Camada de Ligação Lógica: Ethernet e Protocolo ARP

J.G[1], M.F[2], P.G.[3], T.C.[4]

Licenciatura em Ciências da Computação (LCC)

Universidade do Minho, R. da Universidade, 4710-057 Braga
Licenciatura em Ciências da Computação
gcii@reitoria.uminho.pt

---

[1] João Guedes - A94013
[2] Miguel Freitas - A91635
[3] Pedro Gomes - A91647
[4] Tomás Campinho - A91668

TP2: Camada de Ligação Lógica: Ethernet e Protocolo ARP

Resumo

Neste trabalho procurámos explorar, estudar e explicitar, de uma forma geral, a camada de ligação lógica, concentrando-nos mais na tecnologia Ethernet e o protocolo ARP (Adress Resolution Protocol). Primeiramente, fizemos a captura e análise de tramas Ethernet usando a aplicação Wireshark. Utilizando a rede Ethernet da sala de aula conseguimos analisar o tráfego com base no conteúdo da trama capturada que contém o número de ordem da sequência de bytes correspondente à mensagem HTTP GET enviada do nosso computador para o servidor, tal como a mensagem de resposta HTTP Response. Isto permitiu-nos ver os vários endereços do nível de rede (IP) e endereços nível de ligação lógica (MAC) da interface ativa do nosso computador e também do sistema de destino da trama. De seguida, observamos e analisamos o protocolo ARP em operação. Nesta secção, novamente com o auxílio da aplicação Wireshark, fizemos a captura e localizamos o envio e recepção de mensagens ARP. Para além disto, estudamos o ARP numa topologia CORE, onde podemos obter e verificar endereços de IP, modificações de cache e executar pings entres os diversos sistemas.

Palavras-chave: ARP, IP, PDU, MAC, CORE, HTTP, Wireshark, Ethernet, Router, Host, Switch;

TP2: Camada de Ligação Lógica: Ethernet e Protocolo ARP

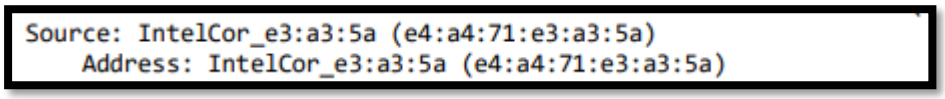
TP2: Camada de Ligação Lógica: Ethernet e Protocolo ARP (Parte I - Respostas)

Captura e análise de Tramas Ethernet.

Abrimos o wireshark e inciamos a captura. Uma vez tudo pronto, entramos apartir do Google Chrome no endereço https://cesium.di.uminho.pt/. Os dados obtidos nesta caputra estão presentes no presente documento.

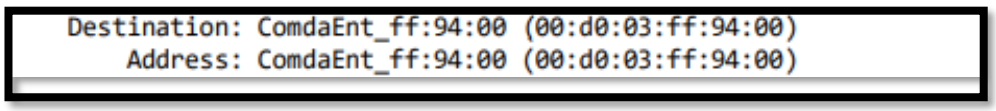
1. **Qual é o endereço MAC da interface ativa do seu computador?**

O endereço Mac da interface ativa do computador utilizado pelo nosso grupo é o seguinte: ***e4:a4:71:e3:a3:5a.*** Obtemos estes valores quando iniciamos a captura de *wireshark*.



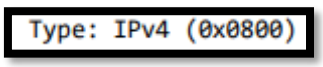
2. **Qual é o endereço MAC destino da trama? A que sistema é destinada essa trama, será o endereço Ethernet do servidor http para cesium.di.uminho.pt? Justifique**

O endereço do destino da trama é o router (Gateway), porque o servidor está numa rede diferente, então primeiro terá de passar pelo router para posteriormente ser reencaminhada para o servidor do https://cesium.di.uminho.pt/, como pode ser observado na imagem capturada:



3. **Qual o valor hexadecimal do campo Type da trama Ethernet? O que significa?**

