

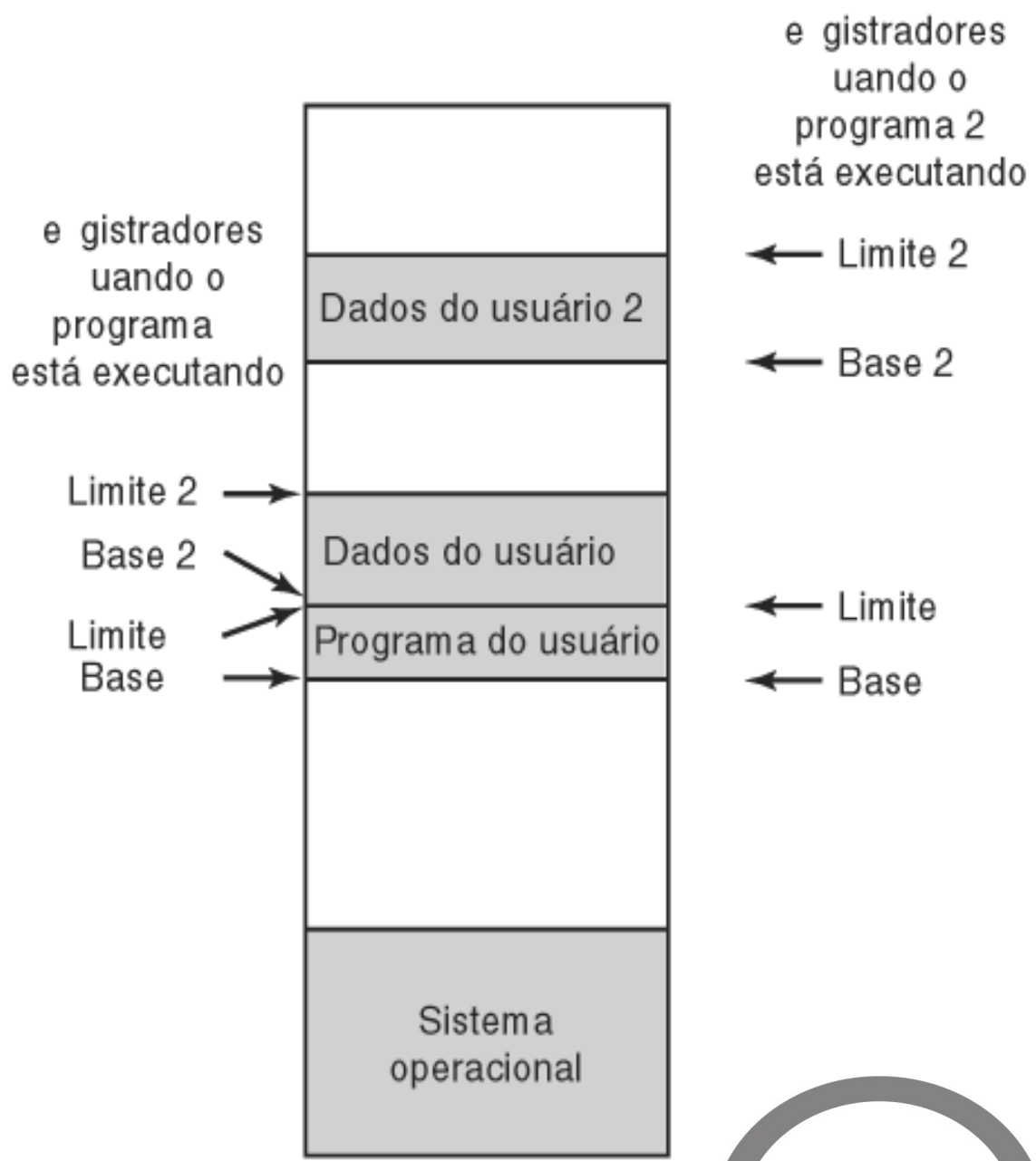
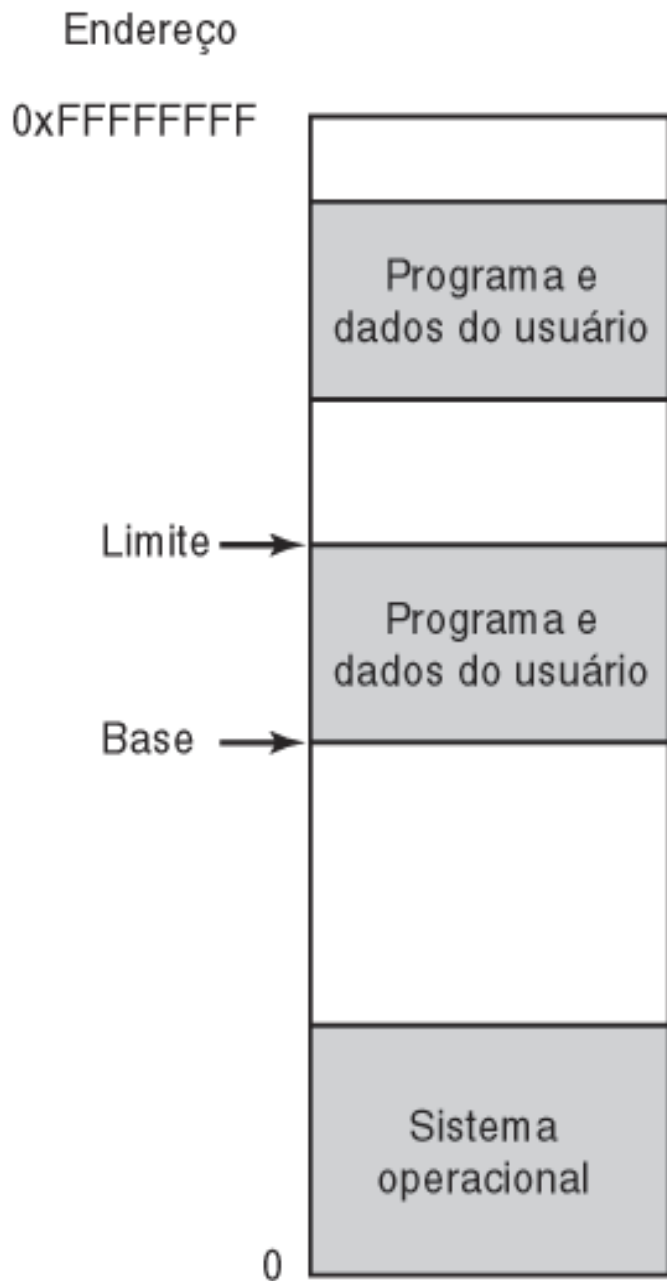


