

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Instituto de Informática

INF01154 - Redes de Computadores N

Prof. Valter Roesler

Laboratório 05

Guilherme Schievelbein

João Luiz Grave Gross

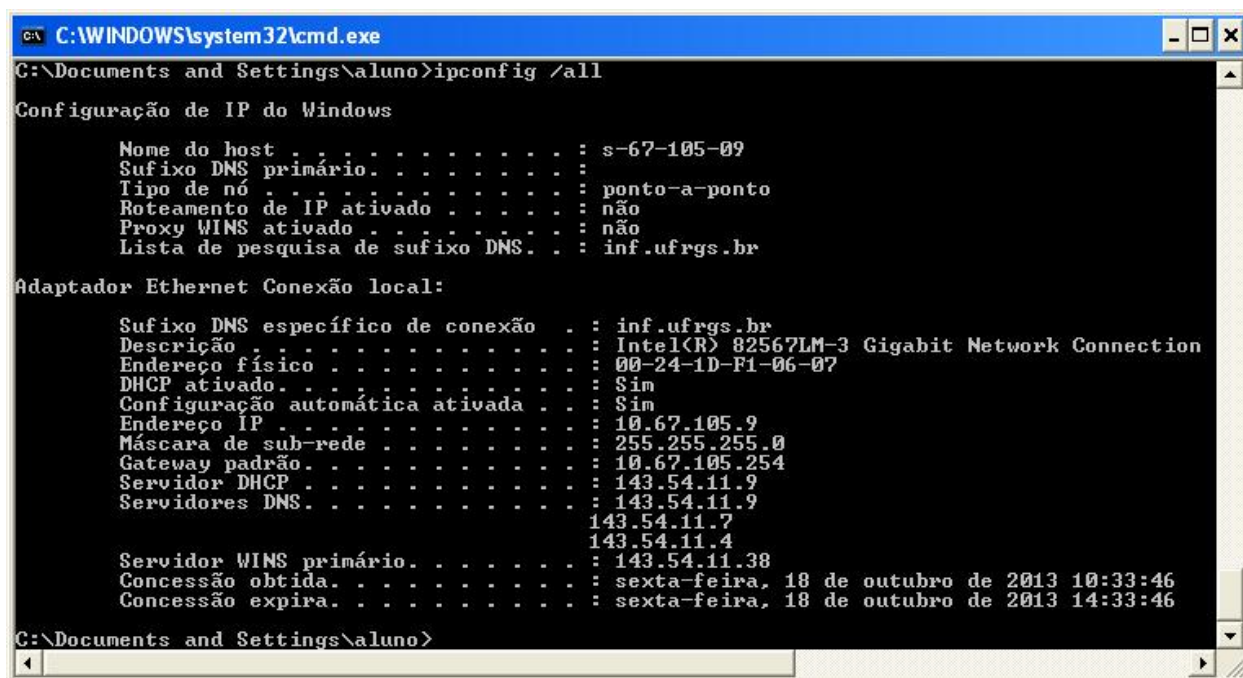
Porto Alegre, novembro de 2013.

Atividades

1. Leia o help do ping (ping /?) e efetue os seguintes testes:

a. Fazer um ping com 8 requisições de echo, tamanho do pacote de 200 bytes, TTL de 80. Mostre o comando utilizado e prove que funcionou através de uma imagem.

Antes de iniciarmos o envio dos pacotes rodamos a instrução ipconfig /all no cmd do Windows para identificar o IP e o MAC do nosso terminal. Também obtivemos as informações do terminal de destino.



```
C:\WINDOWS\system32\cmd.exe
C:\Documents and Settings\aluno>ipconfig /all

Configuração de IP do Windows

Nome do host . . . . . : s-67-105-09
Sufixo DNS primário. . . . . :
Tipo de nó . . . . . : ponto-a-ponto
Roteamento de IP ativado . . . . . : não
Proxy WINS ativado . . . . . : não
Lista de pesquisa de sufixo DNS. . . : inf.ufrgs.br

Adaptador Ethernet Conexão local:

Sufixo DNS específico de conexão . : inf.ufrgs.br
Descrição . . . . . : Intel(R) 82567LM-3 Gigabit Network Connection
Endereço físico . . . . . : 00-24-1D-F1-06-07
DHCP ativado. . . . . : Sim
Configuração automática ativada . : Sim
Endereço IP . . . . . : 10.67.105.9
Máscara de sub-rede . . . . . : 255.255.255.0
Gateway padrão. . . . . : 10.67.105.254
Servidor DHCP . . . . . : 143.54.11.9
Servidores DNS. . . . . : 143.54.11.9
                        143.54.11.7
                        143.54.11.4
Servidor WINS primário. . . . . : 143.54.11.38
Concessão obtida. . . . . : sexta-feira, 18 de outubro de 2013 10:33:46
Concessão expira. . . . . : sexta-feira, 18 de outubro de 2013 14:33:46

C:\Documents and Settings\aluno>
```

Figura 1: informações do terminal de trabalho.

```
C:\WINDOWS\system32\cmd.exe
G:\Documents and Settings\aluno>ipconfig/all

Configuração de IP do Windows

    Nome do host . . . . . : s-67-105-10
    Sufixo DNS primário. . . . . :
    Tipo de nó . . . . . : ponto-a-ponto
    Roteamento de IP ativado . . . . . : não
    Proxy WINS ativado . . . . . : não
    Lista de pesquisa de sufixo DNS. . . : inf.ufrgs.br

Adaptador Ethernet Conexão local:

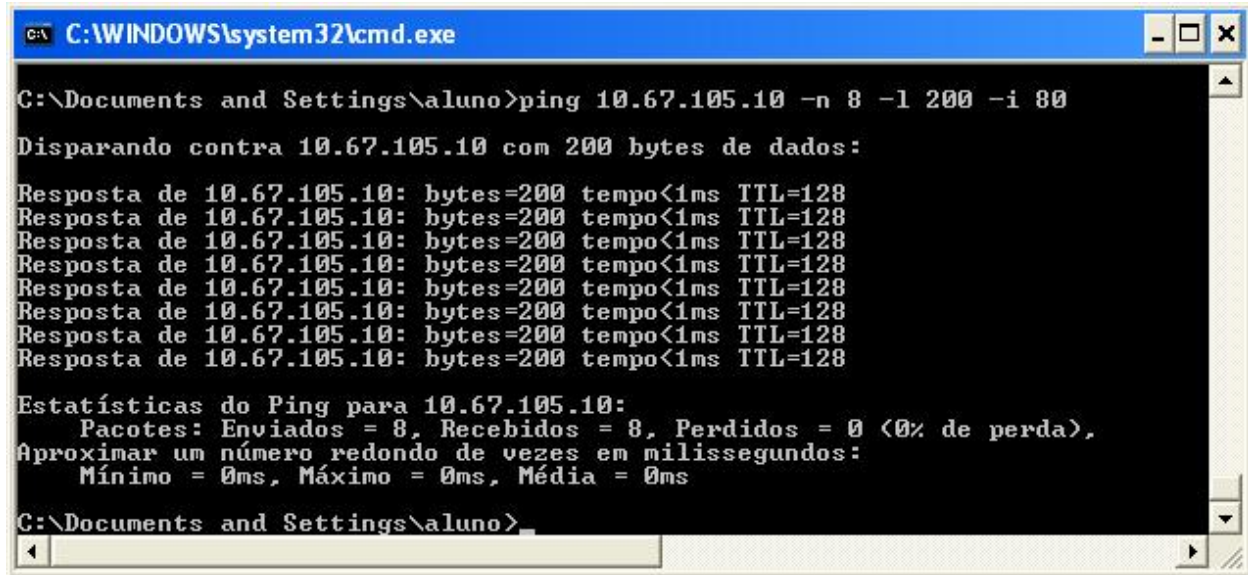
    Sufixo DNS específico de conexão . : inf.ufrgs.br
    Descrição . . . . . : Intel(R) 82567LM-3 Gigabit Network
Connection
    Endereço físico . . . . . : 00-24-1D-F1-05-1F
    DHCP ativado. . . . . : Sim
    Configuração automática ativada . . : Sim
    Endereço IP . . . . . : 10.67.105.10
    Máscara de sub-rede . . . . . : 255.255.255.0
    Gateway padrão. . . . . : 10.67.105.254
    Servidor DHCP . . . . . : 143.54.11.9
    Servidores DNS. . . . . : 143.54.11.9
                             143.54.11.7
                             143.54.11.4
    Servidor WINS primário. . . . . : 143.54.11.38
    Concessão obtida. . . . . : sexta-feira, 18 de outubro de 2013
10:34:06
    Concessão expira. . . . . : sexta-feira, 18 de outubro de 2013
14:34:06

G:\Documents and Settings\aluno>
```

Figura 2: informações da máquina de destino.

Podemos ver que o nosso IP é 10.67.105.9 e o MAC é 00-24-1D-F1-06-07. Já a máquina de destino possui IP 10.67.105.10 e o MAC 00-24-1D-F1-05-1F.

Após isso realizamos o ping de 8 mensagens com tamanho de pacote de 200 bytes e TTL de 80.



```
C:\WINDOWS\system32\cmd.exe

C:\Documents and Settings\aluno>ping 10.67.105.10 -n 8 -l 200 -i 80

Disparando contra 10.67.105.10 com 200 bytes de dados:

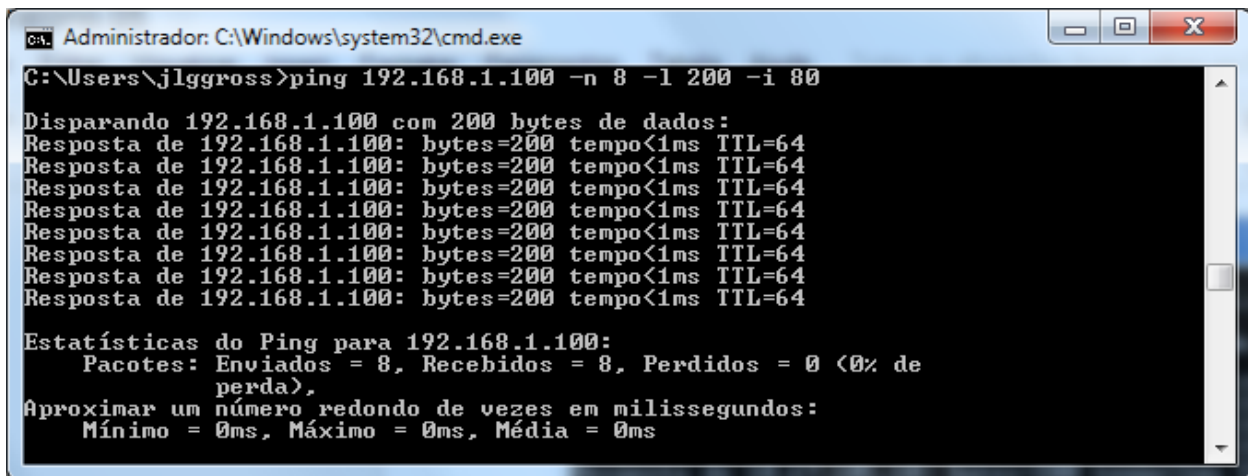
Resposta de 10.67.105.10: bytes=200 tempo<1ms TTL=128
Resposta de 10.67.105.10: bytes=200 tempo<1ms TTL=128
Resposta de 10.67.105.10: bytes=200 tempo<1ms TTL=128
Resposta de 10.67.105.10: bytes=200 tempo<1ms TTL=128
Resposta de 10.67.105.10: bytes=200 tempo<1ms TTL=128
Resposta de 10.67.105.10: bytes=200 tempo<1ms TTL=128
Resposta de 10.67.105.10: bytes=200 tempo<1ms TTL=128
Resposta de 10.67.105.10: bytes=200 tempo<1ms TTL=128

Estatísticas do Ping para 10.67.105.10:
    Pacotes: Enviados = 8, Recebidos = 8, Perdidos = 0 (0% de perda),
Aproximar um número redondo de vezes em milissegundos:
    Mínimo = 0ms, Máximo = 0ms, Média = 0ms

C:\Documents and Settings\aluno>
```

Figura 3: ping para PC de IP 10.67.105.10.

