



Imagens de Paul Souders/Getty

Em 19 de janeiro, Linus Torvalds lançou oficialmente o kernel [Linux](#) 6.13. Não posso chamar o kernel 6.13 de uma versão importante, mas ainda é um avanço em desempenho, segurança e suporte de hardware. Isso não é pouca coisa.

Um dos recursos de destaque do Linux 6.13 é a introdução do suporte à preempção lenta. Este novo modelo simplifica a lógica de preempção e as opções de configuração do kernel, melhorando potencialmente a capacidade de resposta geral do sistema.

Além disso: Como solucionar problemas de inicialização de aplicativos Linux com o comando journalctl

Os desenvolvedores do kernel Linux, especialmente Peter Zijlstra, trabalham nesta área há vários anos. A ideia por trás da preempção lenta é simplificar o agendamento do kernel e fornecer desempenho geral mais rápido. No entanto, essa tarefa é mais fácil de falar do que fazer.

Veja bem, o problema é que não existe uma solução única para determinar se é melhor para um sistema responder rapidamente a um evento, por exemplo, clicar em um botão do mouse ou fornecer trabalhos de CPU de longa duração o tempo todo que eles precisam. . Por



padrão, o kernel do Linux possui quatro maneiras diferentes de lidar com esse problema. Agora, com preempção preguiçosa, temos cinco.

Este novo padrão é simples no fundo. A nova flag, `TIF_NEED_RESCHEDED_LAZY`, informa ao kernel que o reescalonamento será necessário em algum momento, mas não precisa ser agora. O efeito final deve ser um kernel com melhor rendimento para trabalhos longos e código mais limpo.

Eventualmente, como observou o desenvolvedor do kernel Linux Jonathan Corbet: “O objetivo final deste trabalho é ter um agendador com apenas dois modos não em tempo real: `PREEMPT_LAZY` e `PREEMPT_FULL`. O modo lento ocupará um lugar entre `PREEMPT_NONE` e `PREEMPT_VOLUNTARY`, substituindo ambos deles, no entanto, não precisará dos pontos de preempção voluntários que foram adicionados para os dois modos que substitui. Como a preempção agora pode acontecer em quase qualquer lugar, não há mais um. precisa habilitá-lo em pontos específicos.”

O kernel também introduz suporte à otimização AutoFDO e Propeller ao compilar Linux com o cada vez mais popular compilador LLVM Clang. Desenvolvedores Linux sérios que usam este compilador deverão ver construções de código mais rápidas.

Além disso: 10 aplicativos Linux que instalo em cada nova máquina (e por que você deveria também)

Como sempre, o novo Linux 6.13 vem com compatibilidade de hardware aprimorada. No topo da lista está um novo driver AMD 3D V-Cache Optimizer para a nova CPU favorita dos jogadores, o processador multi-CCD Ryzen X3D. Os processadores de servidor AMD EPYC 9005 “Turin” também são suportados para pessoas que se preocupam mais com servidores do que com desktops. Do lado da Intel, o 6.13 suporta gráficos Intel Xe3 e processadores Panther Lake.

Não é apenas o novo hardware que está recebendo suporte. O kernel 6.13 também melhora o suporte para dispositivos Apple mais antigos, incluindo vários iPads e iPhones. Além disso, há suporte para Apple System on Chips (SoCs) pré-M1. Esta atualização é significativa para usuários de hardware legado da Apple, pois estabelece as bases para a execução do Linux em dispositivos que incluem SoCs Apple A7, A8, A8X, A9, A9X, A10, A10X e A11. A atualização também oferece suporte para iPhone 5S, série iPhone 8, série iPhone X, iPad Air e iPad Mini 2/3.

No entanto, antes de ficar muito entusiasmado com este novo suporte da Apple, lembre-se de que este é um suporte estatal básico. Com ele, você pode inicializar o kernel principal do Linux, mas pedir mais é demais. Resumindo, esta opção não se destina a usuários Joe ou Jane, mas a entusiastas de dispositivos Apple da velha escola que desejam explorar a possibilidade de trazer o Linux para essas plataformas mais antigas.



Uma melhoria que mais pessoas irão apreciar é que o Linux agora suporta a execução de Linux em máquinas virtuais protegidas no [ARM Confidential Compute Architecture \(CAA\)](#). Esse suporte é importante para pessoas que exigem segurança computacional confidencial em seus sistemas com processador ARM.

Além disso: Como criar pontos de restauração do sistema no Linux com Timeshift - e por que você deveria

Como Torvalds observou em seu anúncio de lançamento: “Nada horrível ou inesperado aconteceu na semana passada, então marquei e empurrei a versão final 6.13”. Este processo de lançamento tranquilo é um bom presságio para a estabilidade da nova versão do kernel.

Com o Linux 6.13 já disponível, as atenções se voltam para o desenvolvimento do Linux 6.14, com a janela de mesclagem já aberta. Espera-se que o próximo grande lançamento do kernel chegue no final de março de [2025](#). A grande novidade aqui deve ser mais suporte ao Rust no kernel do que antes.

Como sempre, os usuários de distribuições de lançamento contínuo provavelmente verão esta atualização do kernel primeiro, enquanto outros podem precisar esperar pelo próximo ponto ou lançamento principal de sua distribuição para receber o Linux 6.13. Claro, se preferir não esperar, você pode baixar o kernel Linux 6.13 hoje mesmo na árvore git de Torvalds ou no site kernel.org. A maioria das pessoas, entretanto, fará melhor em esperar que seu distribuidor Linux favorito o inclua em seus repositórios de software estáveis.

Finalmente, Linus Torvalds anunciou que premiaria cinco desenvolvedores aleatórios de kernel Linux com seus pedais de efeito de guitarra feitos à mão. O quê, você pode perguntar?

Além disso: Como manter o Linux otimizado (e economizar tempo) com Stacer

Bem, você deve saber que Torvalds é um grande mergulhador, mas talvez não saiba que um de seus hobbies é construir kits de LEGO. Ultimamente, porém, ele também começou a construir pedais de guitarra a partir de kits, que chamou de “LEGO para adultos com ferro de soldar”.

Torvalds construiu cinco pedais e enviará um para desenvolvedores selecionados aleatoriamente. Ele gostou do processo de construção do pedal. Então, agora você sabe o que Torvalds faz quando está frio demais para mergulhar e ele precisa de uma pausa no kernel.