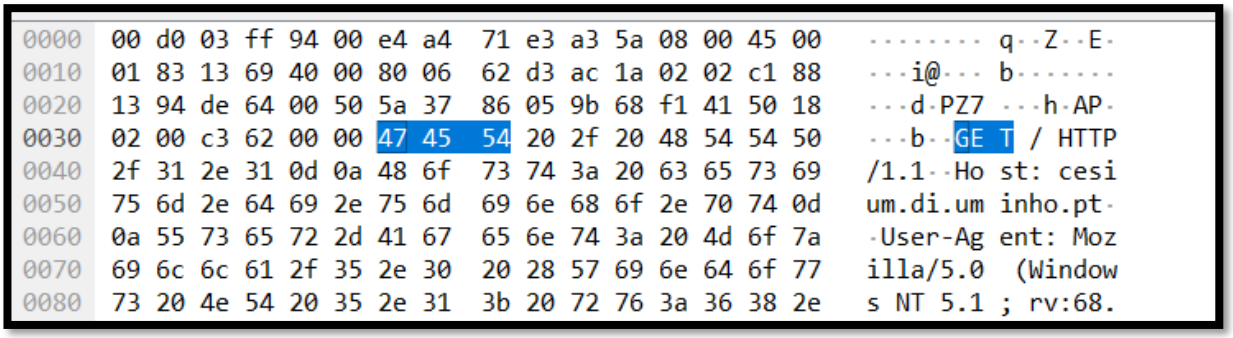
O Valor Hexadecimal é 0x0800 - IPv4, é protocolo de internet que possibilita comunicação entre dispositivos de rede. Como pode ser observado na captura do wireshark:



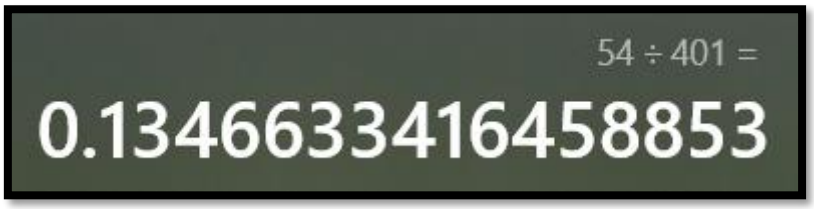
4. **Quantos bytes são usados desde o início da trama até ao caractere ASCII "G" do método HTTP GET?**

São usados 54 bytes.

Captura desde do início da trama observa-se abaixo:



a) **Calcule e indique, em percentagem, a sobrecarga (overhead) introduzida pela pilha** protocolar no envio do HTTP GET.



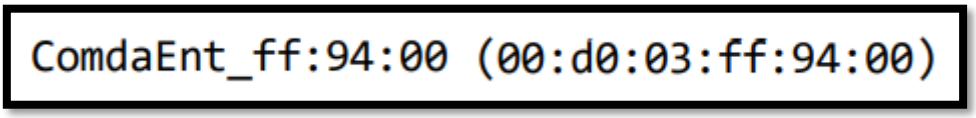
Valor obtido: 13%

b) De acordo com o encapsulamento protocolar acima descrito, e visível no wireshark, como justifica esse overhead?

Overhead: espaço que é usado para informação não útil para o utilizador do host, ou seja, a explicação técnica acerca do pacote.

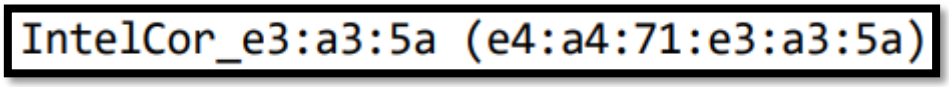
5. **Qual é o endereço Ethernet da fonte? A que sistema de rede corresponde? Justifique.**

O Endereço Ethernet da fonte obtido é o router (Gateway), porque o servidor está numa rede diferente, então primeiro terá de passar pelo router para posteriormente ser reencaminhada para o servidor do https://cesium.di.uminho.pt/, como pode ser observado na imagem capturada::