Os pacotes foram enviados com sucesso, porém o TTL (Time To Live) não foi de 80 como havíamos estabelecido, mas de 128. Isso pois o TTL que aparece na resposta é o TTL do destino até a origem, e por padrão para computadores que utilizam Windows esse valor é de 128. Para computadores que rodam Linux a resposta tem TTL de 64.



```
Administrador: C:\Windows\system32\cmd.exe

C:\Users\jlgross>ping 192.168.1.100 -n 8 -l 200 -i 80

Disparando 192.168.1.100 com 200 bytes de dados:

Resposta de 192.168.1.100: bytes=200 tempo<1ms TTL=64
Resposta de 192.168.1.100: bytes=200 tempo<1ms TTL=64
Resposta de 192.168.1.100: bytes=200 tempo<1ms TTL=64
Resposta de 192.168.1.100: bytes=200 tempo<1ms TTL=64
Resposta de 192.168.1.100: bytes=200 tempo<1ms TTL=64
Resposta de 192.168.1.100: bytes=200 tempo<1ms TTL=64
Resposta de 192.168.1.100: bytes=200 tempo<1ms TTL=64
Resposta de 192.168.1.100: bytes=200 tempo<1ms TTL=64

Estatísticas do Ping para 192.168.1.100:
    Pacotes: Enviados = 8, Recebidos = 8, Perdidos = 0 (0% de
perda),
Aproximar um número redondo de vezes em milissegundos:
    Mínimo = 0ms, Máximo = 0ms, Média = 0ms
```

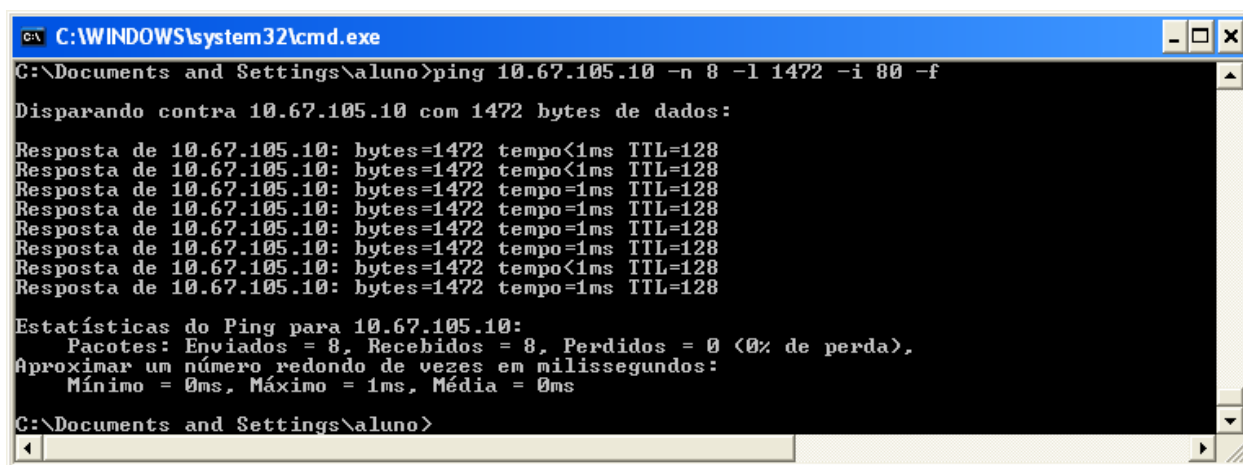
Figura 4: Envio de pacotes para um computador da rede que roda Linux (Ubuntu 13.04).

O TTL é muito importante na rede, pois estabelece o tempo em que o pacote irá trafegar na rede. O “tempo” é contado como a quantidade de roteadores pelos quais o pacote passa. Cada vez que o pacote passa por um roteador o TTL é decrementado de uma unidade, logo, se um pacote sai da sua máquina com TTL 255 e retorna com TTL 5 isso quer dizer que o pacote

passou por 5 roteadores. Esse valor é importante, pois evita que pacotes fiquem circulando na rede por tempo indeterminado, caso ocorram loops. Se o TTL chegar a zero o pacote é descartado.

b. Fazer um ping forçando a não fragmentação do pacote (-f) e tamanho do pacote de 1600 bytes. Verificar qual o máximo tamanho do pacote que funciona. Explique.

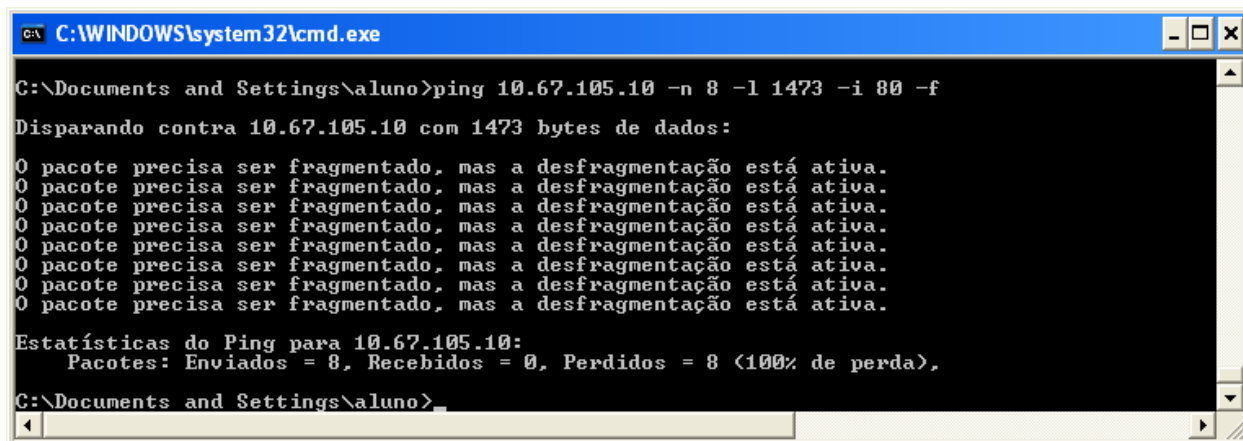
O maior valor de pacote que conseguimos enviar foi de 1472 bytes.



```
C:\WINDOWS\system32\cmd.exe
C:\Documents and Settings\aluno>ping 10.67.105.10 -n 8 -l 1472 -i 80 -f
Disparando contra 10.67.105.10 com 1472 bytes de dados:
Resposta de 10.67.105.10: bytes=1472 tempo<1ms TTL=128
Resposta de 10.67.105.10: bytes=1472 tempo<1ms TTL=128
Resposta de 10.67.105.10: bytes=1472 tempo=1ms TTL=128
Resposta de 10.67.105.10: bytes=1472 tempo=1ms TTL=128
Resposta de 10.67.105.10: bytes=1472 tempo=1ms TTL=128
Resposta de 10.67.105.10: bytes=1472 tempo<1ms TTL=128
Resposta de 10.67.105.10: bytes=1472 tempo=1ms TTL=128
Estatísticas do Ping para 10.67.105.10:
  Pacotes: Enviados = 8, Recebidos = 8, Perdidos = 0 (0% de perda),
  Aproximar um número redondo de vezes em milissegundos:
    Mínimo = 0ms, Máximo = 1ms, Média = 0ms
C:\Documents and Settings\aluno>
```

Figura 5: Tamanho máximo de mensagem, 1472 bytes.

Para valores de pacote maiores do que 1472 bytes, não é possível enviar sem fragmentar.



```
C:\WINDOWS\system32\cmd.exe
C:\Documents and Settings\aluno>ping 10.67.105.10 -n 8 -l 1473 -i 80 -f
Disparando contra 10.67.105.10 com 1473 bytes de dados:
0 pacote precisa ser fragmentado, mas a desfragmentação está ativa.
0 pacote precisa ser fragmentado, mas a desfragmentação está ativa.
0 pacote precisa ser fragmentado, mas a desfragmentação está ativa.
0 pacote precisa ser fragmentado, mas a desfragmentação está ativa.
0 pacote precisa ser fragmentado, mas a desfragmentação está ativa.
0 pacote precisa ser fragmentado, mas a desfragmentação está ativa.
0 pacote precisa ser fragmentado, mas a desfragmentação está ativa.
Estatísticas do Ping para 10.67.105.10:
  Pacotes: Enviados = 8, Recebidos = 0, Perdidos = 8 (100% de perda),
C:\Documents and Settings\aluno>
```

Figura 6: Pacote de 1473 bytes ou mais não podem ser enviados sem fragmentação.

Em uma rede Ethernet o limite de cada pacote de dados é de 1500, mas o limite de 1472 é

imposto devido aos headers dos protocolos IP (20 bytes) e ICMP (8 bytes). Isso foi observado com um programa sniffer, chamado Wireshark. Ao enviar os pacotes conseguimos ver o conteúdo de cada mensagem, bem como a estrutura.

10.67.105.9	10.67.105.10	ICMP	242 Echo (ping) request	id=0x0200, seq=53504/209, ttl=80
10.67.105.10	10.67.105.9	ICMP	242 Echo (ping) reply	id=0x0200, seq=53504/209, ttl=128
10.67.105.9	10.67.105.10	ICMP	242 Echo (ping) request	id=0x0200, seq=53760/210, ttl=80
10.67.105.10	10.67.105.9	ICMP	242 Echo (ping) reply	id=0x0200, seq=53760/210, ttl=128
10.67.105.9	10.67.105.10	ICMP	242 Echo (ping) request	id=0x0200, seq=54016/211, ttl=80
10.67.105.10	10.67.105.9	ICMP	242 Echo (ping) reply	id=0x0200, seq=54016/211, ttl=128

Frame 71: 242 bytes on wire (1936 bits), 242 bytes captured (1936 bits) on interface 0	
Ethernet II, Src: Giga-Byt_f1:06:07 (00:24:1d:f1:06:07), Dst: Giga-Byt_f1:05:1f (00:24:1d:f1:05:1f)	
Destination: Giga-Byt_f1:05:1f (00:24:1d:f1:05:1f)	
Source: Giga-Byt_f1:06:07 (00:24:1d:f1:06:07)	
Type: IP (0x0800)	
Internet Protocol Version 4, Src: 10.67.105.9 (10.67.105.9), Dst: 10.67.105.10 (10.67.105.10)	
Internet Control Message Protocol	
Type: 8 (Echo (ping) request)	
Code: 0	
Checksum: 0xe9bc [correct]	
Identifier (BE): 512 (0x0200)	
Identifier (LE): 2 (0x0002)	
Sequence number (BE): 53760 (0xd200)	
Sequence number (LE): 210 (0x00d2)	

0000	00 24 1d f1 05 1f 00 24 1d f1 06 07 08 00 45 00	..\$.\$E.
0010	00 e4 6a e3 40 00 50 01 d8 9c 0a 43 69 09 0a 43	...j.@.P.Cl..C
0020	69 0a 08 00 e9 bc 02 00 d2 00 61 62 63 64 65 66	i..... ..abcdef
0030	67 68 69 6a 6b 6c 6d 6e 6f 70 71 72 73 74 75 76	ghijklmn opqrstuv
0040	77 61 62 63 64 65 66 67 68 69 6a 6b 6c 6d 6e 6f	wabcdefg hijklmno
0050	70 71 72 73 74 75 76 77 61 62 63 64 65 66 67 68	pqrstuvw abcdefgh
0060	69 6a 6b 6c 6d 6e 6f 70 71 72 73 74 75 76 77 61	ijklmnop qrstuvwa
0070	62 63 64 65 66 67 68 69 6a 6b 6c 6d 6e 6f 70 71	bcdefghi jklmnopq
0080	72 73 74 75 76 77 61 62 63 64 65 66 67 68 69 6a	rstuvwab cdefghij
0090	6b 6c 6d 6e 6f 70 71 72 73 74 75 76 77 61 62 63	klmnopqr stuvwabc
00a0	64 65 66 67 68 69 6a 6b 6c 6d 6e 6f 70 71 72 73	defghijk lmnopqrs
00b0	74 75 76 77 61 62 63 64 65 66 67 68 69 6a 6b 6c	tuvwabcd efghijkl
00c0	6d 6e 6f 70 71 72 73 74 75 76 77 61 62 63 64 65	mnpqrstu vwabcde

Figura 7: Programa sniffer indicando header de 20 bytes do IP na área de dados.

c. Qual a mensagem (pacote de dados) enviado num comando de ping? Sugestão: analise o pacote ICMP.

