FUNDAÇÃO UNIVERSIDADE FEDERAL DO VALE DO SÃO FRANCISCO
CAMPUS JUAZEIRO
CURSO DE ENGENHARIA DA COMPUTAÇÃO

Breno Gabriel de Souza Coelho

Prática com IP

Juazeiro, BA
2022

Breno Gabriel de Souza Coelho

Prática com IP

Relatório requerido pelo professor Fábio Nelson de Sousa Pereira, como parte das exigências de avaliação da disciplina Redes de Computadores I, do curso de Engenharia da Computação, da Universidade do Vale do São Francisco.

Juazeiro, BA
2022

SUMÁRIO

1 INTRODUÇÃO

O IP (Internet Protocol) é o coração da internet. Responsável por permitir o endereçamento dos diferentes dispositivos e serviços presentes na rede, seu funcionamento é essencial para a internet moderna, tanto que leva um nome tanto quanto grandioso, “Protocolo da Internet”, quase sugerindo que não existem outros. Formalmente estabelecido em 1981 (RFC 791), foi pensado para operar na internet então em desenvolvimento. Contudo, com o aumento da relevância da Web e os avanços que tornaram o acesso a rede cada vez mais onipresente, o protocolo estabelecido foi aos poucos apresentando problemas cada vez mais incômodos, levando ao surgimento do modelo IPv6 (RFC 2460), proposto na década de 90 mas com implementação eficaz ainda se desenvolvendo - afinal mudar o protocolo de operação de uma das camadas mais fundamentais da internet não é uma coisa simples - (BABATUNDE, OLABENJO et al, 2014).

O IPv6 mantém algumas das características do seu antecessor, mas fornece adaptações importantes, excluindo componentes obsoletos ou potencialmente problemáticos (como o mecanismo de fragmentação IP, exclusão da soma de verificação do cabeçalho e utilização de rotulação de fluxos), além de estabelecer endereços IP maiores (o que possivelmente é sua principal revolução), com um total de 128 bits. Uma discussão válida é se esse valor não se tornará obsoleto com o tempo, assim como os 32 bits anteriores (do IPv4); isso é improvável. Uma discussão sobre o assunto pode ser encontrada em STALLINGS, WILLIAM (1996), mas rapidamente é interessante ressaltar que, com essa quantidade de bits, são possíveis 3.4e+38 valores distintos, o que é um número verdadeiramente assombroso.

Para efeito de destaque, considere que todos os dispositivos que possuem um IPv6 único tenham exatamente uma grama de massa. Juntos possuiriam cerca de 3.4e+31 quilogramas, o que é dezenas de milhões de vezes maior que a massa da terra (cerca de 5.97e+24 kg). Naturalmente, é possível que alguns dispositivos tenham mais de um IPv6 devido a algum erro, ou ainda que certos valores supostamente em uso sejam referentes a dispositivos descartados ou não mais sendo utilizados, porém mesmo com essas condições é simplesmente inimaginável que o total de endereços IPv6 sejam totalmente ocupados, garantindo o melhor esforço para que valores não mais em uso voltem a ficar disponíveis, seria necessário uma completa reformulação da forma como a internet funciona e que tipo de dispositivos a acessam para que esses valores chegassem a ser todos usados ao mesmo tempo.

BELLOVIN, STEVEN M. (1989) faz um apanhado geral dos principais problemas de segurança da pilha TCP/IP (modelo Internet), destacando spoofing de números de sequência, ataques de roteamento, spoofing de endereços de origem e ataques de autenticação. Algumas dessas vulnerabilidades são consequência do padrão IPv4, que conta com uma série de mecanismos problemáticos, como fragmentação IP, abrindo margem para certos ataques. Outra ocorrência comum é a

utilização do “man in the middle” na troca de datagramas IP. Esse tipo de problema parece ser favorecido pela limitação de tamanho do IPv4, forçando a utilização de redes NAT e, por sua vez, exigindo o uso de sistemas intermediários para algumas aplicações (como P2P). De certa forma, pode-se dizer que o IPv4 foi pensado para uma internet menor, e “menos complicada”, de modo que a situação moderna expõe problemas não antes seriamente considerados. A mudança para o padrão IPv6, no entanto, deve resolver alguns vários desses problemas.