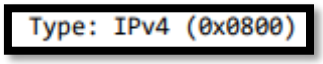
6. **Qual é o endereço MAC do destino? A que sistema corresponde?**

O Endereço MAC do destino obtido obtido é o seguinte, corresponde ao nosso computador que fez o pedido:



7. **Qual é o valor hexadecimal do campo tipo (Type)?**

O Valor Hexadecimal é 0x0800 - IPv4, é protocolo de internet que possibilita comunicação entre dispositivos de rede. Como pode ser observado na captura do wireshark:



8. **Que tipo de resposta foi enviada pelo servidor?**

Foi enviada a resposta do tipo 301, um redirecionamento, provavelmente deveu-se ao facto de ser perguntado por http e o próprio servidor redirecionar para https. Podemos observar essa resposta abaixo:

	Time	Source	Destination	Protocol	Length	Info
446	112.002698	193.136.19.148	172.26.2.2	HTTP	428	HTTP/1.1 301 Moved Permanently (text/html)

TP2: Camada de Ligação Lógica: Ethernet e Protocolo ARP

Protocolo ARP

Abrimos o wireshark e inciamos a captura. Uma vez tudo pronto, entramos apartir do Google Chrome no endereço https://miei.di.uminho.pt/. Os dados obtidos nesta caputra estão presentes neste documento.

9. **Observe o conteúdo da tabela ARP. Diga o que significa cada uma das colunas?**

Mac da interface ativa Internet Adress -- Endereço IP

Physical Address -- Endereço MAC

Type -- Se o endereço é variável ou não (dynamic/static)

Observemos e consideremos a tabela abaixo, obtida através dos passos sugeridos:

```
Interface: 172.26.2.2 --- 0x4
  Internet Address      Physical Address      Type
  172.26.254.254        00-d0-03-ff-94-00     dynamic
  224.0.0.22            01-00-5e-00-00-16     static
  255.255.255.255       ff-ff-ff-ff-ff-ff     static
```

10. **Qual é o valor hexadecimal dos endereços origem e destino na trama Ethernet que contém a mensagem com o pedido ARP (ARP Request)? Como interpreta e justifica o endereço destino usado?**

Valor hexadecimal dos endereço de origem:

```
Source: IntelCor_e3:a3:5a (e4:a4:71:e3:a3:5a)
   Address: IntelCor_e3:a3:5a (e4:a4:71:e3:a3:5a)
```

Valor hexadecimal do endereço de destino:

```
Destination: Broadcast (ff:ff:ff:ff:ff:ff)
   Address: Broadcast (ff:ff:ff:ff:ff:ff)
```

TP2: Camada de Ligação Lógica: Ethernet e Protocolo ARP

O endereço de destino utilizado é o endereço de um Broadcast, é para todas as máquinas existentes na rede de forma a perguntar se alguma delas sabe quem e o ip pretendido.

11. Qual o valor hexadecimal do campo tipo da trama Ethernet? O que indica?

O Valor Hexadecimal é 0x0806 - ARP, é protocolo de internet que possibilita comunicação entre dispositivos de rede. Como pode ser observado na captura do wireshark:

```
Type: ARP (0x0806)
```

12. Qual o valor do campo ARP opcode? O que especifica? Se necessário, consulte a RFC do protocolo ARP http://tools.ietf.org/html/rfc826.html.

O opcode obido é 1, significa que é uma operação de request, ou seja pergunta omo se pode observar abaixo encluindoa tabela informativa obtida:

```
Opcode: request (1)
```

Opcode. 16 bits.

Value	
0	reserved.
1	Request.

http://www.networksorcery.com/enp/protocol/arp.htm

13. A mensagem ARP contém o endereço IP de origem? Que tipo de pergunta é feita?

O endereço IP de origem:

```
Sender IP address: 172.26.2.2
```

A pergunta feita é de quem tem aquele endereço de IP pretendido:

```
Who has 172.26.89.22?
```

14. Localize a mensagem ARP que é a resposta ao pedido ARP efectuado?

DEVIDO A LIMITAÇÕES POR PARTE DA REDE INTERNA DA UNIVERSIDADE DO MINHO, FOI NECESSÁRIO RECORRER A UMA REDE PRIVADA, NO CASO PARTILHA DE REDE DO SMARTPHONE PARA O COMPUTADOR, PERGUNTANDO À REDE SE CONHECE O IP DO GATEWAY (SMARTPHONE).

