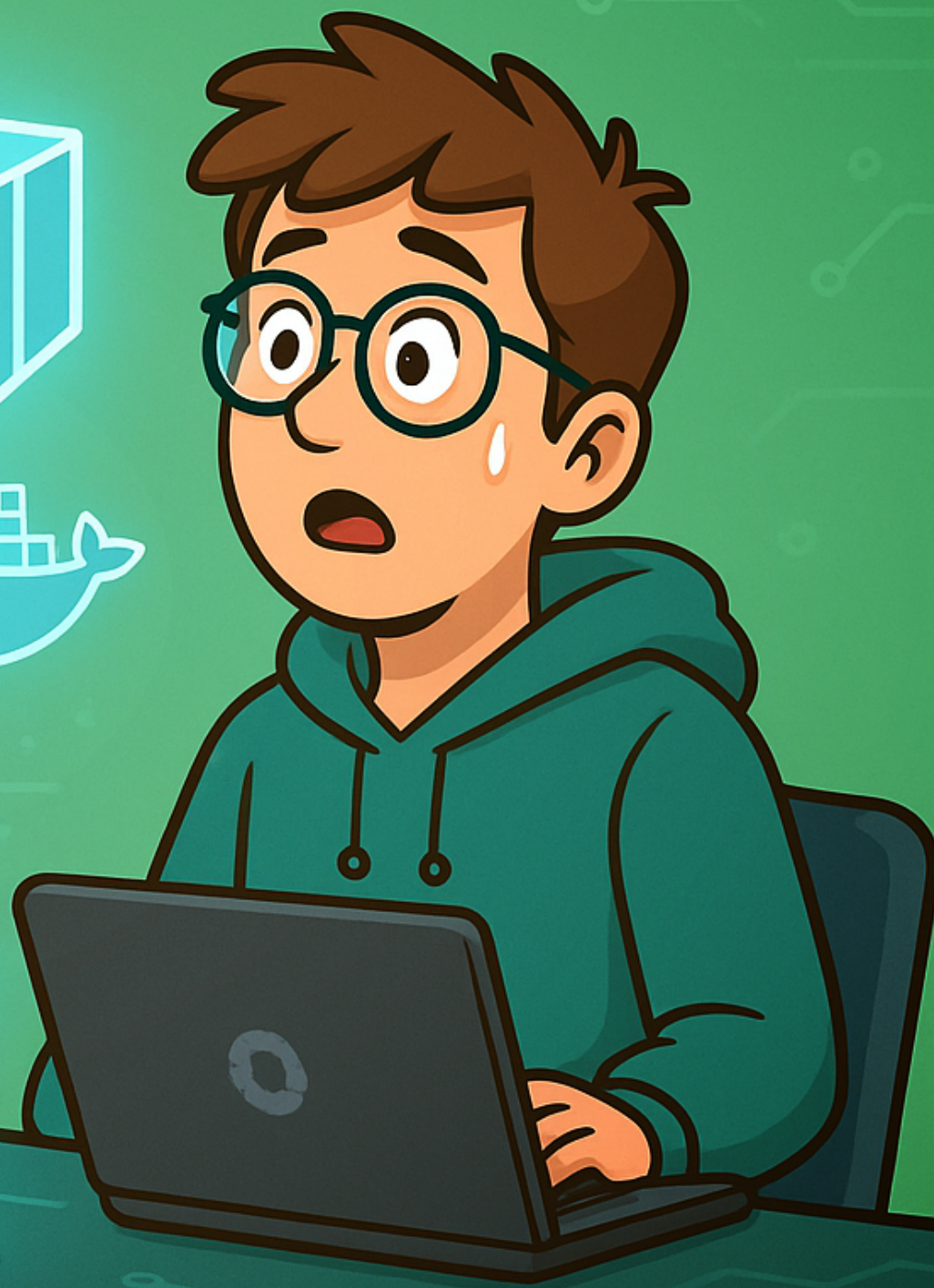
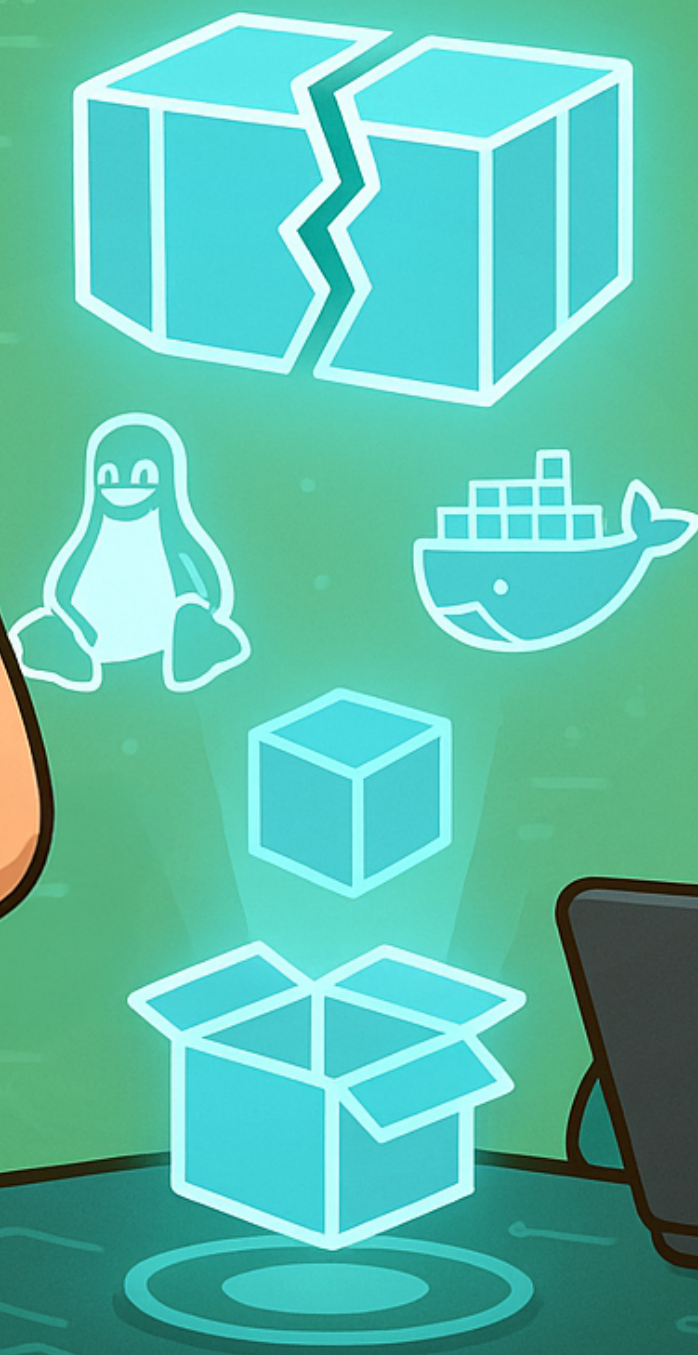
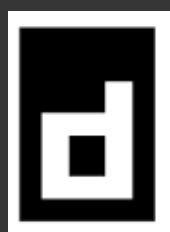


