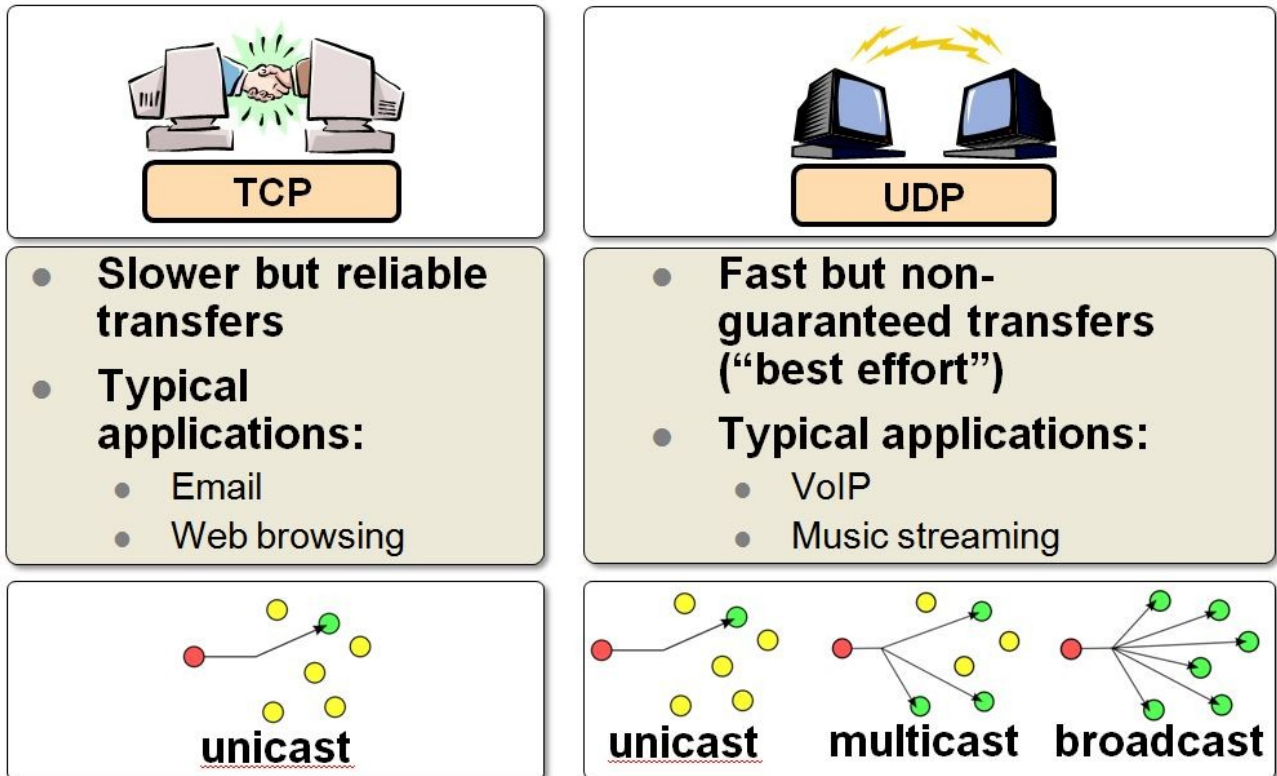


Redes de Computadores

Prof. Robson de Souza

Aulas 19 e 20

Conteúdo: Camada de Transporte: UDP (User Datagram Protocol).



<http://microchipdeveloper.com/tcpip:tcp-vs-udp>

Os protocolos da camada de transporte fornecem comunicação lógica entre os **processos** de aplicação em diferentes hosts.

O lado emissor quebra as mensagens da aplicação em segmentos e envia para a camada de rede. O lado receptor remonta os segmentos em mensagens e passa para a camada de aplicação.

Na Internet podem ser utilizados os protocolos TCP e o UDP. Existem mais de um protocolo de transporte disponíveis para as aplicações.

Camada de Rede → Fornece comunicação lógica entre os hosts.

Camada de transporte → Fornece comunicação lógica entre os processos.

Se fizermos uma analogia com um grupo de crianças que desejam enviar cartas umas para as outras em diversas casas, é como se:

processos → crianças.

Mensagens da aplicação → cartas.

Hosts → Casas

Protocolo de Transporte → O nome do destinatário.

Protocolo de Rede → O serviço postal.

O protocolo TCP garante ordem de entrega, faz controle de congestionamento, faz controle de fluxo e é orientado à conexão.

O protocolo UDP é considerado não confiável e não garante ordem de entrega.

UDP (User Datagram Protocol)

Esse protocolo é um protocolo de transporte da Internet “sem frescuras”. Os segmentos UDP podem ser perdidos ou entregues fora de ordem para a aplicação.

