

# Introdução as Redes de Comunicação - Aula 4

Prof Leandro Dihl

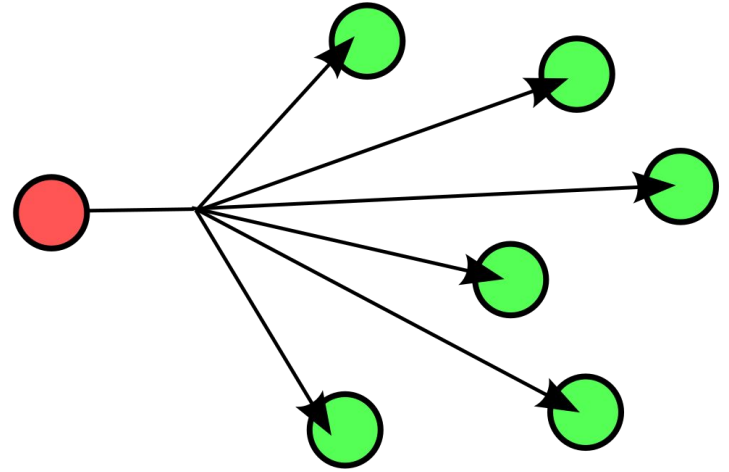
# Sumário

- 1) Exercício 7 - Broadcasting.
- 2) Exercício 8 - Timeout.

# Exercício 7 - BROADCAST

A UDP broadcast é uma técnica que permite o envio de datagrama UDP de uma única fonte para todos os computadores em uma LAN. Para enviar um datagrama UDP endereçado a todos os computadores na rede local, ele precisa ser enviado para um endereço especial chamado endereço de broadcast.

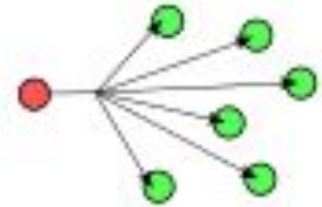
O endereço de broadcast para uma LAN é o endereço mais alto da sub-rede local ou o endereço de broadcast universal: 255.255.255.255. Para receber mensagens de broadcast, o receptor precisa habilitar a opção `SO_BROADCAST` no socket.



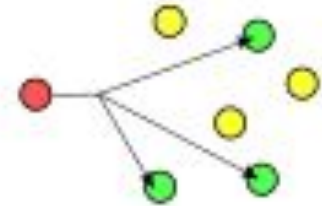
# Exercício 7 - BROADCAST vs MULTICAST

Multicast é a transmissão de informação para múltiplos destinatários simultaneamente, usando a estratégia mais eficiente, onde as mensagens só passam por um link uma única vez e somente são duplicadas quando o link para os destinatários se divide em duas direções.

- **broadcast**



- **multicast**



# setsockopt function

`setsockopt(m_socket,SOL_SOCKET,SO_BROADCAST,(char *) &optval,optlen);`

<https://docs.microsoft.com/en-us/windows/win32/api/winsock/nf-winsock-setsockopt>

## Syntax

C++

```
int setsockopt(  
    SOCKET      s,  
    int         level,  
    int         optname,  
    const char *optval,  
    int         optlen  
);
```

# setsockopt function: Parameters

`s`

A descriptor that identifies a socket.

`level`

The level at which the option is defined (for example, SOL\_SOCKET).

`optname`

The socket option for which the value is to be set (for example, SO\_BROADCAST). The *optname* parameter must be a socket option defined within the specified *level*, or behavior is undefined.

`optval`

A pointer to the buffer in which the value for the requested option is specified.

`optlen`

The size, in bytes, of the buffer pointed to by the *optval* parameter.

# Exemplo servidor:

Define variáveis

Inicia Winsock

Cria o socket

Associa socket ao  
endereço de escuta

Loop (recebe msgs)

# Exemplo cliente:

Define variáveis

Inicia Winsock

Cria o socket

Ativa possibilidade  
de broadcast

Preenche end serv

Envia msg servidor

`int opt = 1;`

`setsockopt(sockfd, SOL_SOCKET, SO_BROADCAST, (char *)&opt, sizeof(opt));`

Obtém o porto

Aguarda resposta

Fecha o socket



## Exercício 8 - Timeout.

O cliente vai aguardar a resposta do servidor somente por um determinado período de tempo.

Também uso da função:

```
setsockopt(m_socket, SOL_SOCKET, SO_RCVTIMEO, (char *)&time,  
sizeof(time));
```

# Exemplo:

Define variáveis

Inicia Winsock

Cria o socket

Config. o socket  
para timeout da  
recepção

`int time= 10000;//msec`

`setsockopt(sockfd, SOL_SOCKET, SO_RCVTIMEO, (char *)&time, sizeof(time));`

Preenche end serv

Envia msg servidor

Obtém o porto

Aguarda resposta

Fecha o socket

# Outras formas de criar timeout

- Linux:
  - `errno=EAGAIN` : Try Again
  - `select()`
  - `SIGALRM`
    - The `ualarm()` function causes the system to generate a `SIGALRM` signal after the number of real-time microseconds (specified by `useconds`) has elapsed.
  - `alarm()`
  - `signal()`