Com um sniffer conseguimos observar qual a mensagem enviada pelo computador de origem.

No.	Time	Source	Destination	Protocol	Length	Info
74129	230.9013870	192.168.1.103	192.168.1.100	ICMP	74	Echo (ping) request
74130	230.9020330	192.168.1.100	192.168.1.103	ICMP	74	Echo (ping) reply

Frame 74129: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface 0	
Ethernet II, Src: IntelCor_bc:09:fa (00:1c:c0:bc:09:fa), Dst: Dell_d8:07:75 (f0:4d:a2:d8:07:75)	
Destination: Dell_d8:07:75 (f0:4d:a2:d8:07:75)	
Source: IntelCor_bc:09:fa (00:1c:c0:bc:09:fa)	
Type: IP (0x0800)	
Internet Protocol Version 4, Src: 192.168.1.103 (192.168.1.103), Dst: 192.168.1.100 (192.168.1.100)	
Internet Control Message Protocol	

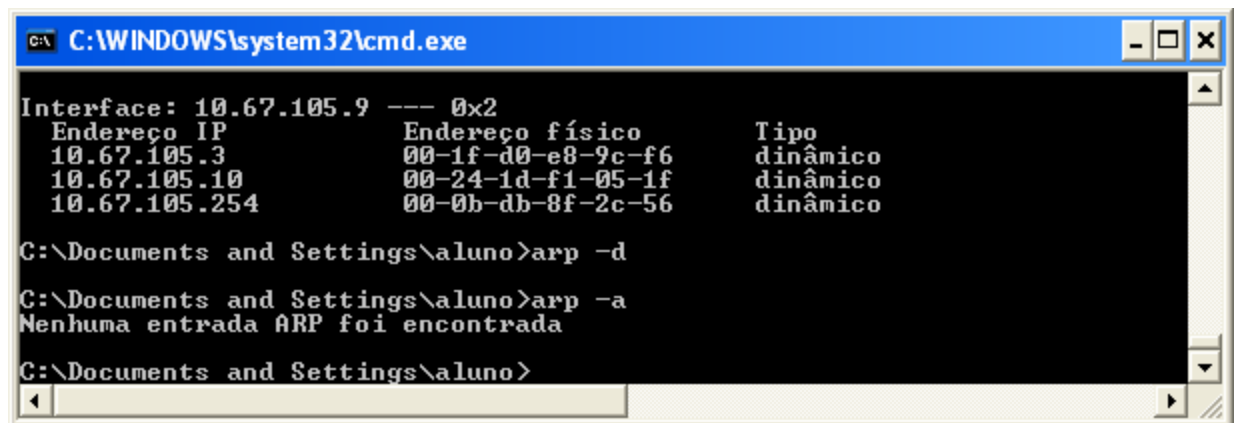
0000	f0 4d a2 d8 07 75 00 1c c0 bc 09 fa 08 00 45 00	.M...u... ..E.
0010	00 3c 06 76 00 00 80 01 b0 2f c0 a8 01 67 c0 a8	..<.V.... ./...g..
0020	01 64 08 00 4d 4a 00 01 00 11 61 62 63 64 65 66	..d..Mj... ..abcdef
0030	67 68 69 6a 6b 6c 6d 6e 6f 70 71 72 73 74 75 76	ghijklmn opqrstuv
0040	77 61 62 63 64 65 66 67 68 69	wabcdefg hi

Figura 8: Pacote na rede Ethernet enviado pela máquina de origem ao destino.

A mensagem na rede Ethernet é formada por 6 bytes do MAC de destino, 6 bytes do MAC de origem, 2 bytes para tipo, área de dados e 4 bytes de CRC-32. A área de dados possui 20 bytes de header IP, 8 bytes de header ICMP e os demais são de dados propriamente ditos. No caso mostrado na figura 8 os dados possuem 32 bytes.

2. Utilizar um sniffer de redes para analisar o funcionamento do ping. Capturar o ping de sua máquina para um vizinho, preencha na tabela linha a linha a sequência de comandos de um ping, explicando cada linha (mostrando o endereço nível 2 e nível 3 envolvido em cada linha). Também deve ser considerado o protocolo ARP, além do ICMP. Pode-se utilizar a simbologia MAC_A (MAC da máquina A), MAC_B (MAC da máquina B) e MAC-R (MAC do roteador), por exemplo. Da mesma forma, pode-se utilizar IP_A, IP_B, etc. Quando não existir pacotes no nível, basta colocar “Não tem” na tabela.

Antes de realizar essa análise, executamos o comando `arp -d` no cmd para excluir a tabela de MACs do nosso computador.



```
C:\WINDOWS\system32\cmd.exe

Interface: 10.67.105.9 --- 0x2
Endereço IP      Endereço físico    Tipo
10.67.105.3      00-1f-d0-e8-9c-f6  dinâmico
10.67.105.10     00-24-1d-f1-05-1f  dinâmico
10.67.105.254    00-0b-db-8f-2c-56  dinâmico

C:\Documents and Settings\aluno>arp -d

C:\Documents and Settings\aluno>arp -a
Nenhuma entrada ARP foi encontrada

C:\Documents and Settings\aluno>
```

Figura 9: Limpando tabela de MACs

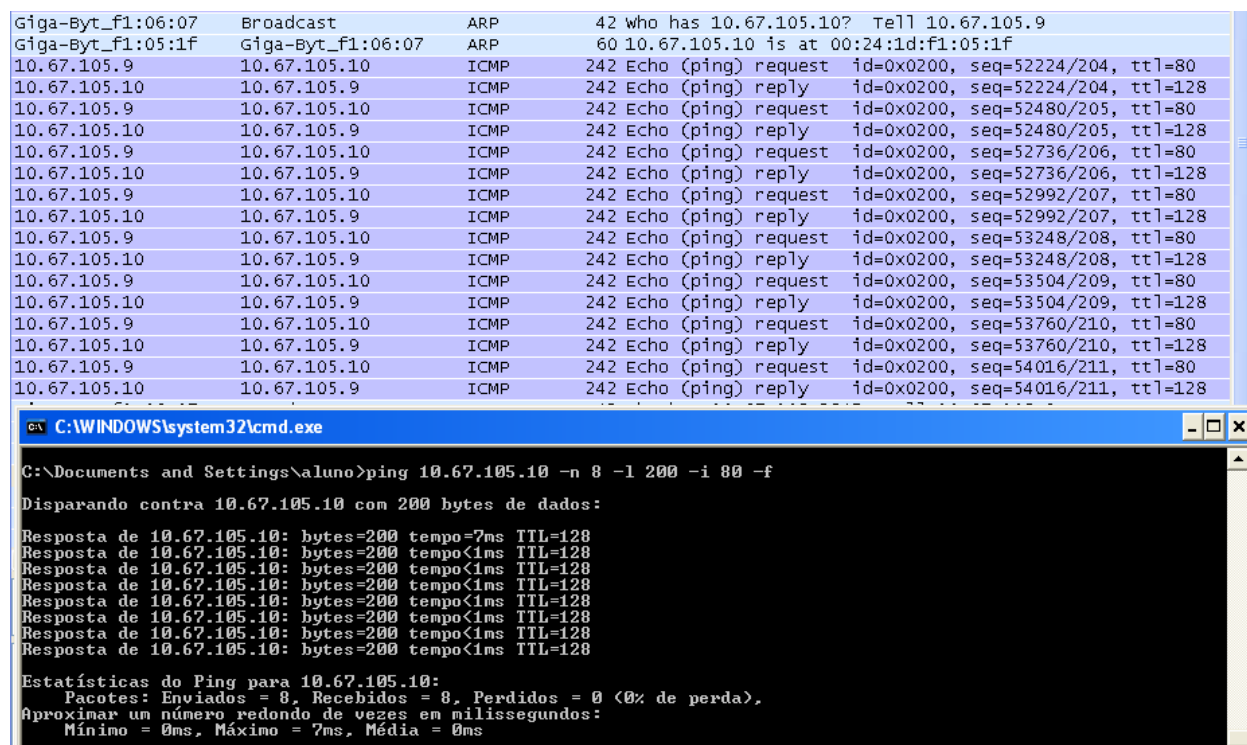


Figura 10: Análise de pacotes com programa sniffer.

Podemos observar que por não saber o MAC de destino, a máquina de origem executa um envio em broadcast de uma mensagem ARP buscando por essa informação. Quando uma máquina souber dessa informação ela envia uma mensagem ARP à origem com o MAC de destino procurado. Em seguida são enviadas as mensagens ICMP, 8 envios e 8 respostas.

Protocolo	End. Nível 2	End. Nível 3	Descrição
ARP request	MACa -> Broadcast	-	Máquina A pede para a rede qual o MAC da máquina de destino.
ARP reply	MACb -> MACa	-	Máquina B possui a informação procurada pela máquina A, logo envia mensagem à máquina A com o MAC procurado.
ICMP request	MACa -> MACb	IPa -> IPb	Origem envia pacote de dados para o destino.
ICMP reply	MACb -> MACa	IPb -> IPa	Destino responde à mensagem recebida.

3. Repetir a tabela acima, porém fazendo ping para uma subrede diferente. Qual a principal diferença?

Protocolo	End. Nível 2	End. Nível 3	Descrição
ICMP request	MACa -> MACb	IPa -> IPb	Origem envia pacote de dados para o destino.
ICMP reply	MACb -> MACa	IPb -> IPa	Destino responde à mensagem recebida.

A diferença é que não é mais enviado a mensagem de ARP, pois a origem conhece o MAC do gateway (roteador) e este conhece o MAC da origem. O gateway encaminha a mensagem adiante até atingir o destino.

4. SOBRE O ARP

a. Explicar o funcionamento do protocolo ARP baseado nos seus campos de cabeçalho. Para isso, pode-se olhar na especificação do protocolo (buscar na RFC) ou analisar no sniffer, pois o mesmo provê todos os campos do protocolo. Explique o motivo do ARP request não conter o endereço MAC do destino (campo zerado), e o motivo que o ARP reply contém todos os campos preenchidos.

O protocolo ARP é usado para encontrar um endereço de enlace (Ethernet, por exemplo), a partir de um endereço de rede (IP, por exemplo). Máquinas que desejam se comunicar e só conhecer o IP uma da outra, usam o protocolo ARP para encontrarem o devido enlace. Para isto, a mensagem do protocolo ARP é enviada em broadcast pedindo o MAC da máquina de destino. Assim que uma máquina tiver a informação solicitada é realizado um reply para a máquina de origem contendo o MAC da máquina de destino.