OBTENDO ASSIM A RESPOSTA PRETENDIDA.

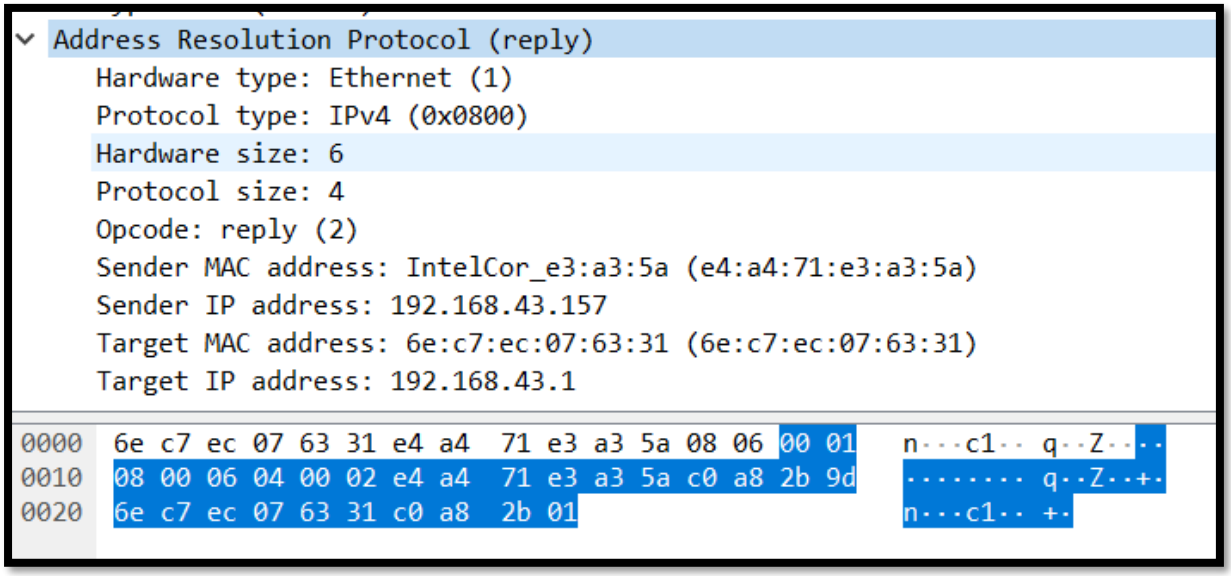
Observemos abaixo a captura:

a) Qual o valor do campo ARP opcode? O que especifica?

O opcode obtido é 2. Indica a resposta.

2	Reply.

b) Em que posição da mensagem ARP está a resposta ao pedido ARP ?



TP2: Camada de Ligação Lógica: Ethernet e Protocolo ARP

15. Quais são os valores hexadecimais para os endereços origem e destino da trama que contém a resposta ARP? Que conclui?

O endereço de destino será quem perguntou, pois a resposta foi enviada em broadcast e quem reconheceu o endereço que foi perguntado irá ser o endereço de origem. Concluímos que funciona como uma pergunta que é direcionada a toda a gente mas responderá somente quem souber a resposta, neste caso se conhece o mac address do ip pretendido.