O UDP é um protocolo sem conexão, ou seja, não há apresentação entre o UDP transmissor e o receptor. Cada segmento UDP é tratado de forma independente dos outros.

As vantagens do UDP é que, como não há estabelecimento de conexão, os atrasos que poderiam resultar dessa tarefa não existem. Outra vantagem é que o cabeçalho de segmento é reduzido. Como não há controle de congestionamento, o UDP pode enviar segmentos tão rápido quanto desejado (e possível).

Esse protocolo é muito usado para aplicações de multimídia contínua (streaming), que são tolerantes à perda.

Para obter transferência confiável sobre UDP é necessário acrescentar confiabilidade na camada de Aplicação. A recuperação de erros é específica de cada aplicação.

Segmento UDP:

Source Port (16 bits)	Destination Port (16 bits)
Length (16 bits)	Checksum (16 bits)
Data....	

<http://faculty.petra.ac.id/resmana/private/tcpip/tyt04fi.htm>

O UDP possui o UDP Checksum, que tem por objetivo detectar erros no segmento transmitido. O transmissor trata o conteúdo de segmento como sequência de inteiros de 16 bits, em seguida, realiza uma soma do conteúdo do segmento e faz o complemento de 1 da soma

O transmissor coloca o valor do checksum no campo de checksum do UDP.

O receptor computa o checksum do segmento recebido e verifica se o checksum calculado é igual ao valor do campo checksum.

Se não for igual, significa que um erro foi detectado. Se for igual significa que não há erros nesses dados, mas ainda assim, podem existir outros erros.

Exemplo:

$$\begin{array}{r}
 1110011001100110 \rightarrow \text{Sequência 1} \\
 + 1101010101010101 \rightarrow \text{Sequência 2} \\
 \hline
 11011101110111011
 \end{array}$$

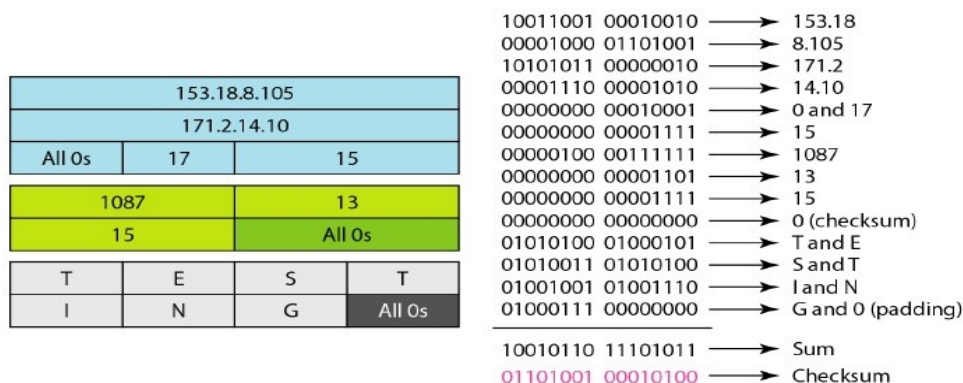
Nesse caso, a soma teve um bit a mais (17 bits ao todo), nesse caso, esse bit é chamado de **wraparound** e deve ser somado ao resultado antes de fazer o complemento de 1:

$$\begin{array}{r}
 1011101110111011 \\
 + 1 \\
 \hline
 1011101110111100
 \end{array}$$

Em seguida faz-se o complemento de 1 desse resultado. O valor final é o checksum:

$$\begin{array}{r}
 1011101110111100 \\
 \text{Complemento de 1} \rightarrow \\
 \hline
 0100010001000011 \rightarrow \text{Checksum}
 \end{array}$$

Figure 23.11 Checksum calculation of a simple UDP user datagram



23.20

<https://www.slideshare.net/WayneJonesJnr/ch23-3361680>

Atividades

Observe se os seguintes segmentos chegaram corretamente ou com erros:

1)

Sequência 1 → 1100101111001010

Sequência 2 → 0010100010100111

Checksum → 0000101100001110

2)

Sequência 1 → 1000101101110010

Sequência 2 → 0000110010111000

Checksum → 0110011111010101

3)

Sequência 1 → 1100110011001001

Sequência 2 → 1000011000001101

Checksum → 1010110100101001

4)

Sequência 1 → 1100001100110110

Sequência 2 → 1001110110000000

Checksum → 1001111101001000

Referências bibliográficas:

TANENBAUM, Andrew. S. Redes de Computadores. São Paulo: *Pearson*, 5ª Ed. 2011.

KUROSE, James F.; ROSS, Keith W. Redes de Computadores e a Internet – Uma Abordagem Top-Down. São Paulo: *Pearson*, 6ª Ed. 2013.