Esse mesmo artigo propõe alguns métodos de defesa para esses problemas, mas estabelece que certos problemas são inerentes à própria natureza dos protocolos usados. KENT, STEPHEN (1989) fornece uma reflexão adicional a respeito do artigo antes mencionado, discutindo algumas dessas soluções propostas e lançando mão de outra visão sobre os mesmos tópicos.

A transição de IPv4 para IPv6 apresenta vários desafios, como já comentado. RAICU e S. ZEADALLY (2003) tratam de discutir como essa mudança se dará, e quais efeitos práticos já eram observados à época. Os dois principais processos para gerar essa mudança são chamados “6 over 4” (ou pilha IPv4/IPv6) e tunelamento IPv4/IPv6. Na prática ambos andam juntos, e de modo resumido consistem na utilização simultânea de datagramas IPv4 e IPv6, colocando os de formato IPv6 dentro de um datagrama IPv4 quando os sistemas adjacente não forem capazes de suportar a versão mais recente do IP. A ideia é criar as condições necessárias para que o IPv6 seja usado e se popularize, mas sem inutilizar os dispositivos que ainda operam somente em IPv4 (até porque isso geraria uma série de problemas para a internet como um todo).

Num caminho diferente, WANG, SHAOQIANG et al discutem sobre a utilização do Wireshark, sniffer de pacotes, no ensino de TCP/IP em cursos de Redes básicos. À época ainda não tão popular, esse artigo ajudou a popularizar essa aplicação ao mostrar o detalhar o funcionamento do software e apresentar dados empíricos sobre ganhos de aprendizado ao usá-lo em aulas.

2 METODOLOGIA

1. Select the first UDP segment sent by your computer via the traceroute command to gaia.cs.umass.edu. (Hint: this is 44th packet in the trace file in the ipwireshark-trace1-1.pcapng file in footnote 2). Expand the Internet Protocol part of the packet in the packet details window. What is the IP address of your computer?

2. What is the value in the time-to-live (TTL) field in this IPv4 datagram's header?

3. What is the value in the upper layer protocol field in this IPv4 datagram's header? [Note: the answers for Linux/MacOS differ from Windows here].

4. How many bytes are in the IP header?

5. How many bytes are in the payload of the IP datagram? Explain how you determined the number of payload bytes.

6. Has this IP datagram been fragmented? Explain how you determined whether or not the datagram has been fragmented

7. Which fields in the IP datagram always change from one datagram to the next within this series of UDP segments sent by your computer destined to 128.119.245.12, via traceroute? Why?

8. Which fields in this sequence of IP datagrams (containing UDP segments) stay constant? Why?

9. Describe the pattern you see in the values in the Identification field of the IP datagrams being sent by your computer.

10. What is the upper layer protocol specified in the IP datagrams returned from the routers? [Note: the answers for Linux/MacOS differ from Windows here].

11. Are the values in the Identification fields (across the sequence of all of ICMP packets from all of the routers) similar in behavior to your answer to question 9 above?

12. Are the values of the TTL fields similar, across all of ICMP packets from all of the routers

13. Find the first IP datagram containing the first part of the segment sent to 128.119.245.12 sent by your computer via the traceroute command to gaia.cs.umass.edu, after you specified that the traceroute packet length should be

3000. (Hint: This is packet 179 in the ip-wireshark-trace1-1.pcapng trace file in footnote 2. Packets 179, 180, and 181 are three IP datagrams created by fragmenting the first single 3000-byte UDP segment sent to 128.119.145.12). Has that segment been fragmented across more than one IP datagram? (Hint: the answer is yes!)

14. What information in the IP header indicates that this datagram been fragmented?

15. What information in the IP header for this packet indicates whether this is the first fragment versus a latter fragment?

16. How many bytes are there in is this IP datagram (header plus payload)?

17. Now inspect the datagram containing the second fragment of the fragmented UDP segment. What information in the IP header indicates that this is not the first datagram fragment?

18. What fields change in the IP header between the first and second fragment?

19. Now find the IP datagram containing the third fragment of the original UDP segment. What information in the IP header indicates that this is the last fragment of that segment?