TP2: Camada de Ligação Lógica: Ethernet e Protocolo ARP

ARP numa topologia CORE

16. Com auxílio do comando ifconfig obtenha os endereços Ethernet das interfaces dos diversos routers.

Endereços obtidos (n2 tem duas interfaces, pois tem duas ligações):

```
root@n1:/tmp/pycore.45396/n1.conf# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:00:00:aa:00:05
          inet addr:10.0.0.1  Bcast:0.0.0.0  Mask:255.255.255.0
          inet6 addr: fe80::200:ff:feaa:5/64 Scope:Link
          inet6 addr: 2001::1/64 Scope:Global
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:36 errors:0 dropped:0 overruns:0 frame:0
          TX packets:9 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:6747 (6.7 KB)  TX bytes:838 (838.0 B)
```

```
root@n2:/tmp/pycore.45396/n2.conf# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:00:00:aa:00:06
          inet addr:10.0.0.2  Bcast:0.0.0.0  Mask:255.255.255.0
          inet6 addr: fe80::200:ff:feaa:6/64 Scope:Link
          inet6 addr: 2001::2/64 Scope:Global
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:93 errors:0 dropped:0 overruns:0 frame:0
          TX packets:49 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:13032 (13.0 KB)  TX bytes:4954 (4.9 KB)

eth1      Link encap:Ethernet  HWaddr 00:00:00:aa:00:03
          inet addr:10.0.1.2  Bcast:0.0.0.0  Mask:255.255.255.0
          inet6 addr: fe80::200:ff:feaa:3/64 Scope:Link
          inet6 addr: 2001:1::1/64 Scope:Global
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:94 errors:0 dropped:0 overruns:0 frame:0
          TX packets:53 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:13178 (13.1 KB)  TX bytes:5298 (5.2 KB)
```

```
root@n3:/tmp/pycore.45396/n3.conf# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:00:00:aa:00:04
          inet addr:10.0.1.1  Bcast:0.0.0.0  Mask:255.255.255.0
          inet6 addr: fe80::200:ff:feaa:4/64 Scope:Link
          inet6 addr: 2001:1::2/64 Scope:Global
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:117 errors:0 dropped:0 overruns:0 frame:0
          TX packets:67 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:15570 (15.5 KB)  TX bytes:6166 (6.1 KB)
```

17. Usando o comando arp obtenha as caches arp dos diversos sistemas

Utilizando a consola interna obteve-se os seguintes resultados, para n1, n2 e n3, as tabelas arp estão vazias:

```
root@n1:/tmp/pycore.45392/n1.conf# arp -v
Entries: 0      Skipped: 0      Found: 0
root@n1:/tmp/pycore.45392/n1.conf#
```

TP2: Camada de Ligação Lógica: Ethernet e Protocolo ARP

18. Faça ping de n1 para n2. Que modificações observa nas caches ARP desses sistemas? Faça ping de n1 para n3. Consulte as caches ARP. Que conclui?

```
root@n1:/tmp/pycore.45388/n1.conf# ping 10.0.1.2
PING 10.0.1.2 (10.0.1.2) 56(84) bytes of data.
64 bytes from 10.0.1.2: icmp_req=1 ttl=64 time=0.030 ms
64 bytes from 10.0.1.2: icmp_req=2 ttl=64 time=0.035 ms
64 bytes from 10.0.1.2: icmp_req=3 ttl=64 time=0.038 ms
64 bytes from 10.0.1.2: icmp_req=4 ttl=64 time=0.038 ms
^C
--- 10.0.1.2 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3028ms
rtt min/avg/max/mdev = 0.030/0.035/0.038/0.005 ms
root@n1:/tmp/pycore.45388/n1.conf#
```

```
root@n1:/tmp/pycore.45388/n1.conf# ping 10.0.1.1
PING 10.0.1.1 (10.0.1.1) 56(84) bytes of data.
64 bytes from 10.0.1.1: icmp_req=1 ttl=63 time=0.062 ms
64 bytes from 10.0.1.1: icmp_req=2 ttl=63 time=0.036 ms
64 bytes from 10.0.1.1: icmp_req=3 ttl=63 time=0.031 ms
^C
--- 10.0.1.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2000ms
rtt min/avg/max/mdev = 0.031/0.043/0.062/0.013 ms
```

Após o ping de n1 para n2, observa-se na tabela de ARP do n1 que consta já em cache o ip de n2.

Após o ping de n1 para n3, conclui-se que na tabela de ARP do n1 que consta já em cache o ip de n2, bem como o HWaddress, pois como não existe ligação direta de n1 para n3, o pacote terá como destino n2 primeiramente.

