

Proposta de Oficina Prática

1. Dados de identificação

1.1. Título da oficina prática

Construção de Linux Embarcado com Suporte a Gerenciamento

1.2. Autor(es), instituição(ções), endereço(s), telefone(s), e-mail(s)

- Carlos Roberto Moratelli, Digitel Indústria Eletrônica S/A, Rua João Inácio, 1165. Telefone (51) 9315-7618. crmoratelli@gmail.com
- Marcelo Veiga Neves, Digitel Indústria Eletrônica S/A, Rua João Inácio, 1165. Telefone (51) 9316-0821. marcelo.veiga@gmail.com

1.3. Indicação do(s) autor(es) que ministrará(rão) a oficina prática

Carlos Roberto Moratelli e Marcelo Veiga Neves

2. Dados gerais

2.1. Objetivos da oficina prática

Esta oficina prática tem como objetivo a construção de um sistema embarcado baseado em Linux com suporte a gerenciamento via rede. A construção será realizada a partir do zero. Isso implica em obter o kernel Linux e compilá-lo com suporte a pilha de protocolos, criar um sistema de arquivos básico e instalar e configurar serviços de gerenciamento. Finalmente, usar uma máquina virtual para executá-lo e testar seu funcionamento.

2.2. Revisão bibliográfica: tópicos a serem abordados

Um sistema embarcado é um sistema microprocessado no qual o computador é dedicado ao dispositivo ou sistema que ele controla. Diferente de computadores de propósito geral, como os computadores pessoais, um sistema embarcado realiza um conjunto de tarefas predefinidas, geralmente com requisitos específicos. Já que o sistema é dedicado a tarefas específicas, é possível reduzir o tamanho e os recursos computacionais, assim, diminuindo seu custo [1].

Existe uma ampla variedade de sistemas embarcados. Eles estão presentes no nosso dia-a-dia e apresentam-se nas mais diferentes formas e tarefas atribuídas. Por exemplo, um DVD player é um sistema embarcado dedicado a reprodução de filmes. MP3

players, muito difundidos atualmente, são dedicados a reprodução de músicas. Telefones celulares são sistemas embarcados que podem ser encontrados em uma ampla variedade de modelos com os mais variados recursos.

Existem tipos de sistemas embarcados que operam em condições especiais e necessitam ser gerenciáveis remotamente. Por exemplo, os sistemas embarcados dedicados à comunicação de dados, como os roteadores, que são amplamente usados na interconexão de redes, precisam ser configuráveis e monitoráveis remotamente. Ao contrário de sistemas onde o usuário tem acesso físico ao dispositivo, muitas vezes roteadores estão presentes em locais distantes e de difícil acesso. Além disso, uma rede pode ter um número elevado destes dispositivos espalhados por uma área geograficamente extensa. Partindo-se deste pressuposto, faz-se necessário uma forma de gerenciamento remoto deste tipo de equipamento.

Com o crescente aumento desta tecnologia, cria-se a necessidade de gerenciar diversos tipos de sistemas embarcados, à medida que tais sistemas passam a ser integrados às redes existentes. Basicamente, basta adicionar uma interface de rede e os protocolos adequados aos sistemas embarcados para viabilizar o gerenciamento via rede.

Entende-se por gerenciamento, as atividades de controle e monitoramento dos elementos da rede. As tarefas básicas do gerenciamento de redes, simplificada, são realizar configurações e obter informações, podendo utilizar estas informações para diagnosticar falhas e monitorar desempenho [2]. Para cumprir estes objetivos, funções de gerenciamento devem ser embutidas nos diversos componentes de uma rede que, no contexto desta oficina, são os dispositivos embarcados.

Normalmente, o gerenciamento de sistemas embarcados baseado em Linux é realizado através de um conjunto de protocolos, tais como SNMP, HTTP e SSH [3]. O protocolo SNMP (Simple Network Management Protocol) é o padrão de fato para gerenciamento de redes. SSH (Secure Shell) permite o acesso à linha de comando do equipamento por um canal criptografado, tornando possível sua administração. Existem também outras formas de gerenciamento via Web, utilizando HTTP. Este último é bastante utilizado pois permite a utilização do navegador para acessar uma interface de configuração no equipamento, que normalmente é mais amigável que uma sessão SSH.