20. What is the IPv6 address of the computer making the DNS AAAA request? This is the source address of the 20th packet in the trace. Give the IPv6 source address for this datagram in the exact same form as displayed in the Wireshark window5.

21. What is the IPv6 destination address for this datagram? Give this IPv6 address in the exact same form as displayed in the Wireshark window.

22. What is the value of the flow label for this datagram?

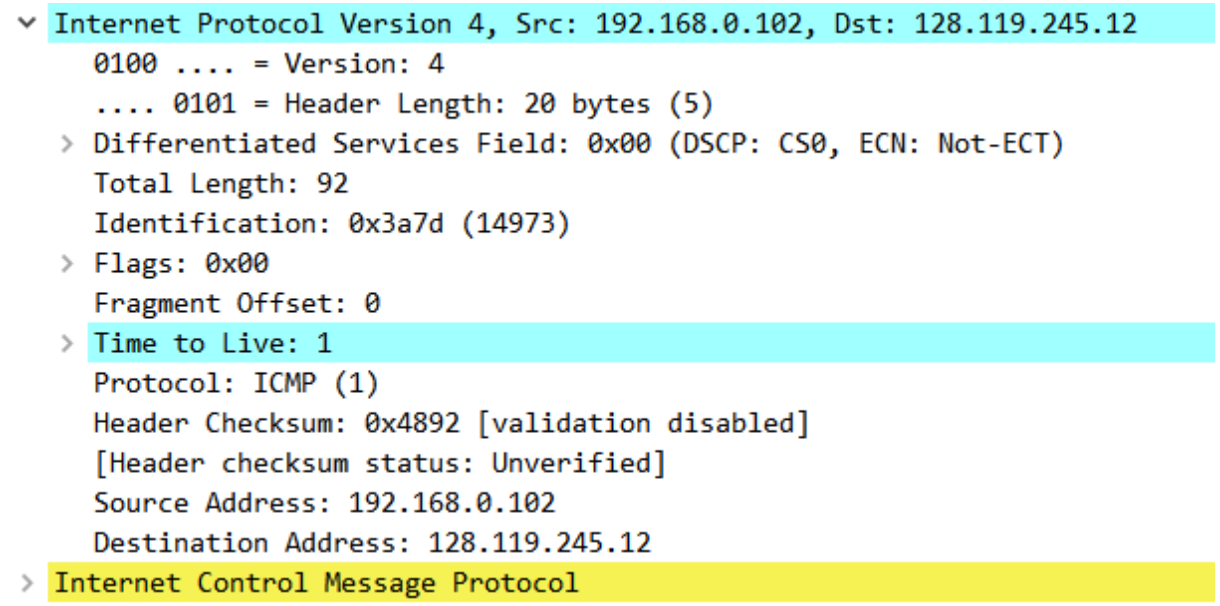
23. How much payload data is carried in this datagram?

24. What is the upper layer protocol to which this datagram's payload will be delivered at the destination?

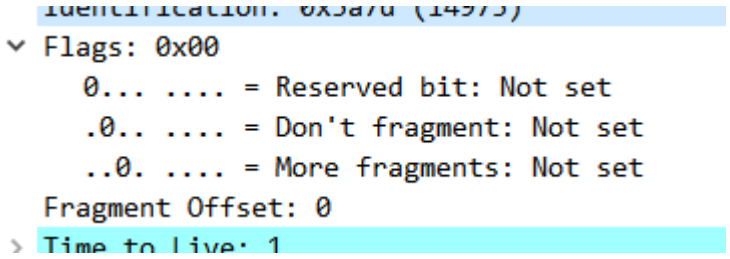
25. How many IPv6 addresses are returned in the response to this AAAA request?

26. What is the first of the IPv6 addresses returned by the DNS for youtube.com (in the ip-wireshark-trace2-1.pcapng trace file, this is also the address that is numerically the smallest)? Give this IPv6 address in the exact same shorthand form as displayed in the Wireshark window

3 DISCUSSÃO



1 - A imagem acima mostra os dados presentes no cabeçalho IP do primeiro pacote ICMP enviado. Pode-se saber que o primeiro pois seu TTL = 1. Analisando os dados, vemos que meu IP é 192.168.0.102.
2 - Como mencionado, TTL = 1.
3 - Pela imagem, ICMP (1)
4 - Também pela primeira linha da imagem, 20 bytes
5 - Pelo cabeçalho, sabemos que o tamanho total do datagrama é de 92 bytes, ao passo que o tamanho do cabeçalho é de 20 bytes. Portanto o tamanho do payload (dados) do datagrama deve valer 92 - 20 = 72 bytes.