19. Em n1 remova a entrada correspondente a n2. Coloque uma nova entrada para n2 com endereço Ethernet inexistente. O que acontece?

Observa-se que é perdida a cache de ARP já criada anteriormente.

```
root@n1:/tmp/pycore.45390/n1.conf# arp -v
Entries: 0      Skipped: 0      Found: 0
root@n1:/tmp/pycore.45390/n1.conf# S
```

20. Faça ping de n5 para n6. Sem consultar a tabela ARP anote a entrada que, em sua opinião, é criada na tabela ARP de n5. Verifique, justificando, se a sua interpretação sobre a operação da rede Ethernet e protocolo ARP estava correto.

```
root@n5:/tmp/pycore.45391/n5.conf# ping 10.0.2.21
PING 10.0.2.21 (10.0.2.21) 56(84) bytes of data.
64 bytes from 10.0.2.21: icmp_req=1 ttl=64 time=0.052 ms
64 bytes from 10.0.2.21: icmp_req=2 ttl=64 time=0.024 ms
64 bytes from 10.0.2.21: icmp_req=3 ttl=64 time=0.035 ms
64 bytes from 10.0.2.21: icmp_req=4 ttl=64 time=0.025 ms
64 bytes from 10.0.2.21: icmp_req=5 ttl=64 time=0.033 ms
64 bytes from 10.0.2.21: icmp_req=6 ttl=64 time=0.028 ms
^C
--- 10.0.2.21 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5000ms
rtt min/avg/max/mdev = 0.024/0.032/0.052/0.012 ms
```

É criada a entrada para esse ip (10.0.2.21) em conjunto com o seu endereço físico, mantida em cache ARP.

Com isto podemos observar que no print abaixo consta corretamente aquilo que foi enunciado no parágrafo acima:

```
root@n5:/tmp/pycore.45391/n5.conf# arp
Address                  HWtype  HWaddress           Flags Mask            Iface
10.0.2.21                ether   00:00:00:aa:00:06   C                     eth0
root@n5:/tmp/pycore.45391/n5.conf#
```

TP2: Camada de Ligação Lógica: Ethernet e Protocolo ARP

TP2: Camada de Ligação Lógica: Ethernet e Protocolo ARP (Parte II - Respostas)

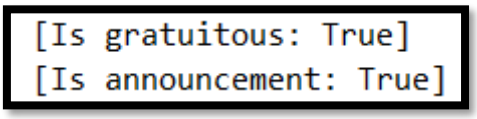
ARP Gratuito

1. **Identifique um pacote de pedido ARP gratuito originado pelo seu sistema. Verifique quantos pacotes ARP gratuito foram enviados e com que intervalo temporal?**

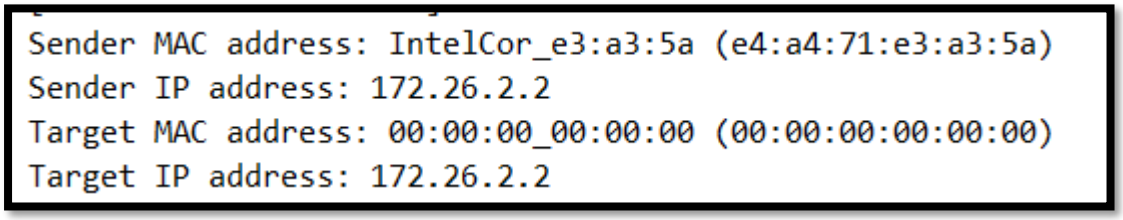
Pedido identificado, no nosso caso somente apareceu um pedido

2. **Analise o conteúdo de um pedido ARP gratuito e identifique em que se distingue dos restantes pedidos ARP. Registe a trama Ethernet correspondente. Qual o resultado esperado face ao pedido ARP gratuito enviado?**

O que distingue dos restantes:



O resultado esperado é o os endereços serem iguais, pois é um announcement da minha própria máquina.



