

Sistemas Operacionais

Ítalo Mendes da Silva Ribeiro

Universidade Federal do Ceará

22 de janeiro de 2017

Sumário

Definição

Objetivos

Histórico dos SOs

Exemplos de SO

Componentes do SO

Processos

Segurança

Sumário

Definição

Objetivos

Histórico dos SOs

Exemplos de SO

Componentes do SO

Processos

Segurança

Definição

- ▶ Apenas com o hardware o computador seria de difícil utilização
- ▶ Comandos executados em linguagem de máquina (Assembly)
- ▶ Cada computador possui arquitetura e hardware diferente
- ▶ Seria uma grande dificuldade para os programadores desenvolver um programa de computador complexo
- ▶ Necessidade de uma camada intermediária entre hardware e programadores

Sistema operacional (SO) é um software ou conjunto de softwares, especialmente desenvolvido para oferecer, da forma mais simples e transparente possível, os recursos de um sistema computacional aos seus usuários, controlando e organizando o uso destes recursos de maneira que se obtenha um sistema eficiente e seguro.

Definição

EXEMPLO:

- ▶ Windows
- ▶ Linux
- ▶ Mac OS
- ▶ Free BSD
- ▶ Android
- ▶ iOS
- ▶ Windows Phone

Definição

Objetivos

Histórico dos
SOs

Exemplos de SO

Componentes do
SO

Processos

Segurança

Sumário

Definição

Objetivos

Histórico dos SOs

Exemplos de SO

Componentes do SO

Processos

Segurança

Objetivos

- ▶ Organiza a execução do hardware e do software do computador
- ▶ Gerencia a execução de múltiplos programas executando simultaneamente
- ▶ O sistema operacional faz o controle dos recursos do computador
- ▶ Evita violações de segurança, como controle de acesso aos dados de cada usuário
- ▶ Otimização dos recursos
- ▶ Facilidade para desenvolvedores com a transparência de comunicação com o hardware do computador

Objetivos

- ▶ Um programa para enviar comandos para um hardware utiliza funções do SO
- ▶ Diferentes hardwares instalados no computador respondem as mesmas funções do SO
- ▶ SO usa *drivers* para comunicação com os hardware
- ▶ Um *driver* é um software que possibilita a comunicação do SO com um hardware específico

Sumário

Definição

Objetivos

Histórico dos SOs

Exemplos de SO

Componentes do SO

Processos

Segurança

Histórico dos SOs

1940 a 1950

- ▶ Execução de programas requeria grande preparação com fitas, cartões perfurados e usuário especializado
- ▶ Computador executava apenas um programa por vez
- ▶ Programa era chamado de **job**
- ▶ Após execução do programa, fitas, cartões e outros componentes eram recolhidos para liberar a sala para o próximo usuário

Histórico dos SOs

1940 a 1950

- ▶ Os primeiros sistemas operacionais (SOs) tinham a função simples de suportar a separação de usuários e equipamentos
- ▶ Facilitavam a transição de um usuário para outro
- ▶ Um operador levava os equipamentos necessários para execução do job (programa) de um usuário e após a execução do job, os resultados eram recolhidos e levados pelo operador para o usuário

1960

- ▶ Computadores executam mais rapidamente, permitindo interatividade nos programas
- ▶ SOs passaram a suportar programas em tempo real
- ▶ Operadores passaram a ser um problema, pois poderiam não conhecer detalhes e especificidades importantes para execução de um programa, como o que fazer de cordo com a mensagem de erro apresentada pelo programa, pois normalmente só o usuário sabia

Definição

Objetivos

Histórico dos
SOs

Exemplos de SO

Componentes do
SO

Processos

Segurança

Histórico dos SOs

1970

- ▶ Computadores muito caros e mais rápidos
- ▶ Executar apenas um programa por vez não é suficiente
- ▶ SOs permitem execução de mais que um programa por vez
- ▶ O tempo de execução do processador é dividido igualmente entre os programas executados no computador (**multiprogramação**)

2000 em diante

- ▶ Computadores com processador de vários núcleos
- ▶ SO deve distribuir a execução de tarefas entre os núcleos para aumentar o desempenho do computador (balanceamento)
- ▶ SO pode dividir uma programa em partes menores para cada parte ser executada em um núcleo (escalonamento)

