

”

E-fólio B | Instruções para a realização do E-fólio



SISTEMAS EM REDE | 21106

A preencher pelo estudante

UNIDADE CURRICULAR: Sistemas em Rede

CÓDIGO: 21106

DOCENTE: Professor Arnaldo Santos

NOME: Francisco José Pinto de Amaral

N.º DE ESTUDANTE: 1802876

CURSO: Licenciatura em Engenharia Informática

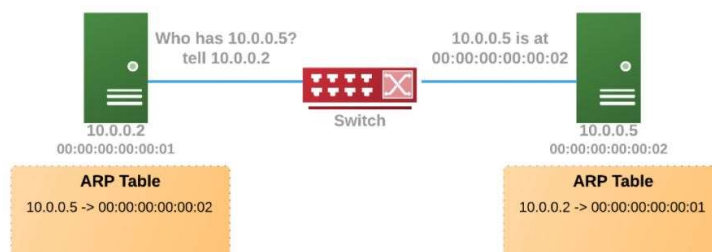
DATA DE ENTREGA: 29 – 12 – 2021

TRABALHO / RESOLUÇÃO:

Questão nº 1 - Indique o que é, e como funciona, o protocolo **ARP**.

R.: **ARP** significa **Adress Resolution Protocol**, em português: Protocolo de Resolução de Endereços, é um padrão de Redes de Computadores publicado por David C. Plummer em 1982 pela **RFC 826**. Este protocolo/procedimento permite mapear/rastrear um IP dinâmico através de um endereço de máquina física denominado por **MAC – Media Access Control**.

A função do ARP é essencialmente traduzir endereços, sendo que os endereços de IPv4 são de 32 bits e os endereços MAC de 48 bits. As Placas de Interface Rede ou **NIC – Network Interface Card**, Ethernet, na sua fabricação, é solicitado um endereço ao **IEEE – Institute of Electrical and Electronics Engineers** de forma a garantir que não existam na mesma rede dois endereços iguais. Cada placa ou equipamento, capaz de lidar com pacotes IPv4, possuem uma tabela ARP, que permite armazenar os endereços IP e MAC dos dispositivos na rede em forma de *cache*, caso os endereços ainda não existam listados o protocolo modifica a tabela com os novos dados. Para verificar a corrente tabela ARP é usado o comando *arp -a* na linha de comandos (SO Windows).



Para exemplificar o método de funcionamento do protocolo ARP temos a imagem acima: o computador da esquerda (emissor) envia uma pergunta (ARP Request) para a rede em forma de *Broadcast* a questionar qual o MAC do computador com o IP 10.0.0.5, o computador da direita (com o IP em questão) responde ao emissor (ARP Reply) e ambas as máquinas guardam a informação na sua tabela ARP.

De acordo com o modelo **OSI – Open Systems Interconnection**, o protocolo ARP funciona entre as camadas 2 e 3, Enlace de dados e Rede, isto porque o endereço MAC está na camada 2 e o endereço IPv4 na camada 3.

Existem vários tipos de protocolos ARP, dependendo da sua utilização sendo os mesmos: ARP Probe, INARP, ARP Spoofing/ARP Poisoning, Gratuitous ARP, Proxy ARP.

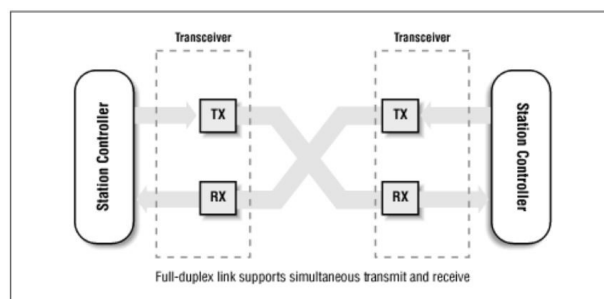
Questão nº 2 - Se os retardos forem registados como números de **8 bits** numa rede com **90 roteadores** e os vetores de retardo forem trocados **duas vezes por segundo**, qual será a largura de banda por linha (**full-duplex**) ocupada pelo algoritmo de roteamento distribuído? Assuma, como princípio, que cada roteador tem 4 linhas para outros roteadores.

R.: Havendo 90 roteadores na rede, que por sua vez ocupam a mesma quantidade de posições (QP), e cada uma ocupa 8 bits (TR), para calcular o tamanho do vetor (TV) será o múltiplo das posições pelo número de bits, ou seja:

$$TV = QP \times TR = 90 \times 8 = 720 \text{ bits}$$

Estando perante uma troca de vetores **duas vezes** por segundo teremos que multiplicar o tamanho do vetor por **2**, dado o valor final ser por segundo:

$$TV \times 2 = 720 \times 2 = 1440 \text{ bits/s ou } 1.44 \text{ kilobits/s}$$

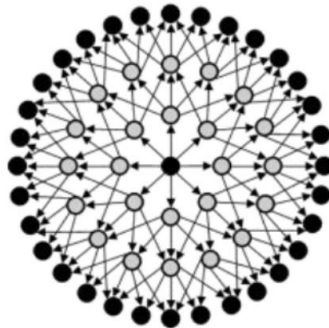


A largura de banda por linha, **full-duplex**, será a mesma que acima calculada (1.44kb/s) pois a mesma suporta a transmissão e receção de dados em simultâneo.

Questão nº 3 - O algoritmo de **inundação (flooding)** é classificado como um algoritmo estático de roteamento. Explique, sucintamente, como funciona.

