

# 1

## Introdução

*Sistema operacional extensível* é uma abordagem de projeto de sistemas operacionais que foi originada no final da década de 1960 (1) e tem sido bastante explorada a partir da década de 1990 até a atualidade (2, 3, 4, 5). Nessa abordagem, é sustentado que podemos aumentar a flexibilidade, confiabilidade e desempenho de um sistema operacional através da sua extensibilidade. Essa abordagem se baseia no princípio que sistemas operacionais precisam permitir o uso de extensões para atender a mudanças nos seus requisitos iniciais de projeto (1). Os sistemas operacionais extensíveis focam na idéia-chave de que sistemas operacionais de uma forma geral não são capazes de antever os requisitos de todas as suas aplicações, portanto eles devem permitir a adaptação do seu funcionamento para poder atender a novos requisitos (6).

Em outro contexto, o desenvolvimento de aplicações, existe uma importante tendência em dividir sistemas complexos em duas partes — *núcleo* e *configuração* — e em combinar linguagens de programação de sistema e linguagens de script para o desenvolvimento dessas partes (7, 8). A linguagem de programação de sistema é utilizada para escrever os componentes que formam o núcleo do sistema; e a linguagem de script é utilizada para escrever a configuração, ou seja, definir o comportamento final do sistema, ligando os componentes do núcleo (7, 8). Linguagens de programação de sistema, como por exemplo C e C++, são linguagens de mais baixo nível e geralmente estaticamente tipadas e compiladas. Linguagens de script, como por exemplo Lua, Tcl, Python e Perl, são linguagens de mais alto nível e geralmente dinamicamente tipadas e interpretadas. Essa abordagem de desenvolvimento de aplicativos é comumente chamada de *scripting*. O uso desta abordagem traz vantagens significativas ao desenvolvimento de aplicativos. Primeiro, ela aumenta a produtividade de desenvolvimento, oferecendo um ambiente mais ágil pelo uso de uma linguagem de mais alto nível e interpretada (7). Segundo, ela aumenta a flexibilidade do desenvolvimento, pois oferece uma linguagem para a extensão do aplicativo em tempo de execução, em vez da simples customização através da configuração de parâmetros (8).

Neste trabalho, nós propomos a junção dessas duas abordagens de de-

envolvimento — *sistema operacional extensível e scripting* — para possibilitar o uso de linguagens de script no desenvolvimento de sistemas operacionais. Nós chamamos essa nossa proposta de abordagem de *sistema operacional scriptável*. De forma semelhante ao *scripting* de aplicativos, um sistema operacional scriptável utiliza scripts para ligar seus componentes e definir o seu comportamento final. De forma semelhante a um sistema operacional extensível, um sistema operacional scriptável possibilita a extensão do seu funcionamento, contudo, utilizando uma linguagem de script como linguagem de desenvolvimento das extensões.

Baseados na idéia de sistema operacional scriptável, desenvolvemos um protótipo para possibilitar o *scripting* de *kernel* de sistema operacional utilizando Lua, ou seja, para possibilitar o *scripting* da principal parte de um sistema operacional. Desenvolvemos esse protótipo, chamado *Lunatik*, para NetBSD e Linux. O desenvolvimento de Lunatik foi iniciado no nosso trabalho anterior com o objetivo apenas de executar scripts Lua dentro do *kernel* Linux (9). Desta forma, a primeira versão de Lunatik consistia basicamente em um interpretador Lua embutido no *kernel* Linux. A versão de Lunatik que apresentamos nessa dissertação é uma infra-estrutura de *scripting* que possui, além de um interpretador Lua embutido em *kernel*, uma interface de programação para desenvolvedores de *kernel* e uma interface de usuário para a carga de scripts no *kernel*. Entretanto, Lunatik não apresenta um sistema operacional scriptável em si, e sim um conjunto de ferramentas que possibilitam a adaptação de *kernels* com objetivo de torná-los “scriptáveis”.

O resto desta dissertação é organizada como segue. O capítulo 2 apresenta os conceitos básicos que serão utilizados no restante desta dissertação. A seção 2.1 apresenta a definição para o conceito de sistema operacional. A seção 2.2 apresenta uma discussão sobre sistemas operacionais extensíveis e seus requisitos e características de projeto e implementação; A seção 2.3 apresenta uma discussão sobre linguagens de script e seu uso e motiva a escolha de Lua como linguagem para *scripting* de *kernels* de sistema operacional. O capítulo 3 descreve a nossa proposta de sistema operacional scriptável, apresenta uma discussão sobre suas principais características, alguns casos de uso e trabalhos relacionados. O capítulo 4 apresenta Lunatik, o nosso protótipo para *scripting* de *kernel* baseado em Lua; descreve o seu projeto e implementação; e apresenta experimentos de *scripting* de *kernel*. O primeiro experimento foi aplicado ao subsistema CpuFreq do Linux, que é responsável pelo controle de frequência e voltagem de CPU (*central processing unit*) no Linux. O segundo experimento foi aplicado à composição de chamadas de sistema no NetBSD. Finalmente, o capítulo 5 apresenta as nossas conclusões.