification = 38758
Flags = 0x0
 . 0 . .  . . . .  = may fragment
 . . 0 .  . . . .  = last fragment
Fragment offset = 0 bytes
Time to live = 255
Protocol = 0x01 (ICMP)
Header checksum = 0xAD3E (correct)
Source address = [192.168.89.89]  (kepler.labcom.uminho.pt)
Destination address = [192.168.89.12]  (pc2.labcom.uminho.pt)
No options

-------- ICMP HEADER --------
Type = 0 (Echo reply)
Code = 0
Checksum = 0x859E
Identifier = 57887
Sequence number = 0
[56 bytes of data]

[Normal end of "ICMP header"]
```