TP2: Camada de Ligação Lógica: Ethernet e Protocolo ARP

Domínios de colisão

1. **Faça ping de n1 para n2. Verifique com a opção tcpdump como flui o tráfego nas diversas interfaces dos vários dispositivos. Que conclui?**

O tráfego flui de n1 para n2, observado através do comando tcpdump.

```
root@n1:/tmp/pycore.45394/n1.conf# ping 10.0.0.10
PING 10.0.0.10 (10.0.0.10) 56(84) bytes of data.
64 bytes from 10.0.0.10: icmp_req=1 ttl=64 time=0.057 ms
64 bytes from 10.0.0.10: icmp_req=2 ttl=64 time=0.081 ms
64 bytes from 10.0.0.10: icmp_req=3 ttl=64 time=0.074 ms
64 bytes from 10.0.0.10: icmp_req=4 ttl=64 time=0.075 ms
64 bytes from 10.0.0.10: icmp_req=5 ttl=64 time=0.079 ms
```

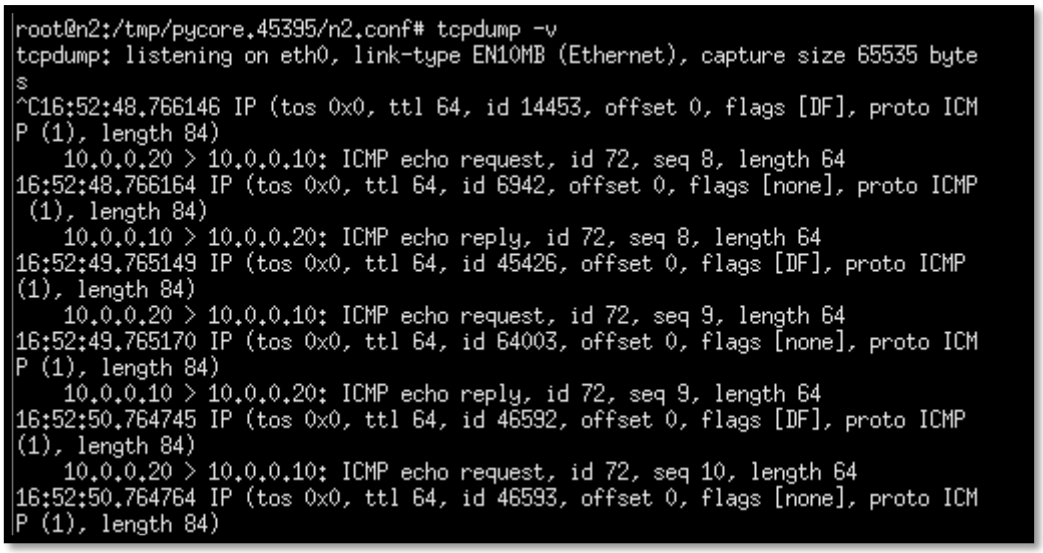
Output do tcp dump em n2:

```
16:48:22.364601 IP 10.0.0.10 > 10.0.0.20: ICMP echo reply, id 240, seq 24, lengt
h 64
16:48:23.365624 IP 10.0.0.20 > 10.0.0.10: ICMP echo request, id 240, seq 25, len
gth 64
16:48:23.365643 IP 10.0.0.10 > 10.0.0.20: ICMP echo reply, id 240, seq 25, lengt
h 64

10 packets captured
10 packets received by filter
0 packets dropped by kernel
```

2. **Na topologia de rede substitua o hub por um switch. Repita os procedimentos que realizou na pergunta anterior. Comente os resultados obtidos quanto à utilização de hubs e switches no contexto de controlar ou dividir domínios de colisão. Documente as suas observações e conclusões com base no tráfego observado/capturado.**

Utilizando o switch, realizando o ping igual a anteriormente, observa-se que o comando tcp dump alberga muito mais informação:



Ao utilizar um switch nota-se um outro tipo de inteligência superior na rede, onde é obtida informação mais completa, podemos concluir isso pois o switch funciona numa camada de rede superior à camada do hub.