Histórico dos SOs

2000 em diante

- ▶ Computador realiza grande comunicação com outros computadores através das redes de computadores
- ▶ SO gerencia a comunicação e os recursos que podem estar disponíveis através da Internet
- ▶ Surgimento de computadores mais simples para tarefas específicas (sistemas embarcados ou *embedded systems*)
- ▶ SO de sistemas embarcados deve economizar bateria, executar programas com pouca ou sem interação humana, etc

Definição

Objetivos

Histórico dos
SOs

Exemplos de SO

Componentes do
SO

Processos

Segurança

Sumário

Definição

Objetivos

Histórico dos SOs

Exemplos de SO

Componentes do SO

Processos

Segurança

Exemplos de SO

Windows



- ▶ Mais usado em computadores pessoais
- ▶ Desenvolvido pela Microsoft
- ▶ SO comercial, ou seja, deve-se pagar para usá-lo
- ▶ Grande variedade e quantidade de software para as mais diversas atividades
- ▶ Devido a grande quantidade de usuários é muito visado por hackers
- ▶ Problemas de instabilidade e mal funcionamento

Exemplos de SO

Linux



- ▶ Mais usado em servidores
- ▶ Kernel criado por Linus Torvalds
- ▶ Baseado no sistema operacional UNIX
- ▶ Kernel desenvolvido por entusiastas e empresas mas com supervisão de Linus Torvalds

Exemplos de SO

Linux



- ▶ SO gratuito sob a licença GNU GPL (software livre)
- ▶ Código fonte aberto, ou seja, disponível para acesso, modificação e distribuição por qualquer um sob a licença GNU GPL
- ▶ Diversas distribuições como Ubuntu, Debian, Mint, Slackware, Gentoo, etc
- ▶ Uma **distribuição Linux** é um SO que usa o kernel do Linux mas que possui softwares e configurações próprias

Exemplos de SO

Linux



- ▶ Estável: frequência baixa de problemas de execução
- ▶ Portátil: usado em vários tipos de dispositivos
- ▶ Boa segurança: problemas descobertos e solucionados pela comunidade de desenvolvedores
- ▶ Pouco visado por hackers
- ▶ Pequena variedade de software para algumas tarefas

Exemplos de SO



FreeBSD

- ▶ Baseado no UNIX
- ▶ Kernel desenvolvido pela Universidade da Califórnia na cidade de Berkeley
- ▶ Mantido até hoje principalmente pela Universidade
- ▶ O mascote (Beastie) faz referência aos *daemons* que são programas do FreeBSD que ficam na memória para servir requisições de dados na memória

Exemplos de SO



FreeBSD

- ▶ Difícil aprendizado para usuários pouco experientes com computadores
- ▶ Usado principalmente em servidores
- ▶ Grande estabilidade
- ▶ Compatibilidade binária com o Linux, ou seja, programas executáveis do Linux funcionam com baixíssima perda de desempenho
- ▶ Variedade muito pequena de softwares

Exemplos de SO

OS X



- ▶ Usado em computadores da Apple
- ▶ Desenvolvido pela Apple
- ▶ Gratuito
- ▶ Segundo a usar GUI com janelas
- ▶ Primeiro a usar o conceito de ícones e desktop (área de trabalho)

Exemplos de SO

OS X



- ▶ Baseado no FreeBSD
- ▶ Objetivo de simplificar e facilitar os comandos e a interação do usuário
- ▶ Baixa variedade de software para algumas tarefas, embora vários softwares importantes possuam versão para OS X

Exemplos de SO

ios iOS

- ▶ Desenvolvido pela Apple
- ▶ SO móvel dos smartphones da Apple
- ▶ Suporte aos vários tipos de interação de dispositivos móveis (toque, acelerômetro, gestos, etc)
- ▶ Loja de aplicativos: **App Store**
- ▶ Lucro com vendas de aplicativos, além do controle de qualidade e segurança dos aplicativos

Exemplos de SO

iOS

- ▶ Siri: assistente pessoal
- ▶ Executa comandos feitos por voz como ligação de telefone, pesquisas web, envio de mensagens, etc
- ▶ Integração com o OS X e outros dispositivos Apple
- ▶ Variações para carros e smartwatches