6 - O campo flag vale 0, portanto esse é o fragmento final ou um pacote não fragmentado. Como o Fragment Offset (deslocamento) vale 0, sabemos que, se esse fosse um fragmento, estaria na posição 0. Isso é equivalente a dizer que o pacote não foi fragmentado, pois é o primeiro de uma possível fragmentação e ao mesmo tempo o último dela.

	Time	Source	Destination	Protocol	Length	Info
62	7.403552	192.168.0.102	128.119.245.12	ICMP	106	Echo (ping) request id=0x0001, seq=86/22016, tt
63	7.546806	192.168.0.1	192.168.0.102	ICMP	70	Time-to-live exceeded (Time to live exceeded in
64	7.549183	192.168.0.102	128.119.245.12	ICMP	106	Echo (ping) request id=0x0001, seq=87/22272, tt
65	7.558784	192.168.0.1	192.168.0.102	ICMP	70	Time-to-live exceeded (Time to live exceeded in
66	7.560846	192.168.0.102	128.119.245.12	ICMP	106	Echo (ping) request id=0x0001, seq=88/22528, tt
67	7.563364	192.168.0.1	192.168.0.102	ICMP	70	Time-to-live exceeded (Time to live exceeded in
88	13.158801	192.168.0.102	128.119.245.12	ICMP	106	Echo (ping) request id=0x0001, seq=89/22784, tt
89	13.169338	192.168.1.1	192.168.0.102	ICMP	134	Time-to-live exceeded (Time to live exceeded in
90	13.172054	192.168.0.102	128.119.245.12	ICMP	106	Echo (ping) request id=0x0001, seq=90/23040, tt
91	13.174984	192.168.1.1	192.168.0.102	ICMP	134	Time-to-live exceeded (Time to live exceeded in

7 - O campo “identifier” sempre muda. Isso ocorre pois esse campo é utilizado para mostrar a existência de uma fragmentação. Todos os fragmentos possuem o mesmo número de identificação. Como os pacotes enviados não foram fragmentados, todos possuem o mesmo valor neste campo.

8 - A maioria. Os valores de IP de origem e IP de destino, seus tamanhos, flags, tamanho do cabeçalho, etc. A única diferença essencial entre eles é o TTL, e o valor de identificação tem que mudar também por uma questão de reconhecimento. Tudo que não precisa mudar, não é mudado.

9 - Todos são sempre acrescidos em uma unidade. (são valores em hexadecimal. Pegando alguns dos primeiros, vê-se: 0x3a7d, 0x3a7e, 0x3a7f, 0x3a80, …)

No.	Time	Source	Destination	Protocol	Length	Info
63	7.546806	192.168.0.1	192.168.0.102	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
65	7.558784	192.168.0.1	192.168.0.102	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
67	7.563364	192.168.0.1	192.168.0.102	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
73	7.609219	192.168.0.1	192.168.0.102	ICMP	70	Destination unreachable (Port unreachable)
81	9.180480	192.168.0.1	192.168.0.102	ICMP	70	Destination unreachable (Port unreachable)
84	10.646169	192.168.0.1	192.168.0.102	ICMP	70	Destination unreachable (Port unreachable)
89	13.169338	192.168.1.1	192.168.0.102	ICMP	134	Time-to-live exceeded (Time to live exceeded in transit)
91	13.174984	192.168.1.1	192.168.0.102	ICMP	134	Time-to-live exceeded (Time to live exceeded in transit)
93	13.180622	192.168.1.1	192.168.0.102	ICMP	134	Time-to-live exceeded (Time to live exceeded in transit)
97	13.194890	192.168.1.1	192.168.0.102	ICMP	120	Destination unreachable (Port unreachable)

