

# Laboratório: Comunicação em Canal Não Confiável

---

**Disciplina:** Redes de Computadores II

**Professora:** Angelita Rettore de Araujo

## Introdução: A Realidade dos Canais de Comunicação

No nosso laboratório anterior, exploramos o cenário de uma comunicação em um "canal ideal", onde os pacotes trafegavam do emissor para o receptor sem qualquer problema. Essa era uma simplificação didática fundamental para introduzir os conceitos básicos de transporte de dados.

No entanto, no mundo real das redes de computadores, a realidade é bem diferente. O meio físico de transmissão – seja ele um cabo de cobre, fibra óptica ou ondas de rádio – está constantemente sujeito a imperfeições: ruídos eletromagnéticos, interferências, atenuação do sinal, entre outros fatores. Essas imperfeições podem corromper os bits de um pacote de dados durante sua jornada.

Quando um pacote chega ao seu destino com um ou mais bits corrompidos, ele se torna inválido. As camadas de rede inferiores possuem mecanismos para detectar essas corrupções (como o checksum). Ao identificar um pacote corrompido, a resposta padrão é descartá-lo, pois não há como garantir que a informação ali contida seja a original. Dessa forma, para os protocolos de transporte (como o UDP e o TCP), um **erro de bit** no canal se manifesta como uma **perda de pacote**.

Neste laboratório, daremos um passo importante em direção à realidade. Iremos simular um canal "não confiável", introduzindo propositalmente erros de bit para observar diretamente as consequências da perda de pacotes em uma comunicação que não oferece mecanismos de recuperação. Utilizaremos o protocolo UDP, que por sua natureza "não confiável", não tentará retransmitir os pacotes perdidos.

## Objetivos

1. Compreender como erros de bit em um canal podem levar à perda de pacotes e qual o impacto para a comunicação.
2. Observar o comportamento de um protocolo de transporte não confiável (UDP) diante da perda de pacotes.
3. Reconhecer a necessidade e a importância de mecanismos de controle de erros e retransmissão para garantir a entrega confiável de dados em redes reais.

## Material Fornecido

- Script TCL para simulação de comunicação em canal com erros de bit: [02-rdt1-canal-nao-confiavel.tcl](#)
- Ferramenta de visualização: [NAM \(Network Animator\)](#).

## Simulação 01: Observando Perdas em Canal com Erros

Passo 1: Configuração e Execução

1. O script **02-rdt1-canal-nao-confiavel.tcl** configura um link de comunicação entre dois nós, **n0** (Emissor) e **n1** (Receptor). Um **Modelo de Erro (ErrorModel)** foi adicionado ao link para simular a ocorrência de erros de bit.
2. Execute o script utilizando o comando abaixo no seu Terminal.

### Execução no Linux

```
xhost +  
docker run --rm -it -e DISPLAY=$DISPLAY -v $PWD:/ns2 -v /tmp/.X11-  
unix:/tmp/.X11-unix gelirettore/ns2  
ns2 ns 02-rdt1-canal-nao-confiavel.tcl
```

### Execução no macOS

Lembre-se de que o **XQuartz** deve estar configurado e **xhost +** executado.

```
xhost +  
docker run --rm -it -e DISPLAY=host.docker.internal:0 -v $PWD:/ns2  
gelirettore/ns2 ns 02-rdt1-canal-nao-confiavel.tcl
```

## Passo 2: Análise do Comportamento e Impacto

1. Após a execução, o **NAM (Network Animator)** será aberto, exibindo a topologia da sua simulação. Inicie a reprodução para observar o fluxo de pacotes. Você deve notar que, diferentemente da simulação anterior, alguns pacotes que partem do **Emissor** não chegarão ao **Receptor**, sendo descartados no caminho devido aos erros de bit.

### Para Refletir e Discutir:

- Compare o comportamento do fluxo de pacotes nesta simulação com a simulação do "canal ideal" (sem erros). Qual é a principal diferença observada?
- Na simulação anterior, todo pacote enviado era entregue. O que acontece com os dados contidos nos pacotes que são "perdidos" nesta simulação? O receptor tem alguma forma de saber que esses dados não chegaram?
- O protocolo UDP, por ser não confiável, simplesmente envia os dados. Se esta simulação representasse uma aplicação onde a entrega de dados é crítica (por exemplo, uma transferência de arquivo bancário ou uma mensagem de texto importante), quais seriam as consequências da perda de pacotes observada?
- A taxa de erro de bit (**set rate\_ no ErrorModel**) no script está inicialmente definida como **1e-2** (1 erro a cada 100 bits). Experimente alterar esse valor para **1e-0** (taxa de erro muito maior) e observe o impacto no número de perdas. O que aconteceria se fosse **1e-5** (taxa de erro muito menor)?

## Reflexão Final: A Necessidade de Confiabilidade

Este laboratório demonstra a inevitabilidade da perda de pacotes em canais de comunicação reais. Para muitas aplicações, o modelo de transporte não confiável do UDP é insuficiente, pois os dados simplesmente

se perdem sem que o emissor ou o receptor tomem conhecimento ou qualquer atitude para recuperá-los.

Para garantir que os dados cheguem corretamente ao destino, mesmo em face de perdas, os protocolos de transporte confiáveis precisam de mecanismos adicionais. Esses mecanismos devem ser capazes de:

1. **Detectar** que um pacote foi perdido ou corrompido durante a transmissão.
2. **Recuperar** o pacote perdido, geralmente solicitando uma retransmissão ao emissor.

#### **Para a Próxima Etapa: Rumo à Confiabilidade com Stop-and-Wait**

A partir do cenário de canal não confiável que acabamos de explorar, a próxima pergunta natural é: como podemos construir um protocolo de transporte simples que garanta a entrega confiável de dados, mesmo com perdas?