Exemplos de SO

Android



- ▶ SO móvel mais utilizado no mundo
- ▶ Desenvolvido pela Google
- ▶ Mantido por algumas empresas lideradas pela Google
- ▶ Baseado no Linux
- ▶ Gratuito e de código fonte aberto
- ▶ Mascote: BugDroid

Exemplos de SO

Android



- ▶ Fabricantes de dispositivos podem mudar o Android para melhor adaptar-se ao seu produto
- ▶ Variações para smartTVs, carros, smartwatch e video games
- ▶ Loja própria: **Google Play**
- ▶ Menor controle na qualidade dos aplicativos vendidos, mas com grande controle na segurança
- ▶ Grande maioria dos aplicativos gratuitos

Exemplos de SO

Android



- ▶ Grande variedade de aplicativos para diversos tipos de tarefas
- ▶ Muito visado por hackers
- ▶ Constantes problemas de segurança
- ▶ Instalação de aplicativos a partir de lojas piratas de aplicativos

Exemplos de SO

Windows Phone



Windows Phone

- ▶ Desenvolvido pela Microsoft
- ▶ SO comercial
- ▶ Loja própria: **Windows Phone Store**
- ▶ Média variedade de aplicativos

Exemplos de SO

Windows Phone



Windows Phone

- ▶ Alguns aplicativos famosos não possuem ou fornecem baixo suporte ao Windows Phone
- ▶ Assistente: Cortana
- ▶ Recebe comando de voz para realizar várias tarefas
- ▶ Fadado ao fim já que a Nokia usará Android nos seus smartphones

Sumário

Definição

Objetivos

Histórico dos SOs

Exemplos de SO

Componentes do SO

Processos

Segurança

Componentes do SO

Shell

- ▶ Responsável pela comunicação do usuário com o sistema operacional
- ▶ Verifica se o comando do usuário está correto. Quando existem erros no comando, mostra uma mensagem de erro
- ▶ Tipos mais comuns:
 - ▶ **Linha de comando:** terminal ou console, onde todos os comandos são textuais
 - ▶ **GUI (*Graphical User Interface*)** composta por botões, janelas, menus e outros elementos gráficos

Definição

Objetivos

Histórico dos SOs

Exemplos de SO

Componentes do SO

Processos

Segurança

Componentes do SO

Kernel

- ▶ Núcleo, *core* ou *kernel* é a parte do SO responsável por todas as operações de controle do SO
- ▶ Executa os comandos enviados pelo shell
- ▶ Mais importante e mais utilizada
- ▶ Implementada em assembly
- ▶ Gerencia a execução dos programas, sistema de arquivos e a memória do computador

Definição

Objetivos

Histórico dos
SOs

Exemplos de SO

Componentes do
SO

Processos

Segurança

Sumário

Definição

Objetivos

Histórico dos SOs

Exemplos de SO

Componentes do SO

Processos

Segurança

Definição de processo

Processo computacional ou **processo** é uma atividade que ocorre em meio computacional, usualmente possuindo um objetivo definido, duração finita e utilizando uma quantidade limitada de recursos computacionais. Também pode ser chamado de *task*.

Definição de processo

Processo é um programa em execução que envolve:

- ▶ código do programa
- ▶ dados
- ▶ registradores
- ▶ sua pilha (stack)

Definição de processo

Multiprogramação

- ▶ Multiprogramação permite que vários processos sejam executados ao mesmo tempo
- ▶ Alternância entre processos
- ▶ Impressão de paralelismo
- ▶ Paralelismo verdadeiro somente com dois ou mais processadores

Definição de processo

Multiprogramação

- ▶ A alternância de processos é fundamental para o bom desempenho do sistema, pois evita que o processador fique parado esperando dados vindos da memória secundária ou outros dispositivos de E/S
- ▶ Enquanto os dados necessários por um processo são copiados da memória secundária para a memória principal, um outro processo pode ser executado

Definição de processo

Alternância ou chaveamento entre processos

- ▶ **Escalonador** (*scheduler*) coordenada a utilização do processador pelos processos
- ▶ Escalonador mantém uma tabela de processos, onde são guardadas as informações dos processos em execução
- ▶ Possui todas as informações de um processo como sua prioridade e endereços na memória
- ▶ **Temporizador** é um hardware que conta o tempo (*time slice*) de cada processo e informa a CPU quando o tempo acaba