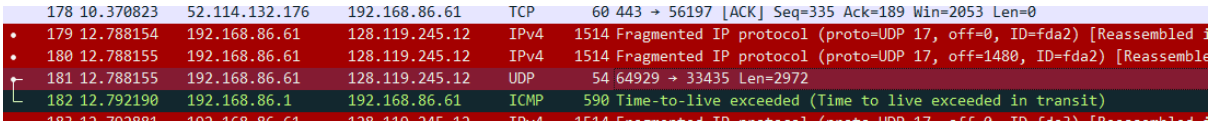
```
> Frame 63: 70 bytes on wire (560 bits), 70 bytes captured (560 bits) on interface \Device\NPF_{B12FAD63-6368-4A7C-8E22-719C0677CBA6}, i
> Ethernet II, Src: TendaTec_86:e6:00 (cc:2d:21:86:e6:00), Dst: Palladiu_da:b3:3f (5c:c9:d3:da:b3:3f)
v Internet Protocol Version 4, Src: 192.168.0.1, Dst: 192.168.0.102
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 56
    Identification: 0x3040 (12352)
  > Flags: 0x00
    Fragment Offset: 0
    Time to Live: 64
    Protocol: ICMP (1)
    Header Checksum: 0xc8cd [validation disabled]
    [Header checksum status: Unverified]
```

10 - O mesmo de antes, ICMP (1)

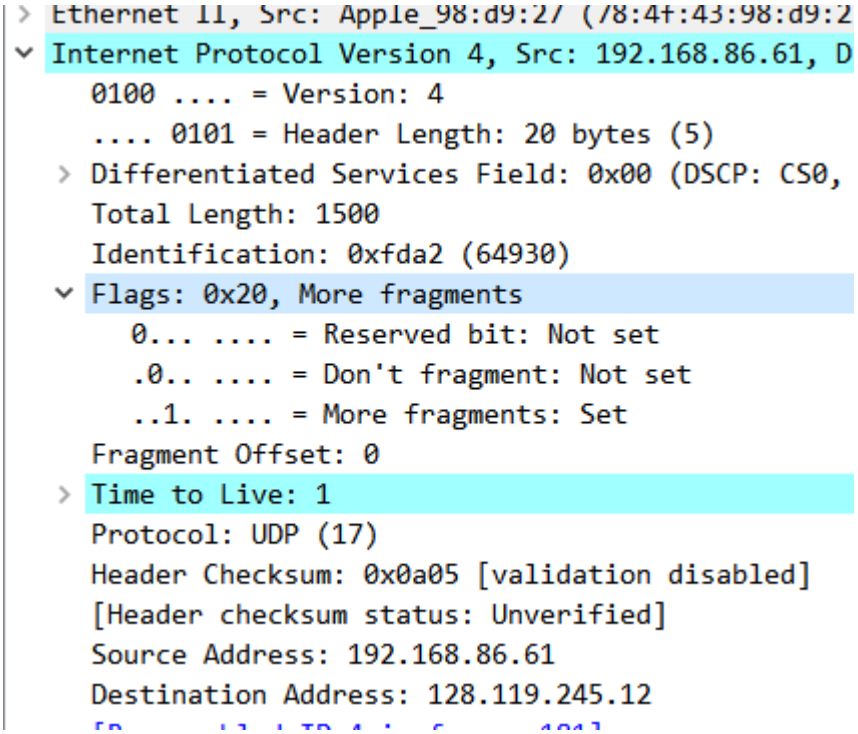
11 - Não. De três em três, o padrão de aumentar o valor de um em um se mantém, porém muda repentinamente. Isso certamente é uma consequência dos pacotes serem enviados três de uma única vez, e todos sempre “morrem” num mesmo roteador. Esse deve portanto enviar três mensagens de erro seguidas, o que explica os três valores seguidos (será que se algum outro erro num pacote diferente ocorresse no meio tempo, esses valores poderiam ter alguma leve discrepância? Talvez, mas deve ser bem improvável de ocorrer)

De toda forma, é natural que os valores de identificação dos Echo Reply's dos roteadores subjacentes sejam totalmente diferentes.

12 - Um pouco. Os primeiros ficavam em torno de 64, o que é resultado desses pacotes terem sido gerados por sistemas Linux (que coloca TTL inicial nesse valor, por padrão). Os mais "distantes" chegavam em torno de 250, que é resultado de serem criados por sistemas Unix possivelmente, que coloca o valor de TTL inicial como 255. As pequenas discrepâncias são resultado dos repasses aos quais os próprios pacotes de Echo Reply são submetidos pelos roteadores no envio até mim, que diminui parte de seus valores TTL iniciais.



