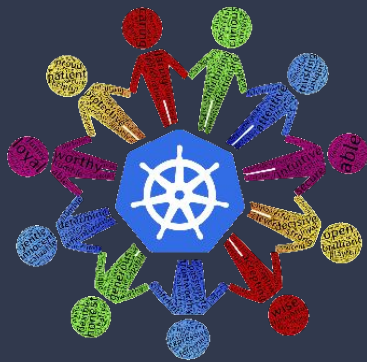


# HostPath

Armazenamento  
persistente e local no  
Kubernetes!



BL<sup>2</sup>



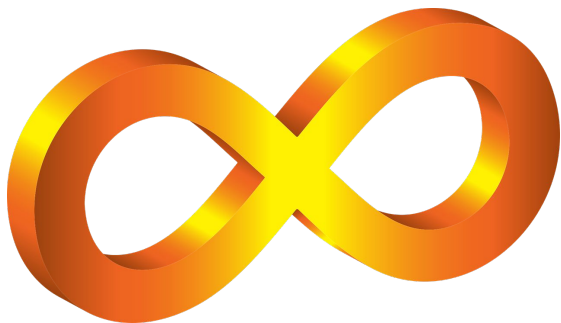
TI direto ao  
ponto!

# Volume Persistente e Local



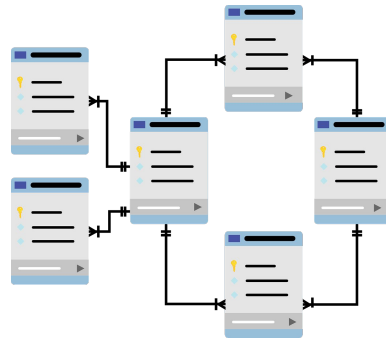
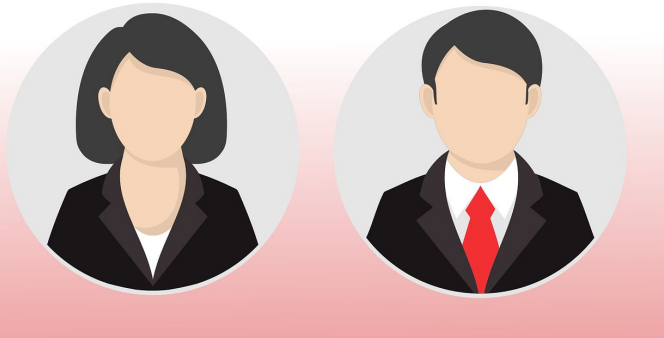
# Volume Persistente e Local

- Necessidade de armazenar os dados por um período (longo?) de tempo
- Garantia de durabilidade
- Volume localizado **na mesma máquina** do POD

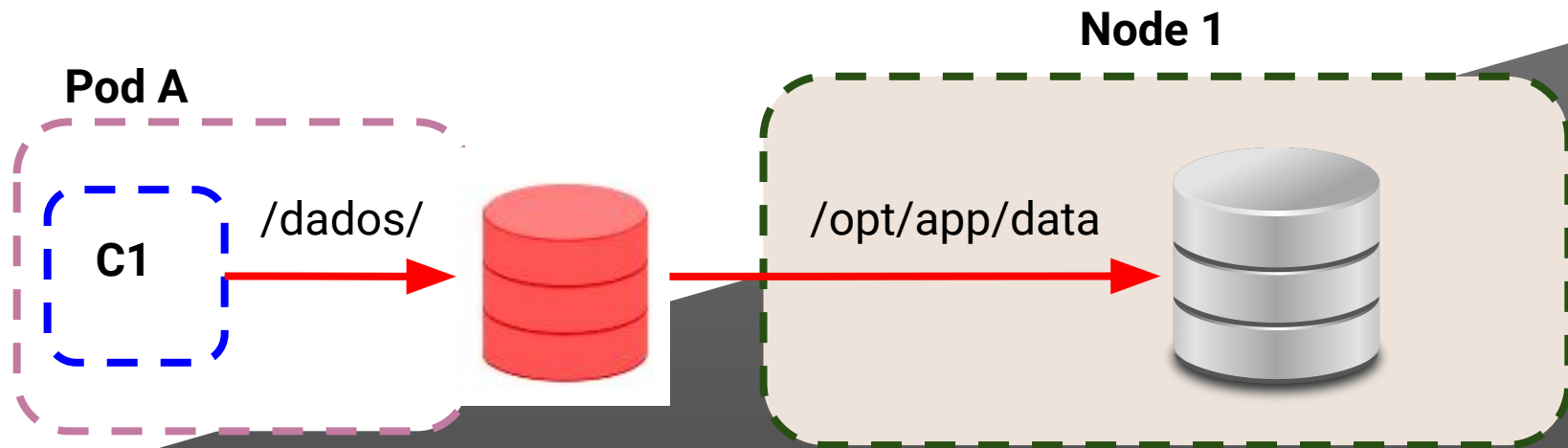


# Exemplos de Dados

- Armazenamento de arquivos enviados pelo usuário
- Informações presentes em um banco de dados
- Documentos gerados pela aplicação
- Logs de aplicações