TP2: Camada de Ligação Lógica: Ethernet e Protocolo ARP

Conclusão

Os resultados obtidos no desenvolver desta atividade de trabalho prático laboratorial, permitiram ao grupo compreender e consolidar os vários tópicos abordados da tecnologia Ethernet e protoclo ARP (Adress Resolution Protocol). Para tal conclusão, estudou-se (com a ajuda da aplicação WireShark) como devíamos identificar o MAC e o IP tanto da fonte como do destino de uma determinada trama, também utilizamos o CORE network emulator onde realizamos testes de rede virtuais, com situações na ficha de trabalho sugeridas para compreender e estudar melhor estas tecnologias.

Neste mesmo estudo, o grupo apresentou algumas dificuldades em resolver e desenvolver algumas questões relacionadas com a parte I. Na parte II do trabalho, chegamos à conclusão que foi mais fácil a sua resolução uma vez que todos os conhecimentos necessários para a parte I ajudaram, dado que já dominamos o WireShark e o CORE network emulator e todos os conhecimentos do protocolo ARP.

Na nossa opinião, achamos que esta atividade decorreu como previsto, devido a todos estes dados serem coincidentes com todas as afirmações acima referidas, contribuindo para uma coerência entre a teoria e a prática. Sublinhamos, contudo, que foi necessário a ajuda da docente ao longo da realização desta atividade.

No entanto, achamos que conseguimos cumprir todos os objetivos da atividade, sendo esta bem-sucedida, uma vez que conseguimos alcançar o principal objetivo da experiência, que consistia em demonstrar os conhecimentos já absorvidos sobre a tecnologia Ethernet e protoclo ARP (Adress Resolution Protocol).

Relembramos que todos os dados foram calculados de forma rigorosa, por isso pensamos que todos os erros que possam surgir possam ter prevenido da realização de capturas em wireshark em diferentes dias e diferentes lugares.

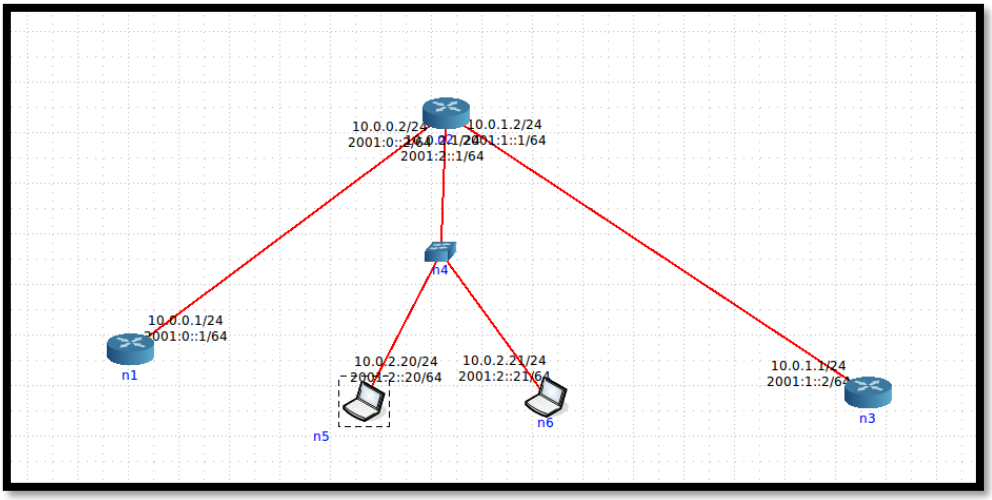
TP2: Camada de Ligação Lógica: Ethernet e Protocolo ARP

Referências

bharnden. (12 de setembro de 2020). *CORE Documentation*. Obtido de GitHub: https://github.com/coreemu/core/blob/master/docs/index.md

networksorcery. (s.d.). *ARP, Address Resolution Protocol*. Obtido de networksorcery: http://www.networksorcery.com/enp/protocol/arp.htm

Organização de Rede #1



Organização de Rede #2