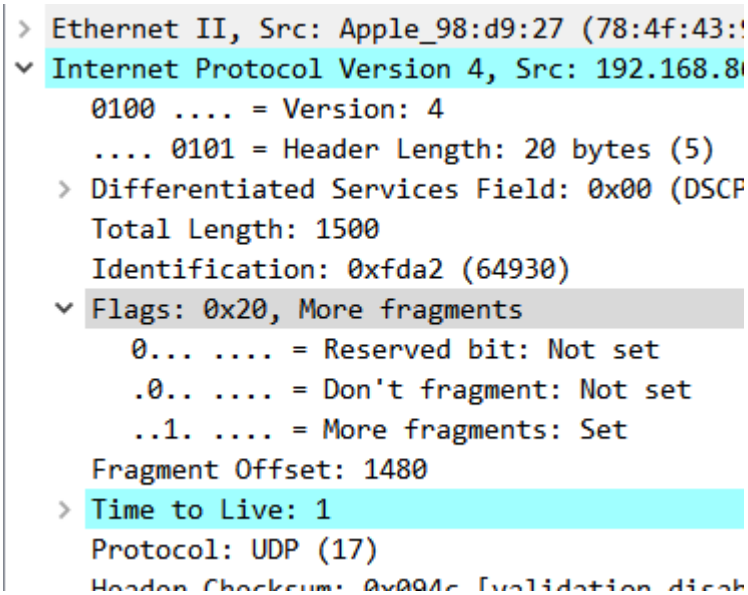
13 - Sim. Pelo comportamento de espera por resposta antes percebido, podemos afirmar que ele deve ter sido dividido em 3 partes. Somente a última foi entendida pelo Wireshark como UDP, as demais eram puramente IPv4.



14 - O bit flag relativo a se existem ou não mais fragmentos por vir está com valor 1.

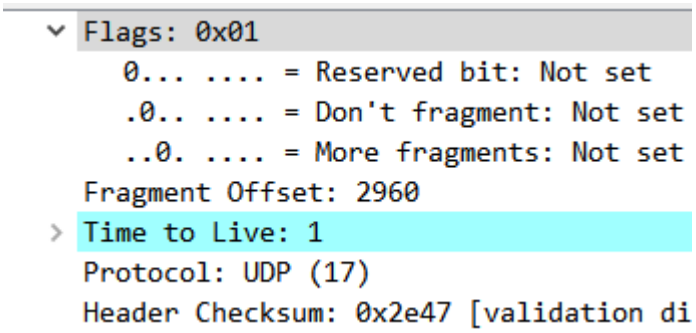
15 - O campo Fragment Offset (deslocamento) informa a posição do fragmento na composição do arquivo original. Como ele vale 0, sabemos que é o primeiro fragmento.

16 - Pelo campo Total Length, são 1500 Bytes. Os dois primeiros fragmentos possuem esse mesmo tamanho inclusive.