## Sistemas Operacionais

Rodrigo Rubira Branco  
rodrigo@kernelhacking.com  
rodrigo@fgp.com.br

# Memoria Compartilhada

- A memoria principal acaba por ser dividida entre diversos programas, o que causa dois problemas imediatos:
  - Protecao de programas um dos outros e protecao do SO de todos
  - Realocacao de processos
- A realocacao de processos apresenta problemas pois quando um programa e compilado e ligado, qual local da memoria fisica o mesmo executara?
- Sempre supoe-se que comecara do 0. Tem-se registradores base e limite.
- Todas as referencias sao somadas a base e verifica se ultrapassou o limite
- Podem haver (e na realidade ha, ja que a MMU controla isto e eh gerenciada pelo SO) muitos outros registradores (como para gerenciar a area de texto e a area de dados dos programas)



# Dispositivos de E/S

## - Controlador de Dispositivos

\* Interface mais simples para o SO (ex: buscar setor X do disco Y sera convertida pela controladora em numero de cilindro, setor e cabeca (contando que os cilindros mais externos tem mais setores que os internos, etc...))

\* Controla o dispositivo fisico em si (ex: aguardar o posicionamento do cabecote de disco e iniciar a leitura e o armazenamento dos dados e entao verificar o checksum dos mesmos)



# Dispositivos de E/S

## - Dispositivo em si

- \* Possuem interfaces simples, que facilitam a padronizacao
- \* Necessario para que um controlador de uma interface (como IDE) possa controlar qualquer dispositivo daquela interface (como discos IDE)



# Dispositivos de E/S

## - Driver de dispositivos

\* Em geral executam no kernel (devido a dificuldades de gerencia do acesso em modo usuario (problemas de micro-kernel)) e se comunicam com o controlador do dispositivo (traduzem as chamadas do kernel para o dispositivo)

## - Carregamento de drivers de dispositivos (3 formas)

\* Religar o driver com o nucleo e reiniciar o sistema

\* Configurar para carregar o driver e reiniciar (no boot, o SO carrega os drivers) -> Windows

\* Carregar drivers enquanto esta em execucao (carregamento dinamico) sem reiniciar -> Linux



## - Registradores de Controladoras

\* Necessarios para comunicacao/controle (ex: endereco do disco, contador de setores, endereco da memoria, indicador de operacao (leitura/escrita))

\* O driver recebe comandos do SO e os traduz para os valores a serem colocados nos registradores da controladora

\* Registradores de dispositivo podem ser gerenciados de duas formas

- Mapeando para o espaco de enderecamento do SO

-> Permite ler/escrever como se fossem a memoria principal

-> Nao sao necessarias instrucoes especiais de E/S

-> Programas de usuario podem ser controlados o acesso assim como a paginas da memoria

-> Gasta espaco de Enderecamento

- Espaco especial de portas de E/S

-> Cada registrador tem seu endereco de porta

-> Precisa de instrucoes (IN/OUT) para que os dispositivos leiam/escrevam nos registradores





FIM! Será mesmo?

DÚVIDAS?!?

Rodrigo Rubira Branco  
rodrigo@kernelhacking.com