No cabeçalho ARP estão especificados:

- Tipo de hardware (do nível de enlace, ethernet)
- Tipo de protocolo (do nível de rede, IP)
- Tamanho do endereço de hardware (6)
- Tamanho do endereço de protocolo (4)
- Opcode (algumas opções)
- Endereço MAC da origem (00:24:00:01:04:db)
- Endereço IP da origem (10.67.105.3)
- Endereço MAC do alvo (00:00:00:00:00:00)

- Endereço IP do alvo (10.67.105.254)

A mensagem de request do ARP possui o destino “zerado”, pois é este campo que não é conhecido e por isso é feito um broadcast do pedido na rede. Já a mensagem de reply possui destino definido, que nada mais é do que o MAC da máquina que realizou o request.

b. Explicar o motivo pelo qual o ARP acontece somente na primeira vez que é feito o ping para uma determinada máquina. Dica: utilizar "arp /?" e "arp -a". Quanto tempo dura essa informação?

O ARP ocorre apenas na primeira vez que realizamos o ping, pois até então a tabela de MACs não possuía o MAC do destino e este precisou ser buscado na rede. Em seguida, um novo ping não necessita mais do ARP, pois a tabela já está preenchida com o MAC de destino.

A informação presente na tabela ARP varia de equipamento para equipamento. Em modes Cisco, por exemplo, os registros possuem timeout de 4 horas.

5. Sobre o traceroute

a. Utilizar o traceroute (ou tracert) para descobrir o número de hops e os roteadores por onde os pacotes estão trafegando até:

i. www.ufrgs.br

```

Administrador: C:\Windows\system32\cmd.exe
Microsoft Windows [versão 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Todos os direitos reservados.

C:\Users\jlkgross>tracert www.ufrgs.br
'tracert' não é reconhecido como um comando interno
ou externo, um programa operável ou um arquivo em lotes.

C:\Users\jlkgross>tracert www.ufrgs.br

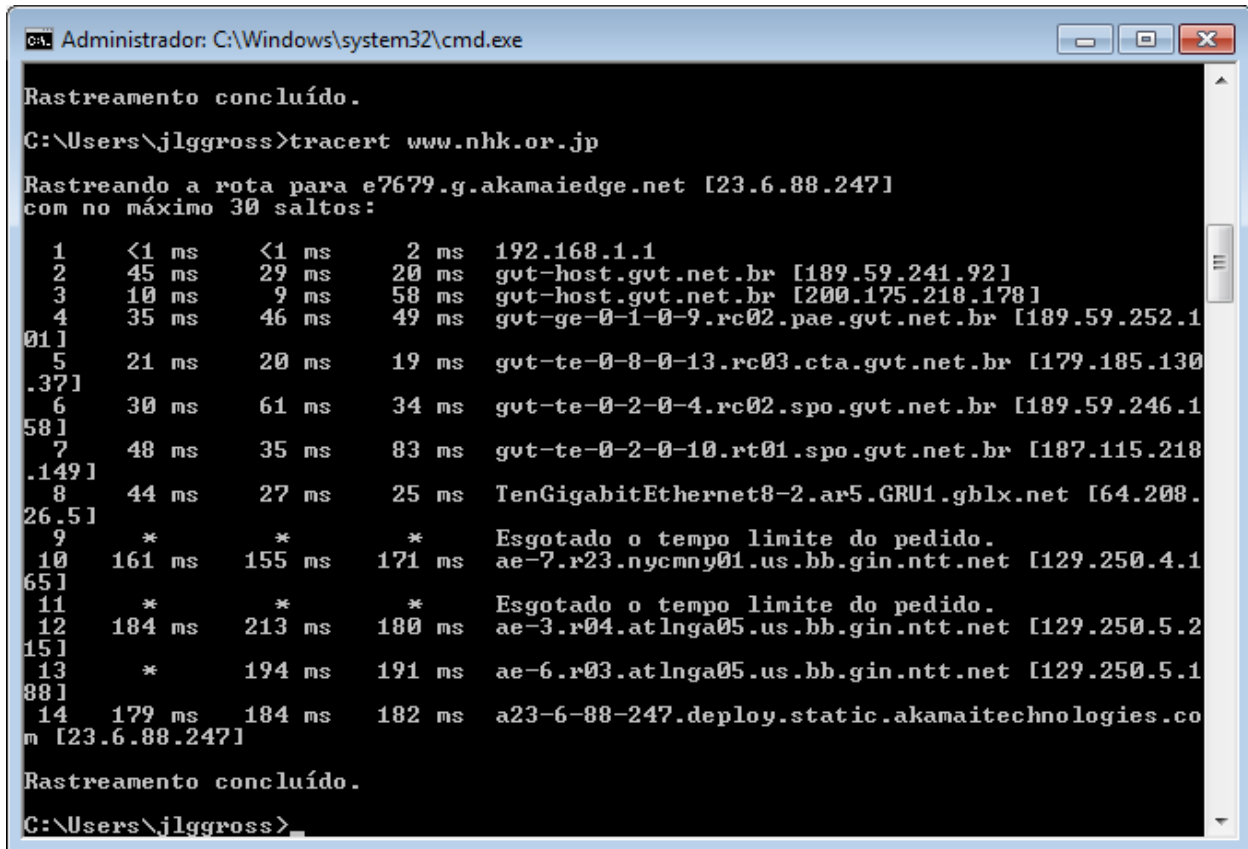
Rastreando a rota para www.ufrgs.br [143.54.2.20]
com no máximo 30 saltos:

 1  <1 ms    <1 ms    <1 ms    192.168.1.1
 2  *        *        *        Esgotado o tempo limite do pedido.
 3  *        *        *        Esgotado o tempo limite do pedido.
 4  89 ms    43 ms    8 ms     gut-ge-0-0-0.rt01.pae.gvt.net.br [189.59.253.13]
 5  15 ms    11 ms    27 ms    as52841.rs.ptt.br [200.219.143.2]
 6  19 ms    7 ms     7 ms     lfs-in.ufrgs.br [143.54.0.241]
 7  *        64 ms    84 ms     www.ufrgs.br [143.54.2.20]

Rastreamento concluído.
C:\Users\jlkgross>_
  
```

Figura 11: Usando tracert para www.ufrgs.br

ii. <http://www.nhk.or.jp>



```
Administrador: C:\Windows\system32\cmd.exe

Rastreamento concluído.

C:\Users\jlkgross>tracert www.nhk.or.jp

Rastreando a rota para e7679.g.akamaiedge.net [23.6.88.247]
com no máximo 30 saltos:

 1  <1 ms    <1 ms    2 ms  192.168.1.1
 2  45 ms    29 ms    20 ms  gvt-host.gvt.net.br [189.59.241.92]
 3  10 ms     9 ms     58 ms  gvt-host.gvt.net.br [200.175.218.178]
 4  35 ms    46 ms    49 ms  gvt-ge-0-1-0-9.rc02.pae.gvt.net.br [189.59.252.1
01]
 5  21 ms    20 ms    19 ms  gvt-te-0-8-0-13.rc03.cta.gvt.net.br [179.185.130
.37]
 6  30 ms    61 ms    34 ms  gvt-te-0-2-0-4.rc02.spo.gvt.net.br [189.59.246.1
58]
 7  48 ms    35 ms    83 ms  gvt-te-0-2-0-10.rt01.spo.gvt.net.br [187.115.218
.149]
 8  44 ms    27 ms    25 ms  TenGigabitEthernet8-2.ar5.GRU1.gblx.net [64.208.
26.5]
 9  *        *        *      Esgotado o tempo limite do pedido.
10 161 ms   155 ms   171 ms ae-7.r23.nycmny01.us.bb.gin.ntt.net [129.250.4.1
65]
11 *        *        *      Esgotado o tempo limite do pedido.
12 184 ms   213 ms   180 ms ae-3.r04.atlga05.us.bb.gin.ntt.net [129.250.5.2
15]
13 *        194 ms   191 ms ae-6.r03.atlga05.us.bb.gin.ntt.net [129.250.5.1
88]
14 179 ms   184 ms   182 ms a23-6-88-247.deploy.static.akamaitechnologies.co
m [23.6.88.247]

Rastreamento concluído.

C:\Users\jlkgross>
```

Figura 12: Usando tracert para www.nhk.or.jp

b. Para que serve o traceroute? Qual sua relação com o TTL? Para que serve o TTL?

O traceroute (ou tracert para windows) é uma ferramenta utilizado para rastrear o caminho que um pacote realiza na rede, da origem até o seu destino. O registro do caminho é feito registrando todos os gateways pelos quais o pacote passa. Isso é feito, enviando um pacote com um TTL (time to live) máximo baixo, assim quando a mensagem atinge o primeiro hop (gateway) o TTL naturalmente é decrementando e se estiver em zero ele é descartado e uma mensagem de resposta "ICMP TIME_EXCEEDED" é enviada à origem, indicando a perda do pacote. Assim, a medida que o traceroute recebe mensagens time_exceeded, ele envia um novo pacote com TTL máximo um pouco maior e vai fazendo isso, capturando todos os gateways intermediários até chegar ao destino.

O TTL serve para indicar a quantidade máxima de gateways pelos quais o paote pode passar até chegar ao destino. A cada gateway o TTL é decrementado e se chegar à zero o pacote é descartado. Esse comportamento é utilizado a favor do traceroute para determinar os gateways intermediários.

6. Instalar o software Polycom PVX (Windows XP) ou Polycom Telepresence m100 (Windows 7) ou Ekiga (Linux ou Windows) e estabelecer uma chamada em duplas. Medir o atraso ida e volta da transmissão com câmera (não é com captura de tela). Disparar o “xnote stopwatch” (ou equivalente) na máquina A. A máquina A filma (com a webcam) a tela da máquina A, transmitindo essa imagem para a máquina B. A máquina B filma a tela. Na máquina A é recebida a imagem transmitida, do seu próprio cronômetro. Captura-se a tela e obtém-se o atraso.