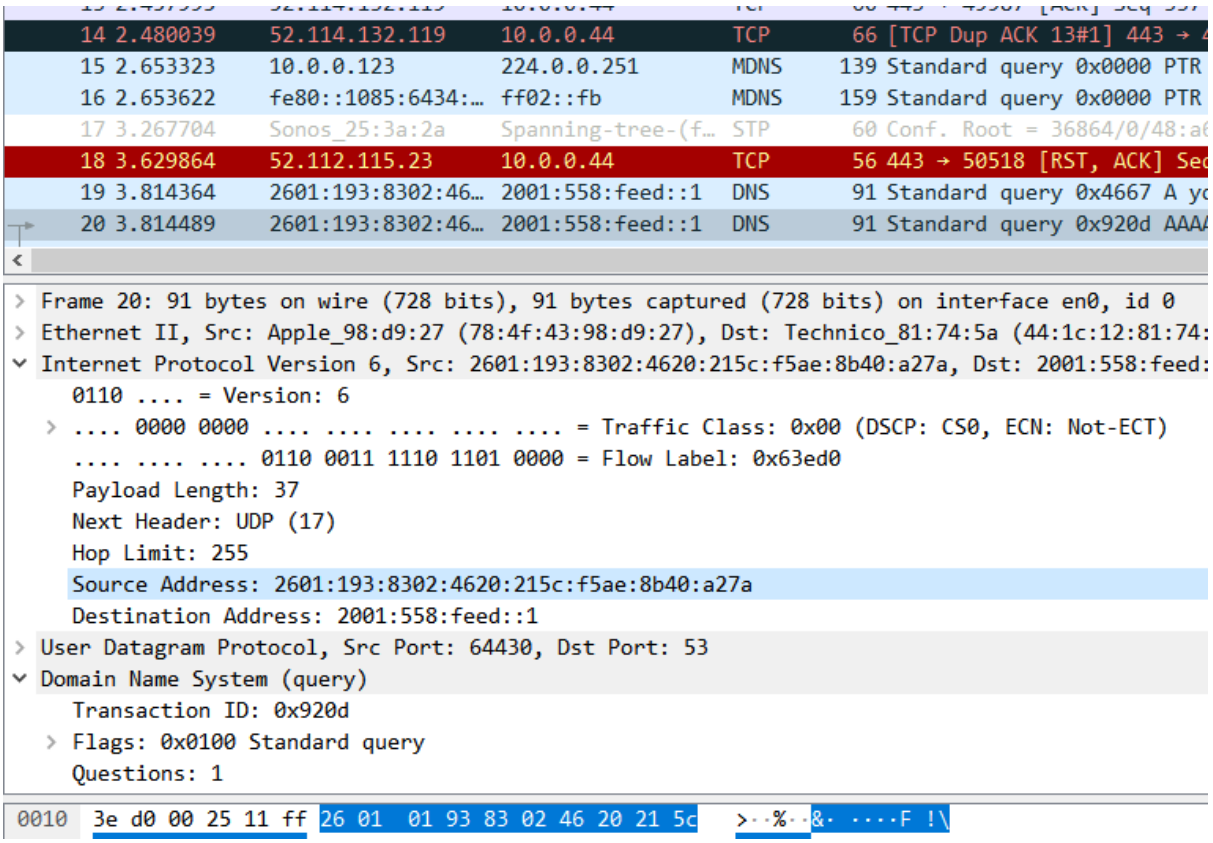


17 - O Fragment Offset agora vale 1480. Isso significa que o primeiro byte do campo de dados carregado é de posição 1480 no arquivo original. Faz sentido, com o cabeçalho IP ocupando 20 bytes, e o primeiro fragmento tendo 1500 bytes ao todo, sobram 1480 bytes para o cabeçalho.

18 - O já mencionado Offset, e o campo com a soma de verificação do cabeçalho (o que faz sentido, já que a informação no cabeçalho foi modificada)



19 - A flag relativa a existência de mais fragmentos agora vale 0. Como o Offset não é nulo, sabemos que esse é o último fragmento de um envio IP fragmentado.