# Volume HostPath



# Host Path

## Características



- Permite montar:
  - Arquivo
  - Diretório
  - Socket
  - Character device
  - Block devices
- Arquivos ou diretórios criados só podem ser escritos usando **root** ou configurando permissão para usuário direto na máquina

# Host Path

## Características



- Interessante para alguns casos:
  - Container necessita de acesso ao Docker. Ex: `/var/lib/docker`
  - Executar cAdvisor [1] no container. Ex: `/sys`
  - Permitir ao POD especificar se um caminho no nó deve existir, ou se deve ser criado, além de garantir que ele deve ser um arquivo ou um diretório por exemplo.

# Host Path

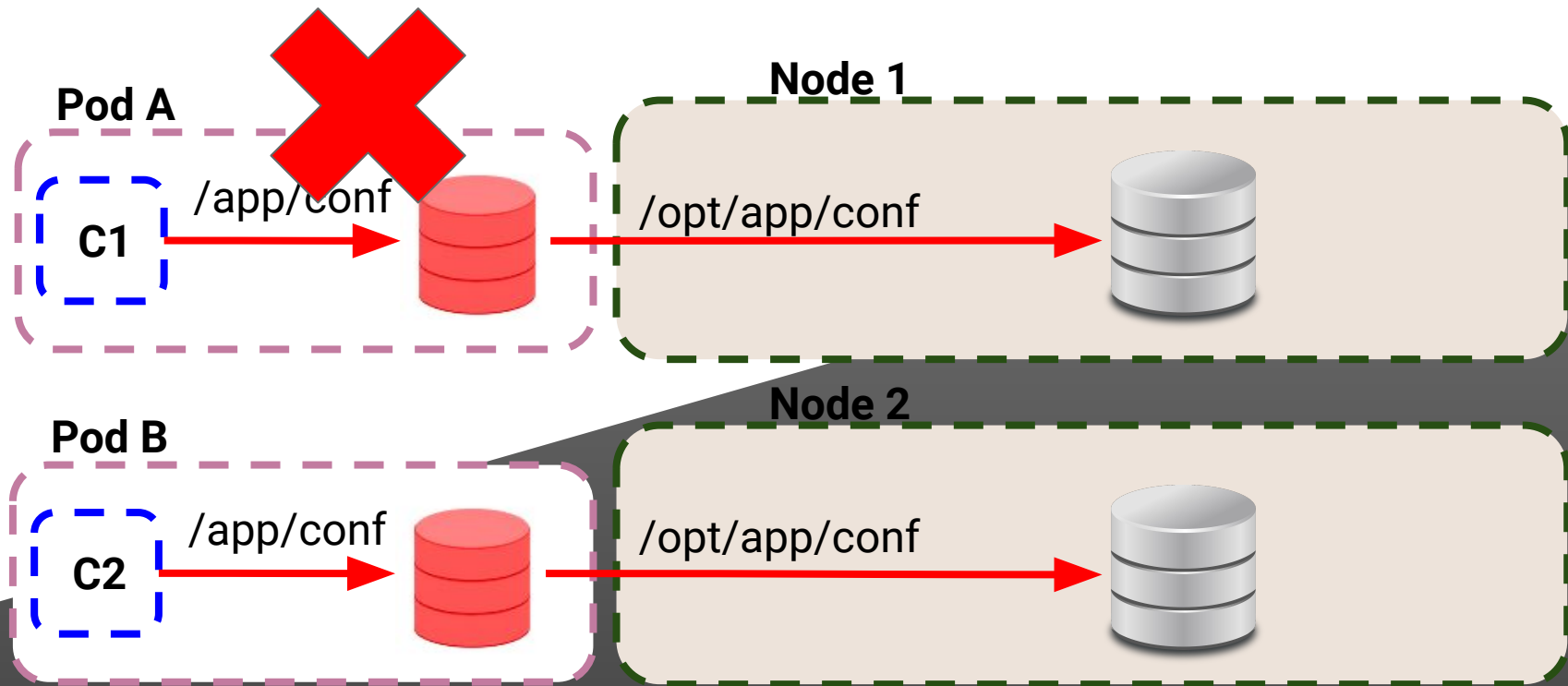
## Perigos e desvantagens



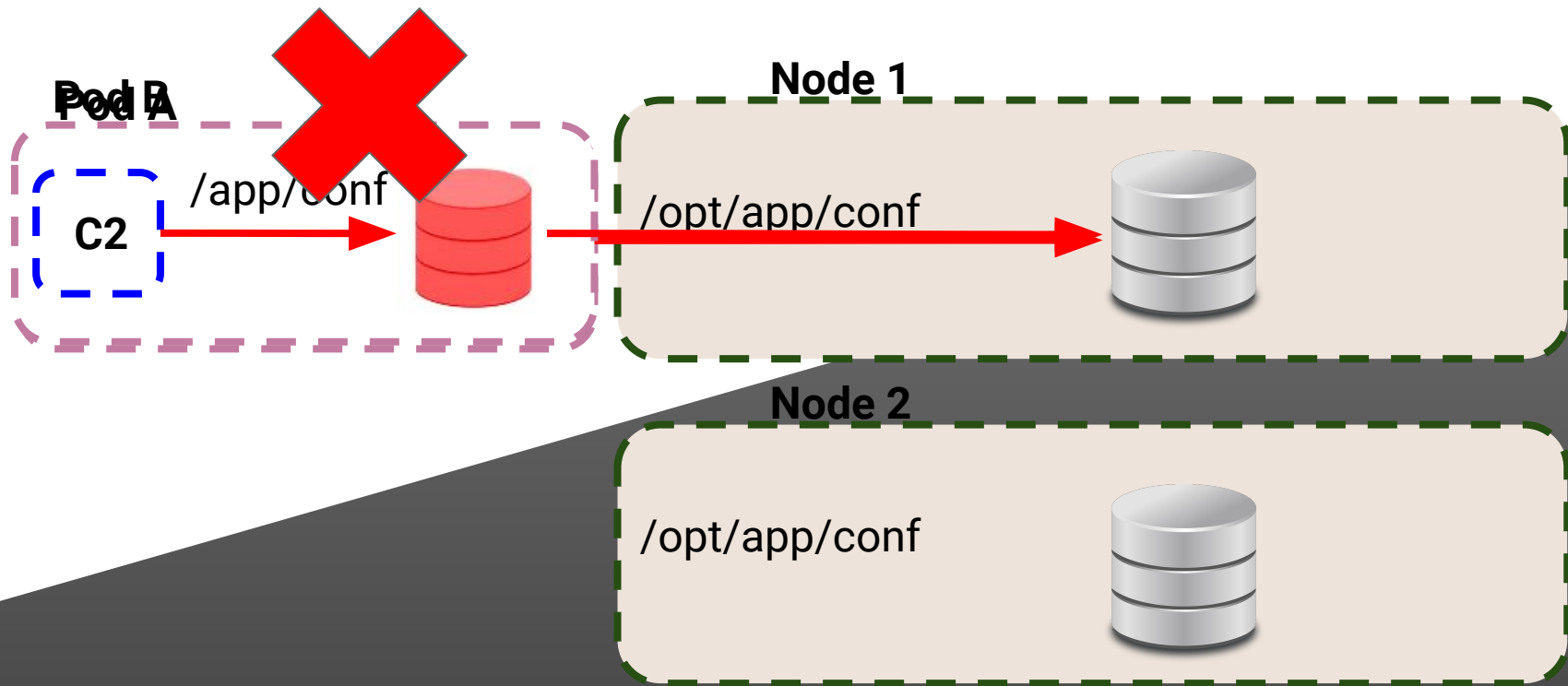
- Pode expor dados sensíveis:
  - Credenciais
  - APIs com privilégios (ex: socket do executor de containers)
- Comportamento de PODs com mesma especificação que rodem em nós diferentes pode variar bastante
- Somente use quando realmente for necessário!!!