Definição

Objetivos

Histórico dos SOs

Exemplos de SO

Componentes do SO

Processos

Segurança

Definição de processo

Alternância ou chaveamento entre processos

- ▶ Quando o tempo de um processo acaba é gerada uma **interrupção**
- ▶ Quando ocorre uma interrupção, o escalonador troca processo em execução na CPU por outro processo
- ▶ Se um processo necessita de informações que precisam ser copiadas da memória secundária para a memória principal, ocorre uma **interrupção de E/S** e o processo é trocado por outro

Definição

Objetivos

Histórico dos SOs

Exemplos de SO

Componentes do SO

Processos

Segurança

Definição de processo

Etapas da troca de processos no processador

- ▶ 1 - O processador empilha o *program counter* e também o conteúdo dos registradores do processador
- ▶ 2 - O estado do processo cuja execução foi interrompida é mudado para **pronto**
- ▶ 3 - O conteúdo dos registradores e o *program counter* empilhados são copiados para uma área de controle própria do processo interrompido (cada processo possui uma área própria). Isso é necessário para preservar o contexto do processo interrompido

Definição de processo

Etapas da troca de processos no processador

- ▶ 4 - O escalonador faz uma consulta para determinar o próximo processo a ser executado no processador
- ▶ 5 - O escalonador cópia o conteúdo dos registradores e do *program counter* armazenados na área de controle do processo que executará para a pilha do processador
- ▶ 6 - O temporizador é atualizado

Definição de processo

Etapas da troca de processos no processador

- ▶ 7 - A interrupção é finalizada
- ▶ 8 - O processador restaura seus registradores e o program counter com base no conteúdo das pilhas e continua a execução do processo atual

Definição de processo

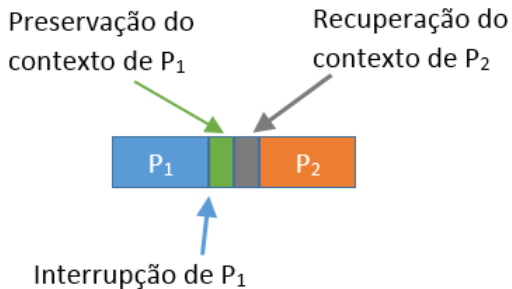


Figura: Representação do escalonamento de processos.

Sumário

Definição

Objetivos

Histórico dos SOs

Exemplos de SO

Componentes do SO

Processos

Segurança

- ▶ SO responsável de várias formas pelo sistema computacional, como as formas citadas a seguir:
- ▶ Evitar que um arquivo se corrompa
- ▶ **Um arquivo esta corrompido** quando não é possível acessar os dados armazenados nele
- ▶ Caso um arquivo seja corrompido, devem existir maneiras de recuperá-lo
- ▶ Impossibilitar a perda de dados vitais para uma pessoa ou empresa

- ▶ Impedir acesso não autorizado para arquivos
- ▶ Contas de usuários
- ▶ Principal usuário de um sistema computacional: administrador ou super usuário
- ▶ Administrador tem total controle e acesso para arquivos e configurações do sistema computacional
- ▶ Falhas de segurança do SO pode permitir que usuários não autorizados (hackers) acessem o sistema como administrador

Segurança

- ▶ SO deve ser constantemente atualizado para correção de bugs (falhas de execução) ou problemas de segurança
- ▶ Software como antivírus e firewalls são importantes para a segurança do computador
- ▶ **Antivírus** monitora os softwares em execução e arquivos copiados para o computador buscando códigos maliciosos (prejudiciais)
- ▶ **Códigos maliciosos** são instruções que buscam roubar informações do computador ou ter acesso de administrador
- ▶ **Firewalls** são softwares que monitoram a entrada e saída de informações do computador pela rede, barrando a entrada ou a saída de dados não autorizadas

- ▶ A principal falha de segurança dos computadores são os próprios usuários
- ▶ Usuários por falta de conhecimento e realização de atividades ilegais instalam softwares com códigos maliciosos
- ▶ Criação de senhas previsíveis, simples e fáceis
- ▶ Exemplo de senhas frágeis: datas pessoais importantes, nome de entes queridos, **123456**, etc