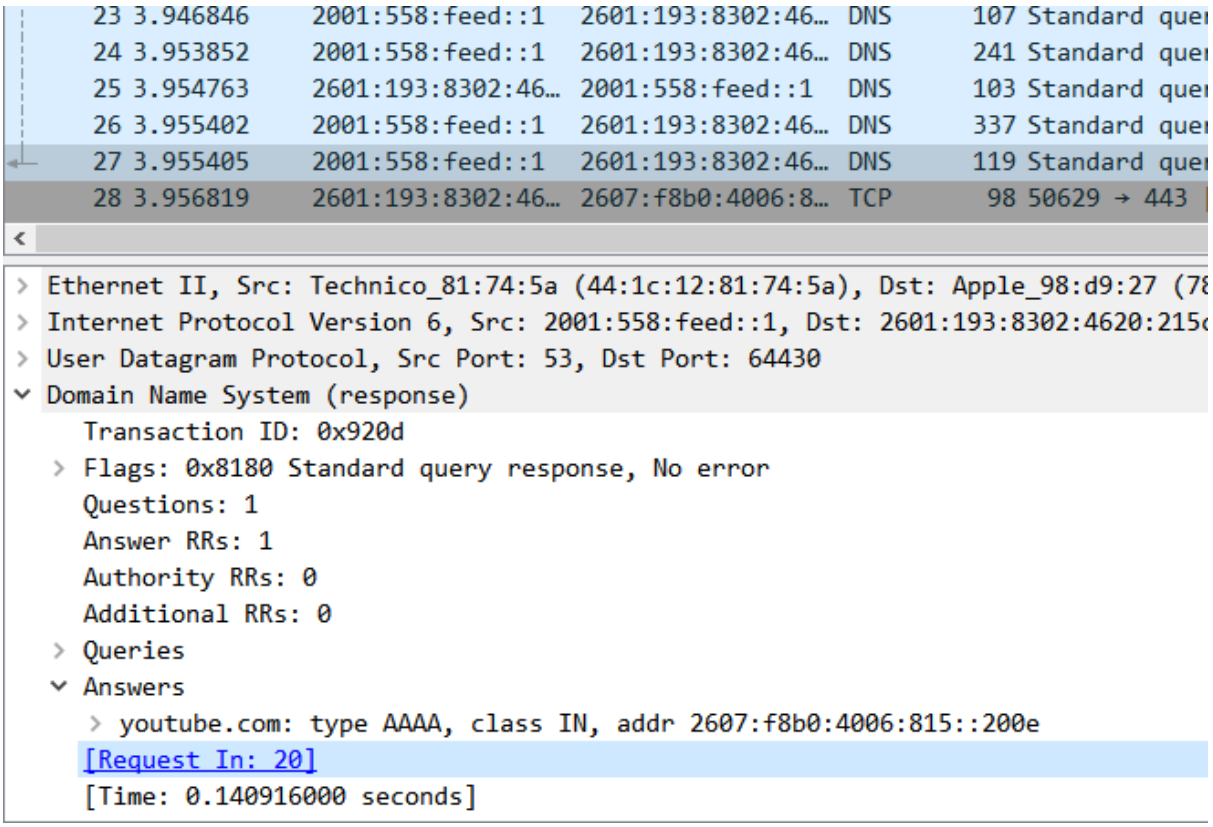
20 - Abrindo os detalhes do datagrama IPv6, vemos que o endereço IP é 2601:193:8302:4620:215c:f5ae:8b40:a27a

21 - Copiando exatamente como está, 2001:558:feed::1

22 - Observando nas primeiras linhas do cabeçalho, ele vale 0x00063ed0 (obtive isso copiando direto do aplicativo. Aparentemente ele oculta os zeros à esquerda)

23 - Pelo campo payload length, são 37 bytes.

24 - Analisando o campo Next Header, temos que o protocolo de nível superior usado é o UDP (17)



25 - Somente um valor de IPv6 foi retornado.

26 - Como só foi retornado um, ele é o 2607:f8b0:4006:815::200e

4 CONCLUSÃO

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

5 REFERÊNCIA BIBLIOGRÁFICA

WANG, SHAOQIANG; XU, DONGSHENG; YAN, SHILIANG. **Analysis and application of Wireshark in TCP/IP protocol teaching.** In: 2010 International Conference on E-Health Networking Digital Ecosystems and Technologies (EDT). p. 269-272. (IEEE, 2010)

BELLOVIN, STEVEN M. **Security problems in the TCP/IP protocol suite**. ACM SIGCOMM Computer Communication Review, v. 19, n. 2, p. 32-48 (1989)

KENT, STEPHEN. **Comments on "Security problems in the TCP/IP protocol suite".** ACM SIGCOMM Computer Communication Review, v. 19, n. 3, p. 10-19 (1989)

BABATUNDE, OLABENJO; AL-DEBAGY, OMAR. **A comparative review of internet protocol version 4 (ipv4) and internet protocol version 6 (ipv6)**. arXiv preprint arXiv:1407.2717 (2014)

STALLINGS, WILLIAM. **IPv6: the new Internet protocol.** IEEE Communications Magazine, v. 34, n. 7, p. 96-108 (1996)

RAICU, IOAN; ZEADALLY, SHERALI. **Evaluating IPv4 to IPv6 transition mechanisms**. In: 10th International Conference on Telecommunications, 2003. ICT 2003. p. 1091-1098. (2003)