Para este exercício prático, deixamos um cronômetro em execução na máquina A e com a câmera A transmitimos a imagem do cronômetro para a máquina B via Polycom PVX. A máquina B filmou a imagem recebida do cronômetro em execução com a câmera B e a transmitiu para a máquina A. Os os tempos dos cronômetros percebidos pela máquina A foram os seguintes:

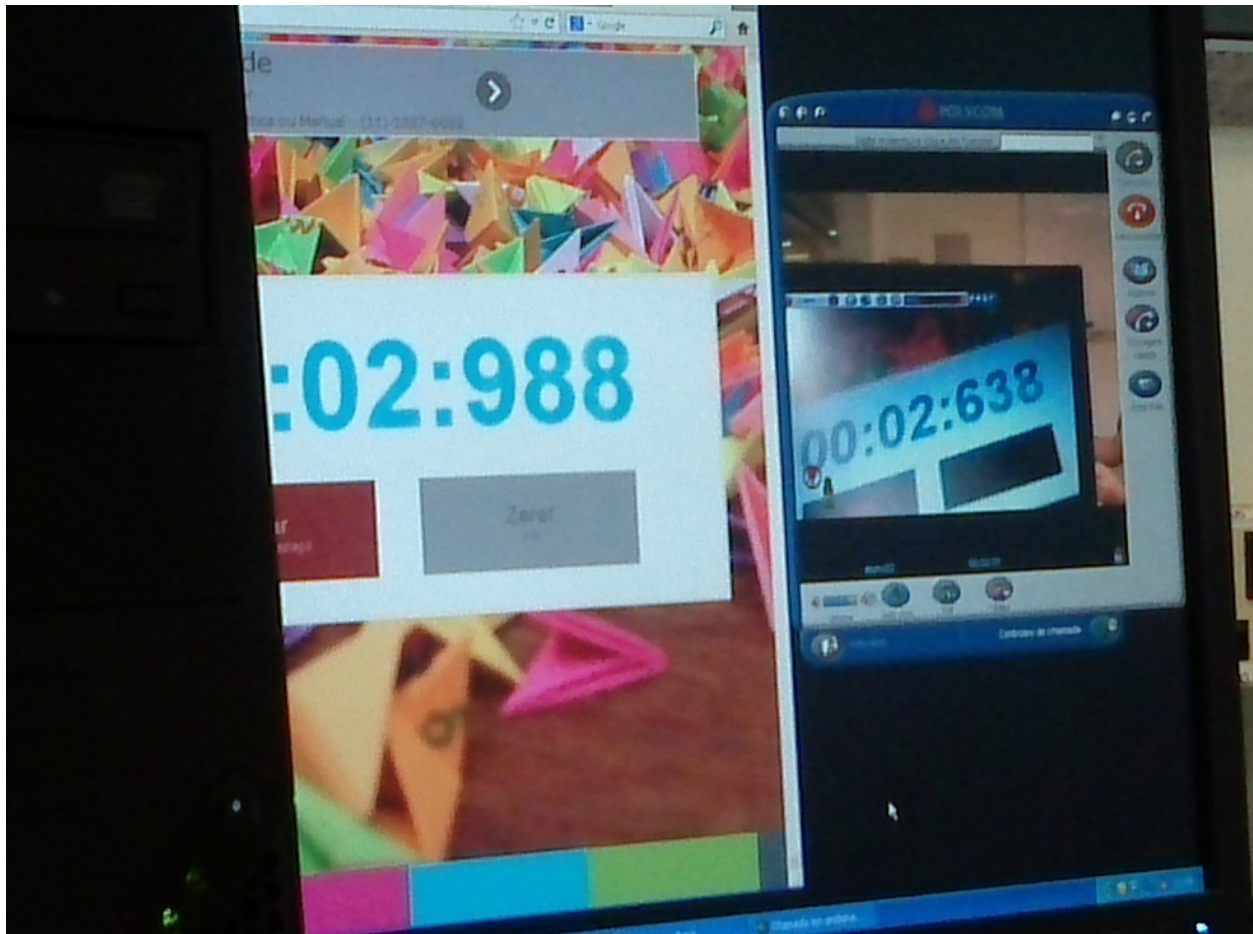


Figura 13: Cronômetro original à esquerda e cronômetro recebido por transmissão de vídeo à direita.

Logo, podemos perceber que há um atraso entre a imagem recebida e o estado atual do cronômetro. Para clarificar: o cronômetro percebido à direita da imagem corresponde a um estado de tempo antigo, sendo o estado de tempo atual do cronômetro o da esquerda. O tempo à direita foi capturado pela câmera A na máquina A, transmitido à máquina B, capturado pela câmera B e transmitido novamente à máquina A.

O atraso foi de $988 - 638 = 350$ milissegundos.

7. Instalar o sniffer de redes Wireshark e fazer uma comunicação em duplas via PVX. Sugestão: usar filtros: `ip.src==...` ou `RTCP` ou `RTP` ou ...

a. Identificar cabeçalhos RTP e RTCP (mostrar uma imagem com RTP e outra com RTCP no wireshark). Qual a diferença de direção e quantidade de pacotes de cada um?

No.	Time	Source	Destination	Protocol	Length	Info
70	16.583772000	10.67.105.2	10.67.105.4	RTP	294	PT=DynamicRTP-Type-127, SSRC=0x15C32201, Seq=0, Time=0
71	16.583823000	10.67.105.2	10.67.105.4	RTP	294	PT=DynamicRTP-Type-127, SSRC=0x15C32201, Seq=1, Time=320
78	16.648204000	10.67.105.2	10.67.105.4	RTP	294	PT=DynamicRTP-Type-127, SSRC=0x15C32201, Seq=2, Time=640
81	16.669696000	10.67.105.2	10.67.105.4	RTP	294	PT=DynamicRTP-Type-127, SSRC=0x15C32201, Seq=3, Time=960
86	16.723396000	10.67.105.2	10.67.105.4	RTP	294	PT=DynamicRTP-Type-127, SSRC=0x15C32201, Seq=4, Time=1280
89	16.739387000	10.67.105.4	10.67.105.2	RTP	294	PT=DynamicRTP-Type-127, SSRC=0xB1818201, Seq=0, Time=0
90	16.744872000	10.67.105.2	10.67.105.4	RTP	294	PT=DynamicRTP-Type-127, SSRC=0x15C32201, Seq=5, Time=1600
94	16.771610000	10.67.105.4	10.67.105.2	RTP	294	PT=DynamicRTP-Type-127, SSRC=0xB1818201, Seq=1, Time=320
97	16.809331000	10.67.105.2	10.67.105.4	RTP	294	PT=DynamicRTP-Type-127, SSRC=0x15C32201, Seq=6, Time=1920
98	16.814552000	10.67.105.4	10.67.105.2	RTP	294	PT=DynamicRTP-Type-127, SSRC=0xB1818201, Seq=2, Time=640
99	16.830829000	10.67.105.2	10.67.105.4	RTP	294	PT=DynamicRTP-Type-127, SSRC=0x15C32201, Seq=7, Time=2240
100	16.846676000	10.67.105.4	10.67.105.2	RTP	294	PT=DynamicRTP-Type-127, SSRC=0xB1818201, Seq=3, Time=960
101	16.889640000	10.67.105.4	10.67.105.2	RTP	294	PT=DynamicRTP-Type-127, SSRC=0xB1818201, Seq=4, Time=1280
102	16.895254000	10.67.105.2	10.67.105.4	RTP	294	PT=DynamicRTP-Type-127, SSRC=0x15C32201, Seq=8, Time=2560
103	16.916738000	10.67.105.2	10.67.105.4	RTP	294	PT=DynamicRTP-Type-127, SSRC=0x15C32201, Seq=9, Time=2880
104	16.932699000	10.67.105.4	10.67.105.2	RTP	294	PT=DynamicRTP-Type-127, SSRC=0xB1818201, Seq=5, Time=1600
105	16.948960000	10.67.105.2	10.67.105.4	RTP	294	PT=DynamicRTP-Type-127, SSRC=0x15C32201, Seq=10, Time=3200
107	16.975704000	10.67.105.4	10.67.105.2	RTP	294	PT=DynamicRTP-Type-127, SSRC=0xB1818201, Seq=6, Time=1920
108	16.991955000	10.67.105.2	10.67.105.4	RTP	294	PT=DynamicRTP-Type-127, SSRC=0x15C32201, Seq=11, Time=3520
109	17.007917000	10.67.105.4	10.67.105.2	RTP	294	PT=DynamicRTP-Type-127, SSRC=0xB1818201, Seq=7, Time=2240
110	17.034919000	10.67.105.2	10.67.105.4	RTP	294	PT=DynamicRTP-Type-127, SSRC=0x15C32201, Seq=12, Time=3840
111	17.050861000	10.67.105.4	10.67.105.2	RTP	294	PT=DynamicRTP-Type-127, SSRC=0xB1818201, Seq=8, Time=2560
112	17.077865000	10.67.105.2	10.67.105.4	RTP	294	PT=DynamicRTP-Type-127, SSRC=0x15C32201, Seq=13, Time=4160
113	17.093816000	10.67.105.4	10.67.105.2	RTP	294	PT=DynamicRTP-Type-127, SSRC=0xB1818201, Seq=9, Time=2880
114	17.115293000	10.67.105.4	10.67.105.2	RTP	294	PT=DynamicRTP-Type-127, SSRC=0xB1818201, Seq=10, Time=3200

Figura 14: Mensagens RTP e RTCP

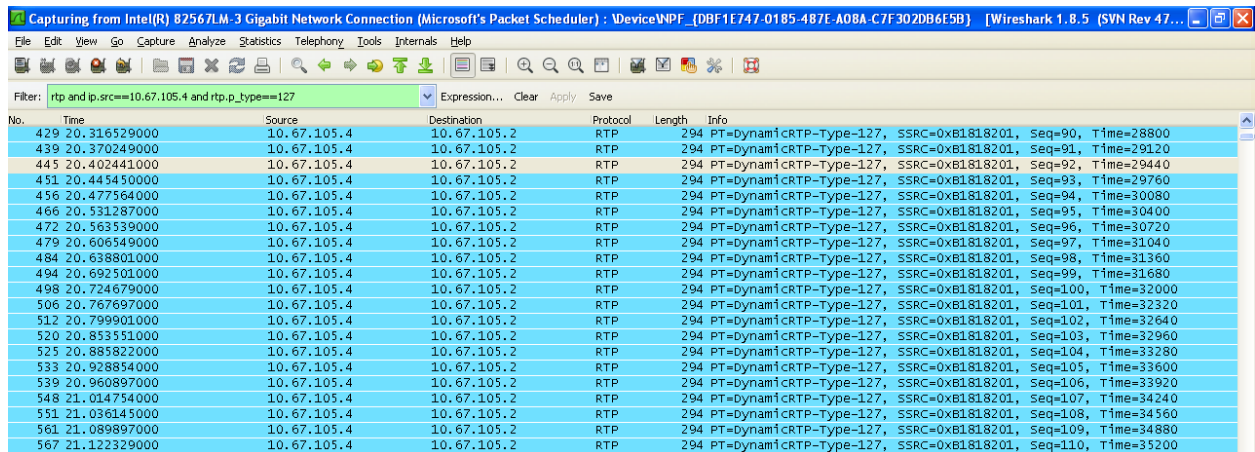
Nesta imagem podemos observar que os dois terminais estão enviando mensagens um para o outro, devido a transmissão de vídeo estabelecida. A quantidade de pacotes que cada máquina manda para a outra é praticamente a mesma tanto de pacote RTP quanto de pacotes RTCP.

b. Para o RTP, identificar e justificar os campos “versão”, “Payload Type”, “Sequence Number”, “timestamp”

- Versão: rfc 1889 version(2) - indica a versão do protocolo, que atualmente é a versão 2.
- Payload Type: DynamicRTP-Type-126 ou 127 - indica o tipo de informação, se de áudio ou de vídeo, por exemplo.
- Sequence Number: valor numérico - indica o número do pacote em uma transmissão.

- Time Stamp: valor numérico - indica à aplicação de destino o tempo no qual a informação do pacote deve ser exibida. Pacotes com o mesmo timestamp são exibidos juntos, pois são partes de informações do mesmo objeto.