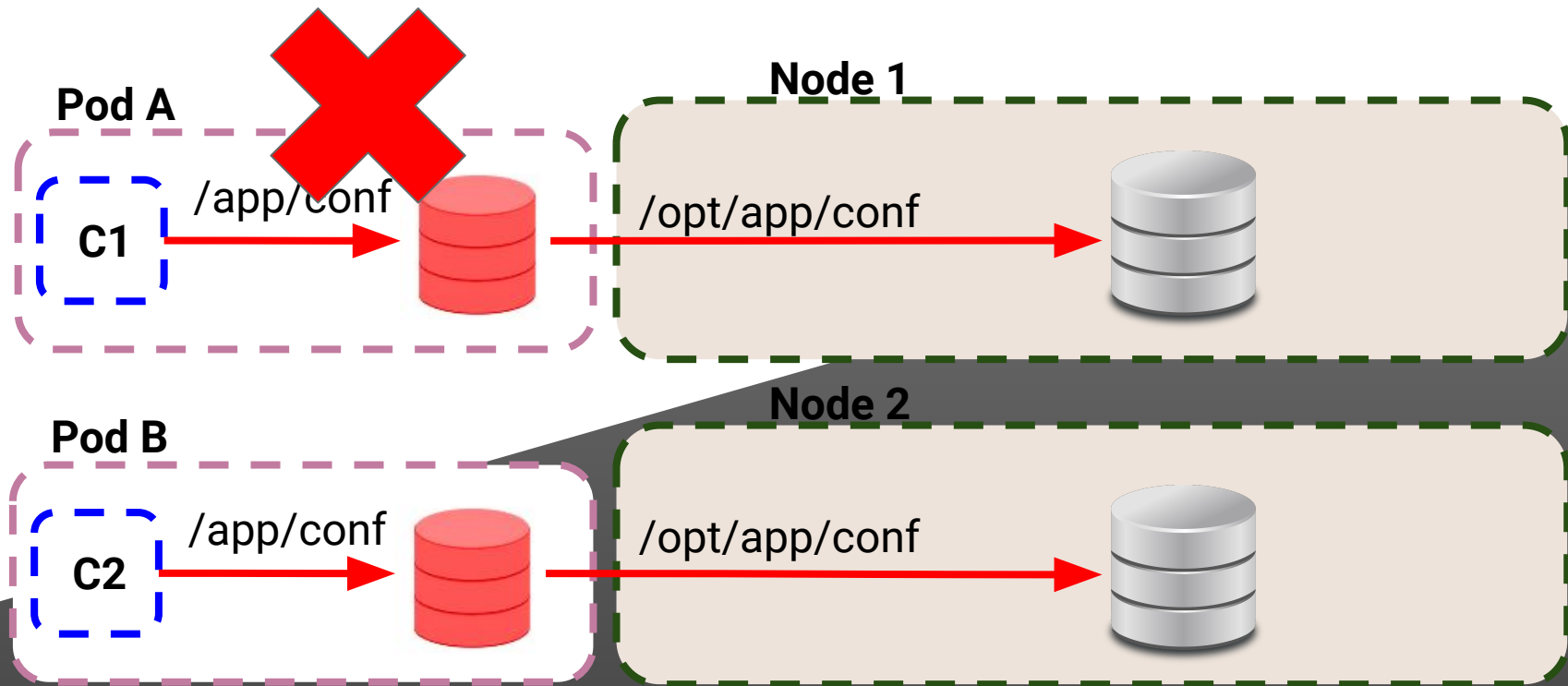
# Volume HostPath



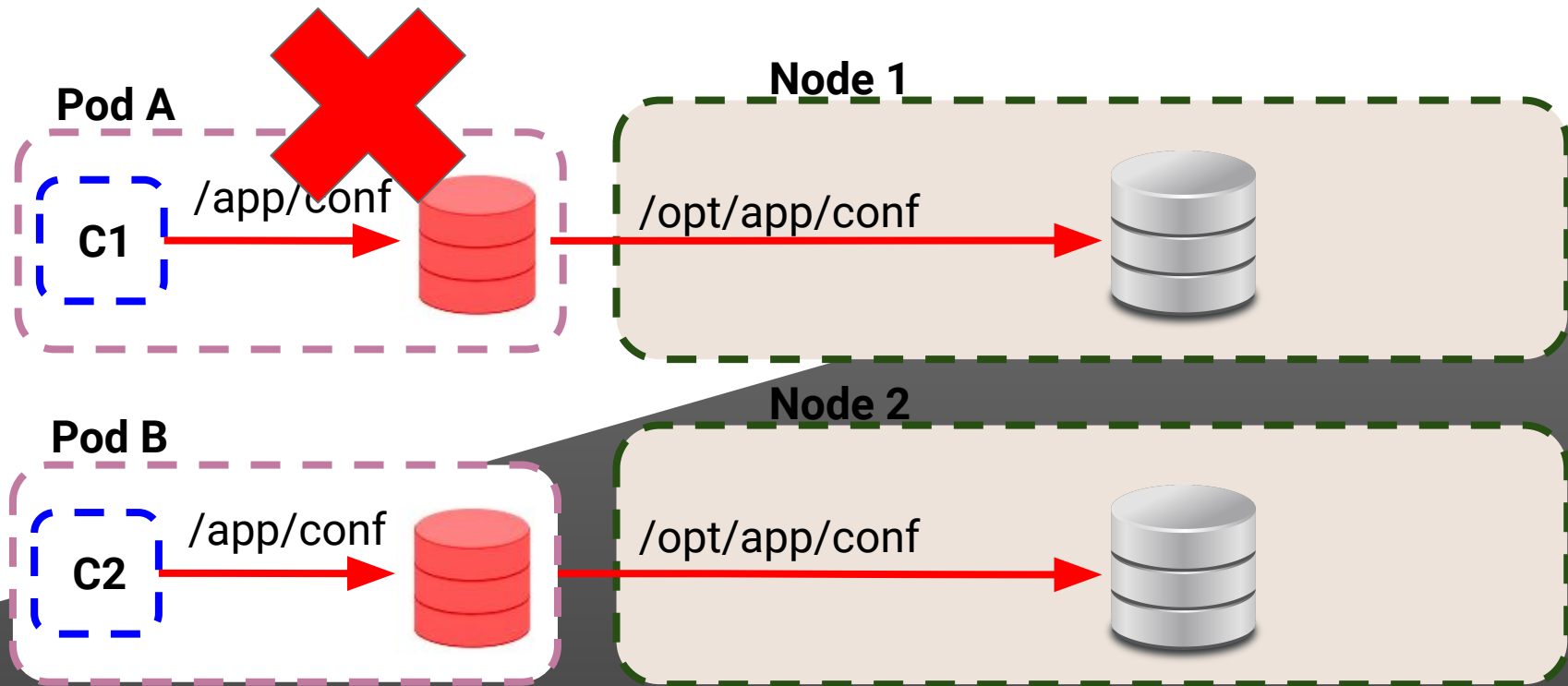
# Volume HostPath



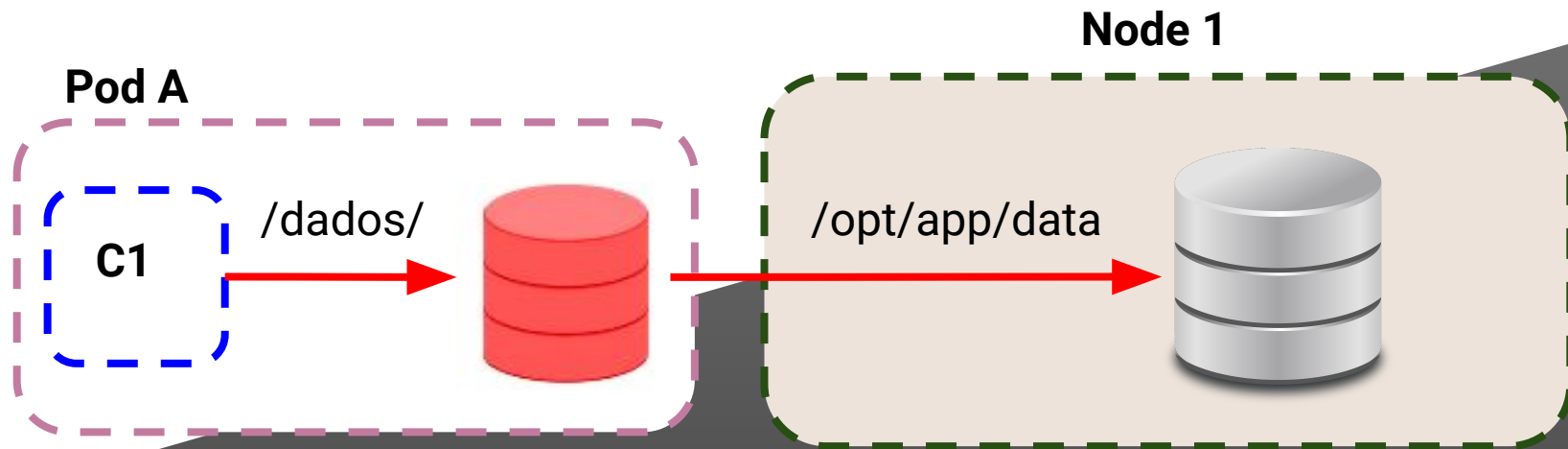
# Volume HostPath



# Volume HostPath



# Local Volume



# Resumo da Ópera



- Mapeamento persistente de um diretório ou arquivo do container para diretório ou arquivo da máquina
- Pode ser útil para acessar recursos da máquina (Docker)
- Pods rodando em máquinas diferentes **podem** ter diferentes comportamentos
- Problemas de segurança (recomendado modo somente de leitura)