R.: O algoritmo de inundação (*flooding*) é um algoritmo estático que permite o envio de cada pacote de entrada para todas as linhas de saída exceto aquela por onde chegou. De forma a garantir que o algoritmo chega ao destino é gerada uma grande quantidade de pacotes duplicados, tendo vários caminhos possíveis em paralelo, o *flooding* escolhe sempre o caminho mais curto.

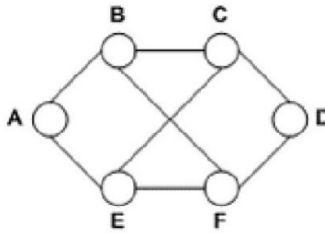
Na imagem, um pacote chega ao nó central *k* e envia o pacote para todos os nós ligados a *k*, e assim sucessivamente, caso haja uma falha em um dos nós intermediários o *flooding* encontrará uma rota alternativa para garantir que o pacote chega aos nós extremos.



Claramente, ao gerar um número infinito de pacotes que poderão ocasionar o efeito de inundação nos nós e canais de comunicações (largura de banda desperdiçada). Para evitar este efeito, caso se saiba o caminho desde a origem até ao destino, é colocar um contador **TTL – Time To Leave** de hops contido no cabeçalho de cada pacote, o contador é decrementado em cada hop, quando o TTL atinge o zero então o pacote é retirado da rede. Caso não haja possibilidade de dimensionar o caminho desde a origem até ao destino poderá iniciar o contador no pior cenário de modo a cobrir toda a rede/sub-rede. Uma outra visão, caso se conheça a rede, é através do *flooding seletivo* no qual o pacote é enviado apenas na direção do destino, evitando saídas para nós vizinhos.

Uma das vantagens do uso deste algoritmo é a sua facilidade de implementação e uma das desvantagens é a largura de banda desperdiçada que em alguns casos é necessário o uso de mecanismos para minimizar o impacto negativo no desempenho da rede.

Questão nº 4 - Considere a seguinte rede de roteadores:



Nesta rede, é utilizado o **roteamento com vetor de distância** (*"Distance-Vector Routing"*) e os vetores a seguir indicados acabaram de entrar no **roteador C**: - de B: (5, 0, 8, 12, 6, 2); - de D: (16, 12, 6, 0, 9, 10); - de E: (7, 6, 3, 9, 0, 4)
Os retardos medidos para B, D e E são 6, 3 e 5, respetivamente.
Explique o funcionamento do algoritmo e indique qual é a nova tabela de **roteamento de C**. Forneça a **linha de saída** a ser usada e o retardo esperado.

R.: O roteamento com vetor de distância ou em inglês *"Distance Vector Routing"*, foi o algoritmo de roteamento original da ARPNET. É conhecido por diversos nomes, sendo o mais comum o algoritmo de roteamento distribuído de Bellman-Ford. O algoritmo funciona de forma a calcular o caminho mais curto para o tráfego de pacotes até ao seu destino passando por todos os nós na rede, em cada salto é criada/atualizada uma tabela na memória de cada roteador onde contém a informação dos custos do caminho calculados através do número de saídas e tamanho do vetor e outros que influenciem o tráfego da rota. Em termos comparativos, para entendimento deste processo, temos o exemplo do sistema de navegação por GPS quando estamos em um local (origem) e pretendemos ir para um destino pelo caminho mais rápido calculando todas as possíveis estradas que irão dar ao destino considerando tráfego, outros, etc.

Vetores a entrar no roteador C:

- de B: (5, 0, 8, 12, 6, 2);

- de D: (16, 12, 6, 0, 9, 10);

- de E: (7, 6, 3, 9, 0, 4);

Retardos medidos:

B → 6 D → 3 E → 5

Calcula-se o retardo esperado que é a soma do vetor mais a retardo vindo de cada roteador:

ROTEADORES	VETORES B	VETORES + RETARDO B	VETORES D	VETORES + RETARDO D	VETORES E	VETORES + RETARDO E
A	5	5+6 = 11	16	16+3 = 19	7	7+5 = 12
B	0	0+6 = 6	12	12+3 = 15	6	6+5 = 11
C	8	8+6 = 14	6	6+3 = 9	3	3+5 = 8
D	12	12+6 = 18	0	0+3 = 3	9	9+5 = 14
E	6	6+6 = 12	9	9+3 = 12	0	0+5 = 5
F	2	2+6 = 8	10	10+3 = 13	4	4+5 = 9

Em cada linha/roteador verificamos qual o caminho mais curto (preenchimento a verde) e desta forma obtemos o retardo esperado e linhas de saída:

ROTEADOR	RETARDO	SAÍDA
A	11	B
B	6	B
C	-	-
D	3	D
E	5	E
F	8	B

De acordo com o calculo/tabela o vetor transmitido é C: (11, 6, -, 3, 5, 8).

Referências bibliográficas:

Livro da Bibliografia da UC: Tanenbaum, A. S., Wetherall, D.J. (2014). Computer Networks. 5th Edition, Pearson New International Edition, USA

<https://www.section.io/engineering-education/address-resolution-protocol/>

<https://blog.pantuza.com/artigos/o-protocolo-arp-address-resolution-protocol>

<https://www.oreilly.com/library/view/ethernet-the-definitive/1565926609/ch04.html>

<https://www.nvtpybridge.com/full-duplex/>

<https://www.geeksforgeeks.org/fixed-and-flooding-routing-algorithms/>

<https://www.sciencedirect.com/science/article/pii/S1570866713000221>

<https://www.youtube.com/watch?v=8SQLVMJJ5T4>

FIM