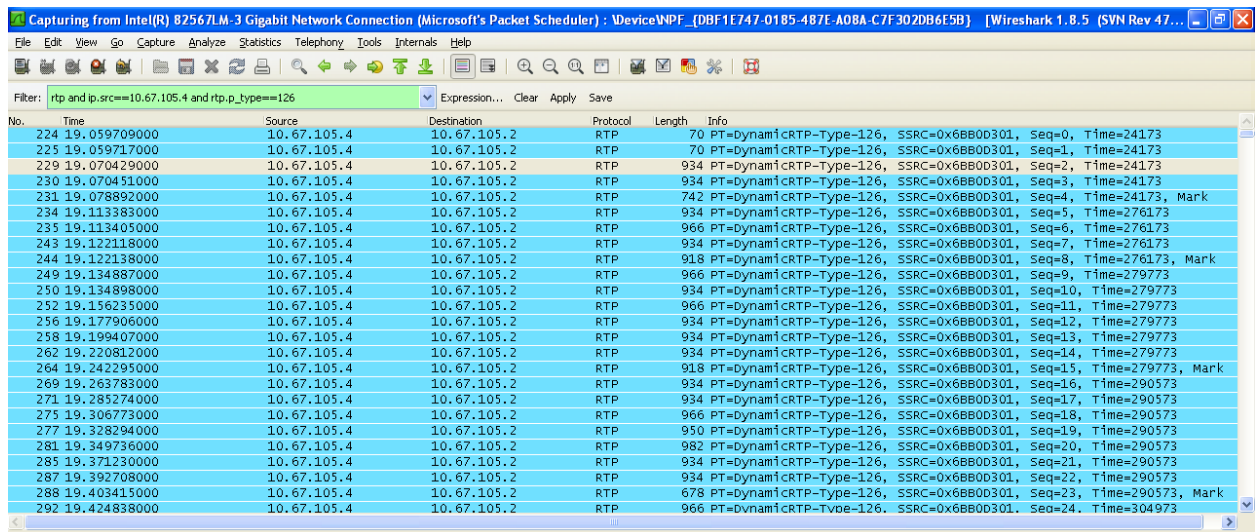
c. Descobrir quais pacotes são de áudio e quais são de vídeo. Justificar, utilizando como base a diferença entre tamanho e quantidade dos pacotes de cada fluxo. Utilize como apoio o timestamp dos pacotes. Sugere-se fortemente filtrar por fluxo.



Filter: `rtp and ip.src==10.67.105.4 and rtp.p_type==127`

No.	Time	Source	Destination	Protocol	Length	Info
429	20.316529000	10.67.105.4	10.67.105.2	RTP	294	PT=DynamiCRTP-Type-127, SSRC=0xB1818201, Seq=90, Time=28800
439	20.370249000	10.67.105.4	10.67.105.2	RTP	294	PT=DynamiCRTP-Type-127, SSRC=0xB1818201, Seq=91, Time=29120
445	20.402441000	10.67.105.4	10.67.105.2	RTP	294	PT=DynamiCRTP-Type-127, SSRC=0xB1818201, Seq=92, Time=29440
451	20.445450000	10.67.105.4	10.67.105.2	RTP	294	PT=DynamiCRTP-Type-127, SSRC=0xB1818201, Seq=93, Time=29760
456	20.477564000	10.67.105.4	10.67.105.2	RTP	294	PT=DynamiCRTP-Type-127, SSRC=0xB1818201, Seq=94, Time=30080
466	20.531287000	10.67.105.4	10.67.105.2	RTP	294	PT=DynamiCRTP-Type-127, SSRC=0xB1818201, Seq=95, Time=30400
472	20.563539000	10.67.105.4	10.67.105.2	RTP	294	PT=DynamiCRTP-Type-127, SSRC=0xB1818201, Seq=96, Time=30720
479	20.606549000	10.67.105.4	10.67.105.2	RTP	294	PT=DynamiCRTP-Type-127, SSRC=0xB1818201, Seq=97, Time=31040
484	20.638801000	10.67.105.4	10.67.105.2	RTP	294	PT=DynamiCRTP-Type-127, SSRC=0xB1818201, Seq=98, Time=31360
494	20.692501000	10.67.105.4	10.67.105.2	RTP	294	PT=DynamiCRTP-Type-127, SSRC=0xB1818201, Seq=99, Time=31680
498	20.724679000	10.67.105.4	10.67.105.2	RTP	294	PT=DynamiCRTP-Type-127, SSRC=0xB1818201, Seq=100, Time=32000
506	20.767697000	10.67.105.4	10.67.105.2	RTP	294	PT=DynamiCRTP-Type-127, SSRC=0xB1818201, Seq=101, Time=32320
512	20.799901000	10.67.105.4	10.67.105.2	RTP	294	PT=DynamiCRTP-Type-127, SSRC=0xB1818201, Seq=102, Time=32640
520	20.853551000	10.67.105.4	10.67.105.2	RTP	294	PT=DynamiCRTP-Type-127, SSRC=0xB1818201, Seq=103, Time=32960
525	20.885822000	10.67.105.4	10.67.105.2	RTP	294	PT=DynamiCRTP-Type-127, SSRC=0xB1818201, Seq=104, Time=33280
533	20.928854000	10.67.105.4	10.67.105.2	RTP	294	PT=DynamiCRTP-Type-127, SSRC=0xB1818201, Seq=105, Time=33600
539	20.960897000	10.67.105.4	10.67.105.2	RTP	294	PT=DynamiCRTP-Type-127, SSRC=0xB1818201, Seq=106, Time=33920
548	21.014754000	10.67.105.4	10.67.105.2	RTP	294	PT=DynamiCRTP-Type-127, SSRC=0xB1818201, Seq=107, Time=34240
551	21.036145000	10.67.105.4	10.67.105.2	RTP	294	PT=DynamiCRTP-Type-127, SSRC=0xB1818201, Seq=108, Time=34560
561	21.089897000	10.67.105.4	10.67.105.2	RTP	294	PT=DynamiCRTP-Type-127, SSRC=0xB1818201, Seq=109, Time=34880
567	21.122329000	10.67.105.4	10.67.105.2	RTP	294	PT=DynamiCRTP-Type-127, SSRC=0xB1818201, Seq=110, Time=35200

Figura 15: Pacotes de áudio.



Filter: `rtp and ip.src==10.67.105.4 and rtp.p_type==126`

No.	Time	Source	Destination	Protocol	Length	Info
224	19.059709000	10.67.105.4	10.67.105.2	RTP	70	PT=DynamiCRTP-Type-126, SSRC=0x6B800301, Seq=0, Time=24173
225	19.059717000	10.67.105.4	10.67.105.2	RTP	70	PT=DynamiCRTP-Type-126, SSRC=0x6B800301, Seq=1, Time=24173
229	19.070429000	10.67.105.4	10.67.105.2	RTP	934	PT=DynamiCRTP-Type-126, SSRC=0x6B800301, Seq=2, Time=24173
230	19.070451000	10.67.105.4	10.67.105.2	RTP	934	PT=DynamiCRTP-Type-126, SSRC=0x6B800301, Seq=3, Time=24173
231	19.078892000	10.67.105.4	10.67.105.2	RTP	742	PT=DynamiCRTP-Type-126, SSRC=0x6B800301, Seq=4, Time=24173, Mark
234	19.113383000	10.67.105.4	10.67.105.2	RTP	934	PT=DynamiCRTP-Type-126, SSRC=0x6B800301, Seq=5, Time=276173
235	19.113405000	10.67.105.4	10.67.105.2	RTP	966	PT=DynamiCRTP-Type-126, SSRC=0x6B800301, Seq=6, Time=276173
243	19.122118000	10.67.105.4	10.67.105.2	RTP	934	PT=DynamiCRTP-Type-126, SSRC=0x6B800301, Seq=7, Time=276173
244	19.122138000	10.67.105.4	10.67.105.2	RTP	918	PT=DynamiCRTP-Type-126, SSRC=0x6B800301, Seq=8, Time=276173, Mark
249	19.134887000	10.67.105.4	10.67.105.2	RTP	966	PT=DynamiCRTP-Type-126, SSRC=0x6B800301, Seq=9, Time=279773
250	19.134898000	10.67.105.4	10.67.105.2	RTP	934	PT=DynamiCRTP-Type-126, SSRC=0x6B800301, Seq=10, Time=279773
252	19.156235000	10.67.105.4	10.67.105.2	RTP	966	PT=DynamiCRTP-Type-126, SSRC=0x6B800301, Seq=11, Time=279773
256	19.177906000	10.67.105.4	10.67.105.2	RTP	934	PT=DynamiCRTP-Type-126, SSRC=0x6B800301, Seq=12, Time=279773
258	19.199407000	10.67.105.4	10.67.105.2	RTP	934	PT=DynamiCRTP-Type-126, SSRC=0x6B800301, Seq=13, Time=279773
262	19.220812000	10.67.105.4	10.67.105.2	RTP	934	PT=DynamiCRTP-Type-126, SSRC=0x6B800301, Seq=14, Time=279773
264	19.242295000	10.67.105.4	10.67.105.2	RTP	918	PT=DynamiCRTP-Type-126, SSRC=0x6B800301, Seq=15, Time=279773, Mark
269	19.263783000	10.67.105.4	10.67.105.2	RTP	934	PT=DynamiCRTP-Type-126, SSRC=0x6B800301, Seq=16, Time=290573
271	19.285274000	10.67.105.4	10.67.105.2	RTP	934	PT=DynamiCRTP-Type-126, SSRC=0x6B800301, Seq=17, Time=290573
275	19.306773000	10.67.105.4	10.67.105.2	RTP	966	PT=DynamiCRTP-Type-126, SSRC=0x6B800301, Seq=18, Time=290573
277	19.328294000	10.67.105.4	10.67.105.2	RTP	950	PT=DynamiCRTP-Type-126, SSRC=0x6B800301, Seq=19, Time=290573
281	19.349736000	10.67.105.4	10.67.105.2	RTP	982	PT=DynamiCRTP-Type-126, SSRC=0x6B800301, Seq=20, Time=290573
285	19.371230000	10.67.105.4	10.67.105.2	RTP	934	PT=DynamiCRTP-Type-126, SSRC=0x6B800301, Seq=21, Time=290573
287	19.392708000	10.67.105.4	10.67.105.2	RTP	934	PT=DynamiCRTP-Type-126, SSRC=0x6B800301, Seq=22, Time=290573
288	19.403415000	10.67.105.4	10.67.105.2	RTP	678	PT=DynamiCRTP-Type-126, SSRC=0x6B800301, Seq=23, Time=290573, Mark
292	19.424838000	10.67.105.4	10.67.105.2	RTP	966	PT=DynamiCRTP-Type-126, SSRC=0x6B800301, Seq=24, Time=304973

Figura 16: Pacote de vídeo.

Os pacotes de áudio possuem tamanho menor se comparado aos pacotes de vídeo, todos são de tamanho 294, enquanto os de vídeo ficam acima de 900 geralmente. Pacotes de áudio possuem payload type igual a 127 e pacotes de vídeo payload type de 126. Alguns pacotes de

vídeo possuem o mesmo timestamp. Isso se justifica pois alguns pacotes possuem informações de partes de uma imagem, logo, cada parte da imagem é enviada em um pacote e todas essas partes devem ser exibidas ao mesmo tempo.

8. Para a codificação de áudio utilizada e características do laboratório, calcule:

a. Tempo médio de inserção

O tempo médio de inserção é dada pela divisão entre o tamanho do quadro em bits (L) e a taxa de bits (T). Para um quadro de áudio de 294 bytes, ou 2352 bits e uma taxa transmissão de 100Mbps/s temos:

$$\text{Tempo de inserção} = L/T = 2352/100.000.000 = 23,52 \text{ us}$$

b. Atraso no meio físico, supondo 40m a distância entre sua máquina e o switch do INF.

Assumindo que a velocidade da luz (C) é igual a 3×10^8 m/s o atraso no meio físico é:

$$\text{Atraso no meio físico} = \text{distância} / \text{velocidade} = 2 * 40\text{m} / (2/3)C \text{ m/s} = 400\text{ns}$$

9. Obtenha um histograma do valor médio do tamanho dos pacotes que trafegaram pela rede durante o período considerado. Ver opção “statistics+packet lengths”. Faça a análise duas vezes:

- a) iniciando a captura e navegando na web (perfil navegação web);
- b) iniciando a captura e fazendo um download de arquivo de tamanho razoavelmente grande (acima de 10 Mbytes). Compare os resultados em relação ao tamanho do pacote. Explique.

