
Trabalho Prático

Sistemas Operacionais

RELATÓRIO 2

Desenvolvimento

RTL (Real-Time Loading)

Adler Melgaço Ferreira

Hiago de Souza Cruz Alves e Silva

Pedro Elias Valadares Castanheira

Ronald Davi Rodrigues Pereira

1 Introdução

É fato que, atualmente, os computadores pessoais estão bastante populares na sociedade. Cada usuário tem um perfil e utiliza seu sistema de acordo com seus interesses. Grande parte desses usuários segue uma “rotina”, isto é, ao iniciar o sistema, abre um conjunto padrão de processos que usa com frequência, sejam eles navegadores, jogos, softwares de comunicação, editores, entre outros.

A partir disso, viu-se a possibilidade de tornar essa rotina mais eficiente. A proposta de alteração do kernel Real-Time Loading (RTL) visa identificar o conjunto de processos que cada usuário utiliza com frequência e inicializá-los previamente de forma automática e prioritária, garantindo uma melhor eficiência para o usuário.

Esse relatório contém o andamento do desenvolvimento dessa aplicação e alguns detalhes sobre ela, como o funcionamento esperado, possíveis problemas e desafios a serem vencidos.

2 Desenvolvimento

Em inicialização de processos, um dos maiores gargalos de eficiência é a troca de contexto de um processo em execução para um outro prestes a ser iniciado. Desse modo, ao realizar o carregamento prévio de partes essenciais de um processo, facilita-se a troca de contexto e, conseqüentemente, diminui-se o tempo para a sua inicialização.

Portanto, para suprir essa necessidade, está sendo utilizada uma ferramenta desenvolvida pelo grupo para inicializar esses processos de uma maneira customizada pelo usuário. Essa ferramenta será posteriormente integrada ao kernel do Linux e inicializada automaticamente.

2.1 O quê já foi feito

A ferramenta consiste em uma estrutura de dados “Command”, que é constituída de uma string para armazenar o comando para inicialização do processo e um contador de eventos (quantas vezes o processo foi executado/incluído). Esse contador será utilizado posteriormente para priorização e customização de inicialização de processos, de forma que os processos que possuem um maior número de eventos sejam prioritários frente a outros com um menor número de eventos.

A ferramenta possui 3 modos distintos, diferenciáveis pelos parâmetros passados por linha de comando:

1. `./realtimeloading.exe init` → Executa a criação de um banco de dados vazio “`rtldatabase.db`”, onde serão armazenados os processos, de forma a manter a informação de configuração e execução posteriormente à execução da ferramenta.
2. `./realtimeloading.exe config` → Modo de customização do usuário por meio de um menu de configuração. Esse modo contém opções de impressão dos processos armazenados no banco de dados do programa, bem como a inclusão manual, exclusão ou a modificação do tamanho máximo de processos que podem ser armazenados.
3. `./realtimeloading.exe` → Modo padrão e automático de inicialização de processos contidos no banco de dados. Esse modo será executado automaticamente a cada inicialização do sistema operacional.

A ferramenta também possui um gerador de log de erros que a execução do programa pode encontrar, como por exemplo tentar inserir um novo processo com o banco de dados lotado. Desse modo, caso algum erro seja detectado em tempo de execução do programa, ele será guardado em um arquivo de texto, evitando perda de informações.

A execução sequencial dessas tarefas acontece na seguinte ordem:

→ Programa começa → Lê o banco de dados → Salva os dados na estrutura de dados
→ Carrega os processos → Incrementa em um o contador de eventos de cada processo
→ Salva os dados resultantes da execução no banco de dados
→ Desaloca a estrutura de dados → Programa finaliza.

2.2 Possíveis problemas e desafios

Ao longo do desenvolvimento da ferramenta, foram descobertos alguns possíveis problemas com que teríamos que lidar durante o restante do trabalho. O primeiro desses problemas foi o de mensurar o tempo que cada processo demora para ser inicializado, ou seja, o tempo para acontecer a troca de contexto ou, pelo menos, a alocação dos recursos necessários à inicialização dele. Por enquanto, uma solução eficiente ainda não foi encontrada, mas ainda será feita uma busca mais aprofundada a respeito desse problema.

O segundo problema, e também um desafio a ser encarado, é o de integrar a ferramenta ao kernel do Linux sem que haja qualquer tipo de interferência no funcionamento do sistema operacional. Para isso, fizemos pesquisas e leituras de documentações do kernel, de modo a proporcionar ao grupo uma visão mais específica e especialista do assunto de integração com o kernel. Essas documentações, no entanto, são complexas e estão espalhadas, o que configura mais uma barreira à compreensão do funcionamento do kernel e como alterá-lo.

Esperamos obter sucesso em ambos desafios, de forma a possibilitar a realização de testes e comprovar a melhora de tempo de resposta do sistema operacional para a inicialização de processos como um todo.

3 Conclusão

O trabalho está agregando bastante experiência ao grupo em relação ao funcionamento dos processos no kernel, principalmente sobre inicialização e manipulação dos mesmos. Já foi obtida uma estrutura básica sobre como a alteração irá funcionar e as próximas etapas já estão planejadas.

Ainda há um longo caminho a ser percorrido para obter o funcionamento completo do novo kernel, mas a experiência obtida até o momento já foi bastante satisfatória e enriquecedora.

4 Referências Bibliográficas

<https://en.wikipedia.org/wiki/Linux>

<https://www.linux.com/what-is-linux>

[https://en.wikipedia.org/wiki/Kernel_\(operating_system\)](https://en.wikipedia.org/wiki/Kernel_(operating_system))

<https://github.com/torvalds/linux>

http://linuxcommand.org/man_pages/ps1.html