**SEM LINUX,
NÃO EXISTE
CONTAINERIZAÇÃO REAL.**



@fabricioveronez



Docker,
Kubernetes,
ECS, OpenShift,
containerd...

Todas
essas tecnologias
se apoiam em recursos
do kernel do Linux.

E é por isso que
**ignorar o Linux pode
comprometer tudo.**

@fabricioveronez

POR QUE ISSO ACONTECE?

Containers não são uma mágica do Docker.

Eles são **processos isolados** que usam funcionalidades como:

- | **cgroups** para controle de recursos

- | **namespaces** para isolamento

- | **chroot** para simular sistemas de arquivos

- | e muito mais...

Todos são recursos
nativos do **kernel Linux**.

@fabricioveronez



Docker não criou containers.

Ele só facilitou
o uso dos recursos que
já existiam no Linux.

Então sim: **é possível criar containers sem Docker,**
mas **NÃO SEM LINUX.**

@fabricioveronez

E O WINDOWS?

Mesmo no Windows,
quando você roda containers...

...**está rodando Linux por baixo,**
via WSL 2 ou Hyper-V.



■ Ou seja:
o Linux **AINDA ESTÁ LÁ,**
você só não vê.

@fabricioveronez

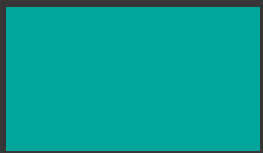
Se você trabalha
com DevOps,

precisa entender o
comportamento de
processos, permissões,



rede e arquivos **dentro do Linux**
porque **é esse sistema que está**
executando suas aplicações
em produção.

@fabricioveronez



Muita gente cria imagens rodando como **root**,
erra permissões, ignora estrutura de diretórios e
acaba gerando containers inseguros e frágeis.
Se falta base em Linux,
sobra falhas em produção.

@fabricioveronez



Se você quer
trabalhar com containers,

PARE DE FUGIR DO LINUX.

Aprender comandos, permissões,
sistema de arquivos e processos é
o que vai te dar **confiança real** pra
lidar com containers e clusters.

@fabricioveronez



Quer aprender
tudo isso
NA PRÁTICA

Leia a legenda
para saber mais sobre a
Formação DevOps Pro