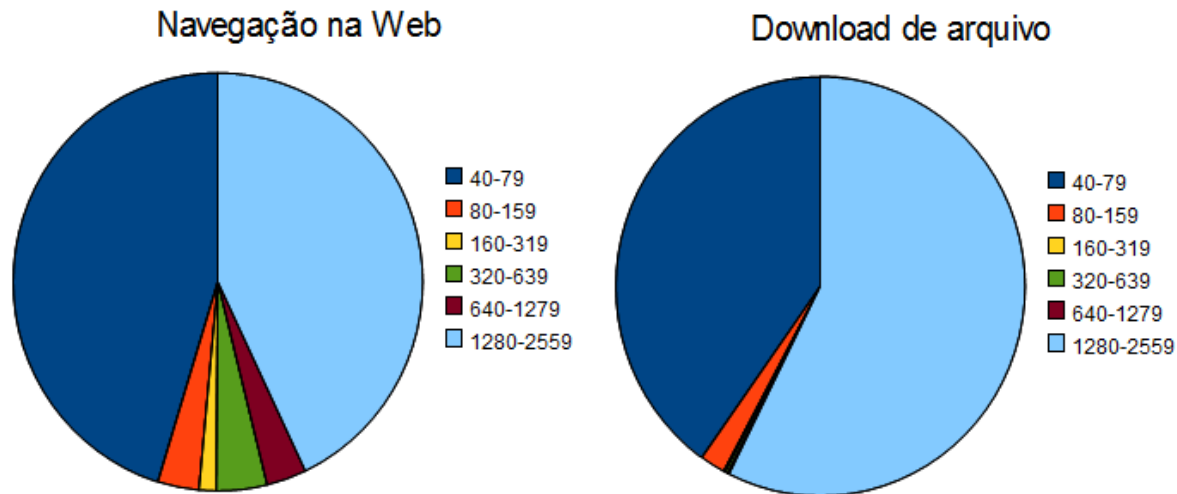


Figura 17: Distribuição de pacotes de acordo com a atividade.

As figuras acima mostram a quantidade de pacotes de cada intervalo de tamanho que foram transmitidas durante uma atividade de navegação na web e ao fazer o download de um arquivo. Em ambos os casos as maiores quantidades de arquivos transmitidos foram para tamanho de 40-79 e 1280-2559. Durante a navegação web a distribuição desses pacotes ficou em 45,27% para 40-79, pela presença de mensagens de controle e acks e 42,98 para 1280-2559, carregamento de imagens, vídeos e propagandas. Já para download de um arquivo essa distribuição passou para 40,18% para 40-79, também por possuir mensagens de ack e de controle e 57,36% para 1280-2559. O aumento dos pacotes de 1280-2559 para download de um arquivo ocorreu a haver um download de informações grande em andamento e por isso pacotes grandes são mais frequentes na rede.

10. Faça uma análise completa de um quadro MAC que contenha encapsulado um pacote IP e TCP (sugestão: faça um download qualquer). Liste todos os valores dos diversos campos encontrados no cabeçalho do quadro MAC e nos cabeçalhos do pacote IP e do segmento TCP encapsulado. Explique os valores encontrados e suas unidades quando for o caso.


```
Frame 1828: 1434 bytes on wire (11472 bits), 1434 bytes captured (11472 bits) on interface 0
Ethernet II, Src: Cisco_9f:fe:c8 (00:21:a0:9f:fe:c8), Dst: Dell_e5:38:aa (f0:4d:a2:e5:38:aa)
  Destination: Dell_e5:38:aa (f0:4d:a2:e5:38:aa)
  Source: Cisco_9f:fe:c8 (00:21:a0:9f:fe:c8)
  Type: IP (0x0800)
Internet Protocol Version 4, Src: 69.4.231.52 (69.4.231.52), Dst: 143.54.13.240 (143.54.13.240)
  Version: 4
  Header length: 20 bytes
  Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))
  Total Length: 1420
  Identification: 0x9cdf (40159)
  Flags: 0x02 (Don't Fragment)
  Fragment offset: 0
  Time to live: 44
  Protocol: TCP (6)
  Header checksum: 0xe32d [correct]
  Source: 69.4.231.52 (69.4.231.52)
  Destination: 143.54.13.240 (143.54.13.240)
  [Source GeoIP: Unknown]
  [Destination GeoIP: Unknown]
Transmission Control Protocol, Src Port: http (80), Dst Port: 49700 (49700), Seq: 154561, Ack: 469, Len: 1380
  Source port: http (80)
  Destination port: 49700 (49700)
  [Stream index: 29]
  Sequence number: 154561 (relative sequence number)
  [Next sequence number: 155941 (relative sequence number)]
  Acknowledgment number: 469 (relative ack number)
  Header length: 20 bytes
  Flags: 0x010 (ACK)
  window size value: 14
  [Calculated window size: 7168]
  [window size scaling factor: 512]
  Checksum: 0xb176 [validation disabled]
  [SEQ/ACK analysis]
  TCP segment data (1380 bytes)
```

Figura 18: Captura no Wireshark.

Ethernet:

Destination: A máquina destino do pacote (O PC que está fazendo o download)

Source: Quem enviou o pacote (o roteador)

Type: IP

IP:

Versão: IPv4

Tamanho do cabeçalho IP: 20 bytes

Serviços extras: Nenhum

Tamanho total (cabeçalho e dados): 1420 bytes.

Identification: Código para que os diversos fragmentos refiram-se ao mesmo pacote IP.

Flags: Identifica se houve fragmentação. Nesse exemplo não houve.

Fragment offset: No exemplo é 0 pois não houve fragmentação.

TTL: 44. Quando esse valor chega a 0 o pacote é descartado. Cada roteador diminui em 1 esse valor.

Protocol: O protocolo da camada superior. No exemplo é TCP.

Checksum: Checksum para testar se o pacote foi corrompido.

Source: O endereço IPv4 da fonte. No exemplo, externo à rede.

Destination: O endereço IPv4 do PC(destino final)

TCP:

Source port: 80. Padrão HTTP.

Destination port: 49700. Porta aberta pelo cliente.

Sequence: 154561. O número de sequência do pacote na sessão.

ACK: Esse é um pacote de ack. 469 é o número do ack n.

Header lenght: Tamanho do cabeçalho. 20 bytes do TCP.

Flags: 0x10 sinaliza que esse é um pacote ACK.

Window size: Tamanho da janela: 14(janela deslizante).

Checksum: 16 bits de checksum.

11. Utilizar o software Nuttcp ou Iperf (iperf -h para help), que deve ser disparado em duas máquinas, uma sendo servidor “iperf -s” e outra cliente. Tem que disparar o servidor de forma diferente para teste em UDP ou TCP. “iperf -s -u” para UDP ou “iperf -s” para TCP.

EXEMPLO DE CLIENTE COM UDP:

iperf -f m -i 1 -c “ipservidor” -t 30 -p 2000 -u -b 10M -l 1400 (formato em Mbit/s, informa banda enviada a cada segundo, tempo de 30s, porta 2000, banda máxima de 10Mbit/s, tamanho de pacote 1400 bytes). No relatório devem constar o resultado dos três seguintes objetivos:

a. Mostrar linhas de comando utilizadas e gerar gráfico udp para 3 bandas diferentes (ex: 1Mbit/s, 10Mbit/s e 30Mbit/s). Mostrar e explicar o gráfico gerado.

1400 bytes datagrams

```
iperf -f m -c 192.168.1.100 -p 2024 -u -b 1M -l 1400 -t 30
```

Transfer:3.58Mbytes

Sent: 2680 datagrams

```
iperf -f m -c 192.168.1.100 -p 2024 -u -b 10M -l 1400 -t 30
```

Transfer: 35.8Mbytes

Sent: 36787 datagrams

```
iperf -f m -c 192.168.1.100 -p 2024 -u -b 30M -l 1400 -t 30
```

Transfer: 107Mbytes

Sent: 80430 datagrams

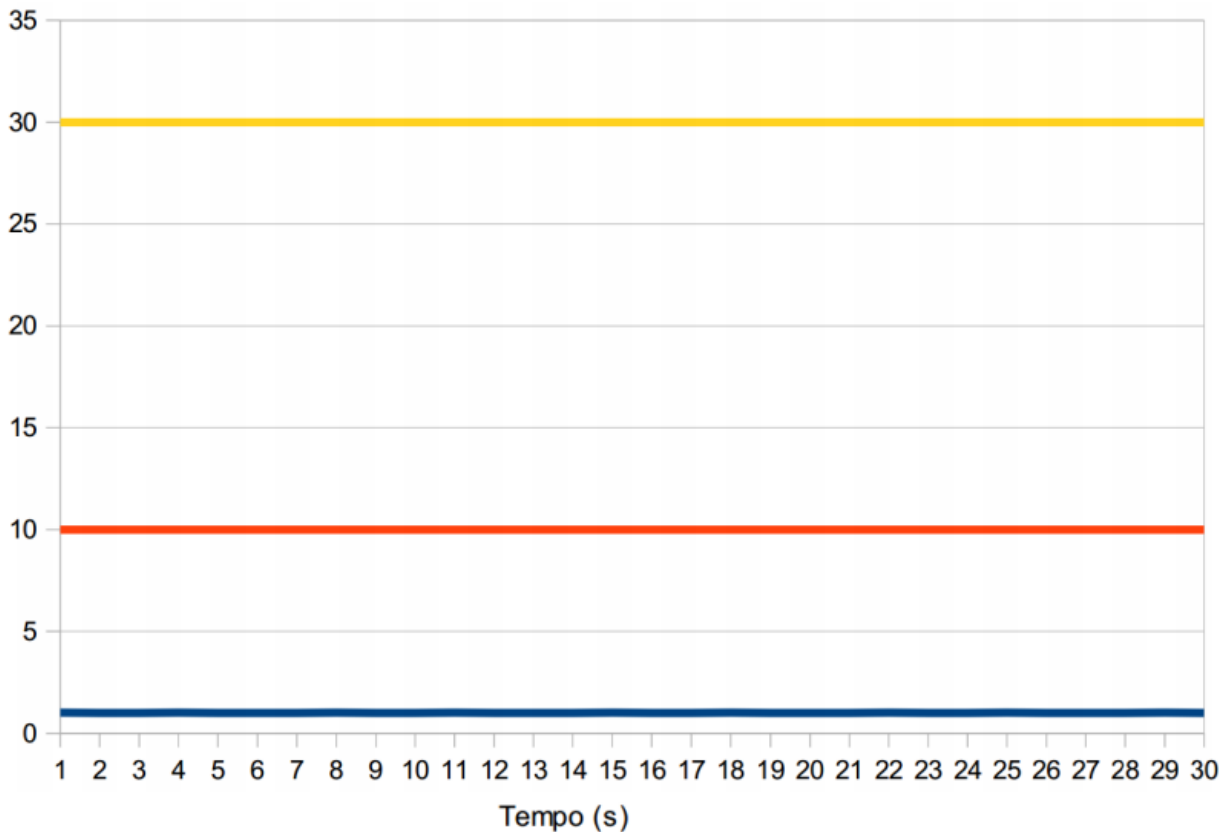


Figura 19: UDP para 3 bandas diferentes.

b. Mostrar linhas de comando utilizadas e gerar gráfico TCP da estação incrementando o número de conexões para uma máquina servidora. Pode-se fazer de duas formas:

a) aumentando o número de clientes gradativamente, com 1, 2 e 3 clientes, utilizando máquinas diferentes para clientes;

b) utilizando o resultado do próprio iperf (opção “-i 1”), que mostra a média a cada segundo, e depois gerando o gráfico com outra ferramenta. Nesse caso, bastam duas máquinas com 3 janelas cliente e 3 servidoras. O objetivo é ver a adaptação do TCP (tcp-friendly).

Comando

```
iperf c 143.54.13.240 f m i 1 t 150 p 2000 l 1400 &
(sleep 30; iperf c 143.54.13.240 f m i 1 t 60 p 2001 l 1400) &
(sleep 60; iperf c 143.54.13.240 f m i 1 t 60 p 2002 ) &
```

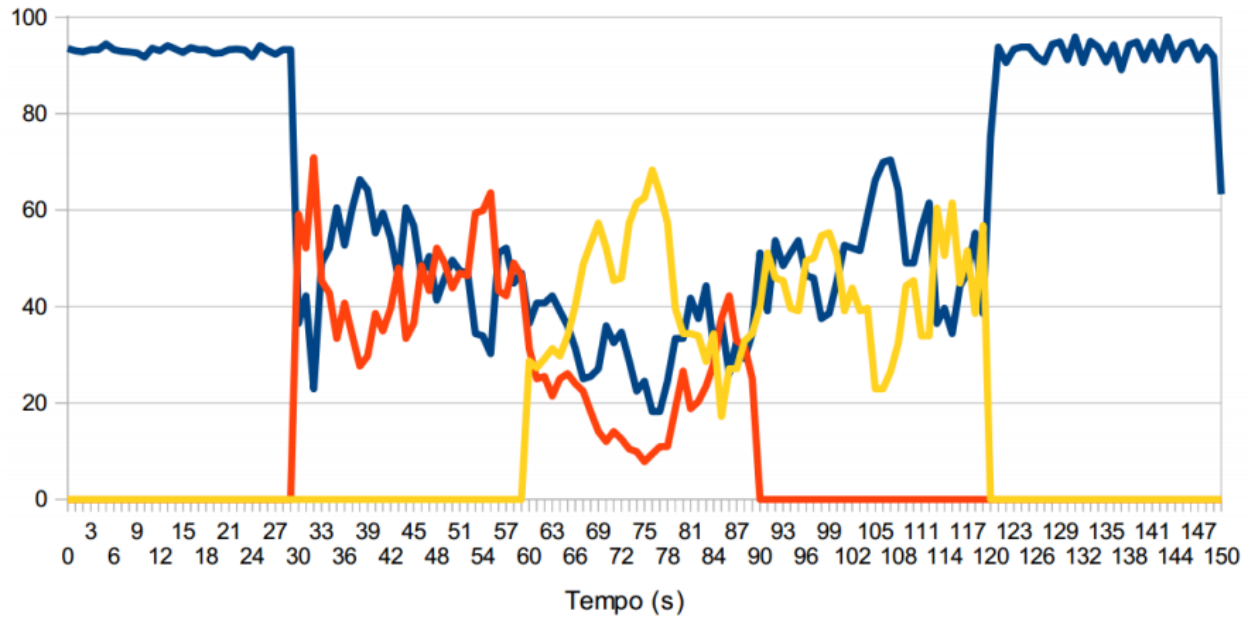


Figura 20: TCP com 3 clientes.

OBS: não esquecer que definição de banda só faz sentido com o UDP. No TCP, não se deve utilizar flags de controle de taxa, pois o controle é feito em nível 4.

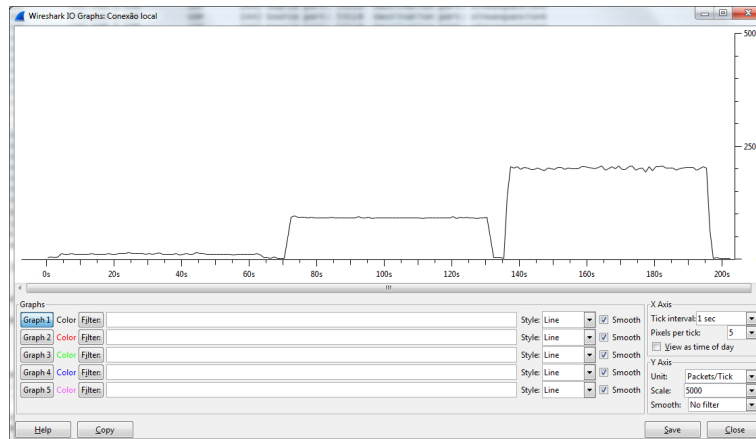
12. Utilizando a ferramenta Wireshark, obtenha a curva de variação da carga da rede local da sua máquina. Explique, sucintamente, como foi obtida e comente os resultados obtidos. Para esta tarefa considere uma estatística de 1 a 5 minutos. Ver opção “statistics+io-graph”. Altere o uso da rede (através do iperf ou outro método) e explique as variações na rede. Gere gráficos de TCP e UDP e compare os resultados.

UDP:

Comandos UDP:

```
iperf -f m -c 192.168.1.100 -p 2024 -u -b 1M 1400 -t 60
iperf -f m -c 192.168.1.100 -p 2024 -u -b 10M 1400 -t 60
iperf -f m -c 192.168.1.100 -p 2024 -u -b 30M 1400 -t 60
```

Os comandos foram executados sequencialmente. A transmissão foi limitada em: 1M, 10M e 30M, respectivamente.

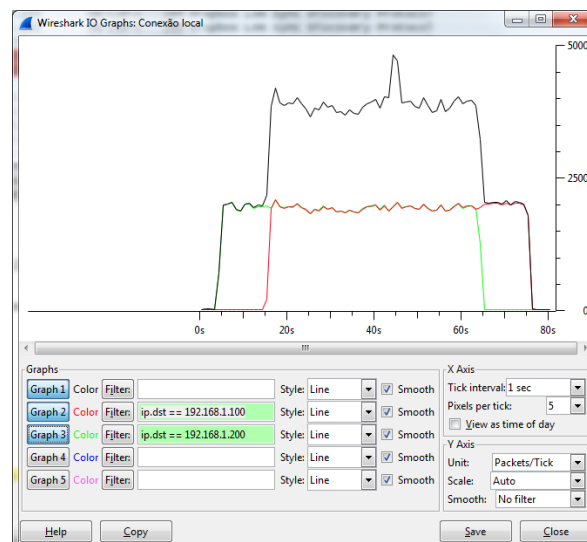


Comandos UDP:

Verde: iperf -f m -c 192.168.1.200 -p 2024 -u -b 30M 1400 -t 60

Vermelho: iperf -f m -c 192.168.1.100 -p 2024 -u -b 30M 1400 -t 60

Comandos executados paralelamente, com uma espera de 10s entre o primeiro e o segundo. Como a banda foram limitados a 30M/s, e a banda disponível é 100M/s. Ambos transmitiram a 30M/s, com a tráfego total de rede chegando a 60M/s.



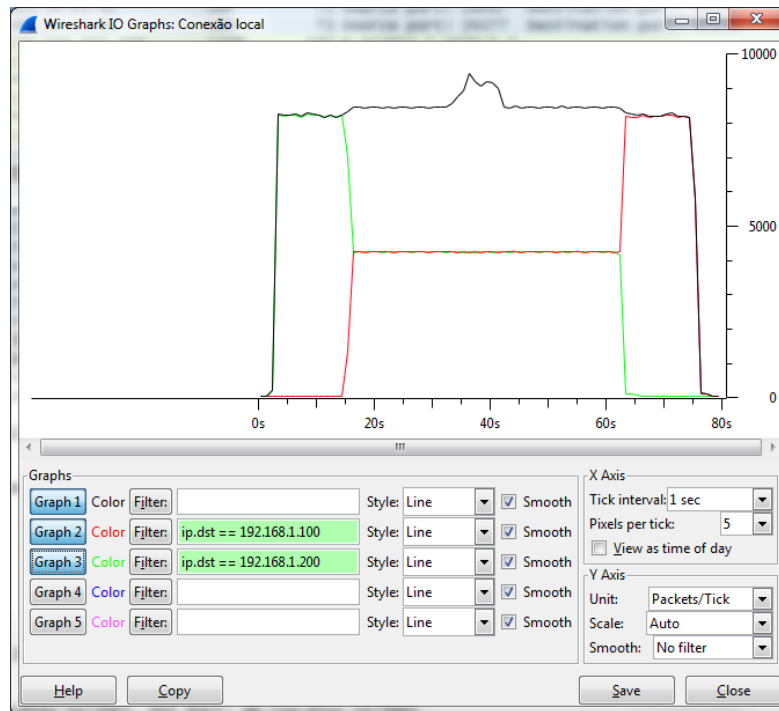
Comandos UDP:

Verde: iperf -f m -c 192.168.1.200 -p 2024 -u -b 100M -I 32k -t 60

Vermelho: iperf -f m -c 192.168.1.100 -p 2024 -u -b 100M -I 32k -t 60

Comandos executados de maneira semelhante ao exemplo anterior, mas como a banda disponível é de 100M/s, no momento que são executados paralelamente as taxas de transmissão de ambas diminuem para 50M/s com o tráfego total de rede se mantendo

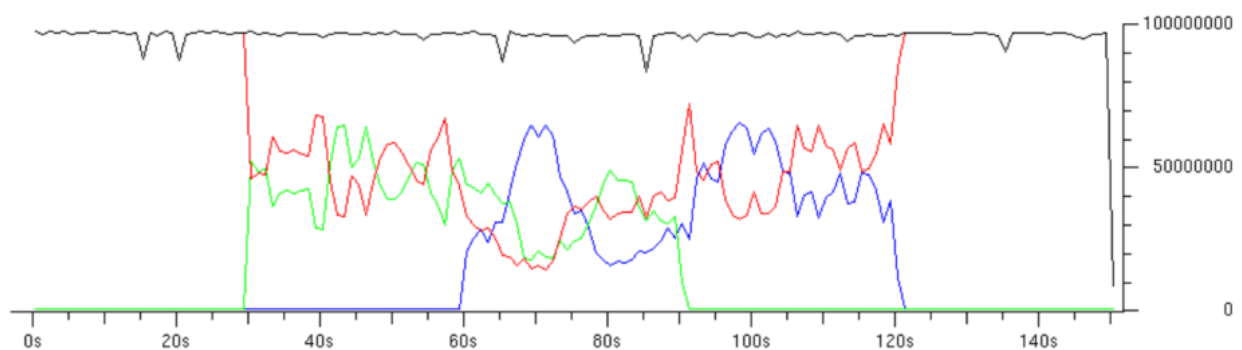
constante em 100M/s.



TCP

Comando TCP:

```
iperf c 143.54.13.240 f m i 1 t 150 p 2000 l 1400 &  
(sleep 30; iperf c 143.54.13.240 f m i 1 t 60 p 2001 l 1400) &  
(sleep 60; iperf c 143.54.13.240 f m i 1 t 60 p 2002 ) &
```



Referências

- TTL padrão para Windows e Linus. Disponível em: <http://www.vivaolinux.com.br/dica/Entendendo-o-campo-TTL-do-ping>. Acesso em 19/10/2013.
- Time to Live. Disponível em: http://en.wikipedia.org/wiki/Time_to_live. Acesso em 19/10/2013.
- Address Resolution Protocol. Disponível em: http://en.wikipedia.org/wiki/Address_Resolution_Protocol. Acesso em: 19/10/2013.
- ARP chaching and timeout. Disponível em: <http://www.networkers-online.com/blog/2009/02/arp-caching-and-timeout/>. Acesso em: 19/10/2013.
- Traceroute - Funcionamento. Disponível em: <http://penta.ufrgs.br/uel/graziela/graznw13.htm>. Acesso em: 10/11/2013.
- Traceroute - Informações. Disponível em: <http://pt.wikipedia.org/wiki/Traceroute>. Acesso em: 10/11/2013.