2.3. Metodologia utilizada

A oficina será conduzida de forma que os conceitos teóricos serão seguidos de atividades práticas. Assim, durante o decorrer da oficina, o aluno terá a oportunidade de construir passo-a-passo um sistema embarcado completo, com funcionalidades de comunicação e gerenciamento.

Primeiramente, serão apresentados conceitos básicos envolvendo sistemas embarcados, bem como sua contextualização e utilização. Serão apresentados exemplos de hardware para sistemas embarcados e os componentes básicos de software necessários para embarcar um sistema Linux. Neste ponto, o aluno terá uma visão geral do funcionamento de um sistema embarcado.

Em seguida, serão apresentados os primeiros passos no desenvolvimento de um sistema embarcado baseado em Linux, que envolve a obtenção, configuração e compilação do kernel. Neste ponto, o aluno irá vislumbrar detalhes e tomar decisões sobre elementos que irão fazer parte do sistema final. Como por exemplo, seleção de drivers de dispositivos, sistemas de arquivos ou mesmo protocolos de rede, cujo o kernel deverá suportar.

No passo seguinte, o aluno descobrirá como construir um sistema de arquivos e quais as ferramentas básicas que tal sistema deverá conter para permitir a inicialização do sistema operacional. Será discutido o que é um sistema de arquivos, porque o kernel precisa dele e como ele dever ser construído. O aluno será instruído a usar ferramentas para gerar um sistema de arquivos personalizado.

Após a geração do sistema de arquivos, o aluno será guiado na compilação e instalação dos serviços de gerenciamento. Estes serviços possibilitarão o gerenciamento do sistema final, que será capaz de responder aos protocolos SNMP, HTTP e SSH. A compilação e instalação dos serviços será realizada utilizando o sistema Buildroot, que disponibiliza uma lista de pacotes que podem ser selecionados para instalação no sistema de arquivos.

Com posse do kernel do Linux compilado e do sistema de arquivos, será dado foco ao sistema de virtualização QEMU. Com tal sistema, o aluno será capaz de emular um hardware embarcado, e assim, poder ver a execução do sistema recém construído. O QEMU, assim como outros sistemas de virtualização, permite compartilhamento dos recursos de hardware com o sistema emulado. Desta forma, o dispositivo de rede da máquina hospedeira, será usado como dispositivo de comunicação pelo sistema embarcado emulado.

Finalmente, o aluno possuirá um sistema embarcado rodando e com capacidade de comunicação. Assim, a oficina seguirá com a configuração de uma rede entre os sistemas embarcados dos participantes. Por fim, será mostrado como configurar e utilizar os serviços de gerenciamento. A fim de demonstração, será utilizada uma ferramenta de descoberta de recursos via SNMP para mapear a rede de dispositivos embarcados criada pelos alunos.

2.4. Estrutura física e softwares necessários

Os recursos necessários para a realização da oficina são: uma sala com projetor e computadores para os alunos executando o sistema operacional Linux, de preferência na distribuição Ubuntu, ou outra qualquer baseada em Debian. Também é necessário que esses computadores tenham acesso à rede.

2.5. Quantidade de vagas

O número máximo de vagas deve ser limitado apenas ao número de computadores no laboratório. O ideal é que se tenha um computador por aluno.

3. Bibliografia principal e/ou básica utilizada na preparação da oficina prática

[1] Sistema embarcado. http://pt.wikipedia.org/wiki/Sistema_embarcado. Acessado em setembro de 2010.

[2] ISO/IEC 10040:1998 - Information technology - Open Systems Interconnection - Systems management overview.

[3] Yaghmour, Karim. Building Embedded Linux Systems, 2003.

4. Currículo do(s) autor(es).

Marcelo Veiga Neves é formado em Ciência da Computação pela Universidade Federal de Santa Maria (UFSM) e mestre pela Universidade Federal do Rio Grande do Sul (UFRGS). Atualmente, está cursando doutorado na área de Processamento de Alto Desempenho na Pontifícia Universidade do Rio Grande do Sul (PUCRS). Possui experiência em projeto e desenvolvendo de software para sistemas embarcados (por exemplo, adaptando o kernel do Linux para novas plataformas, escrevendo drivers de dispositivos, protocolos de rede e programas diversos para sistemas embarcados, etc.). Trabalhou no desenvolvimento de produtos para telecomunicações na empresa Digitel, tendo acumulado experiência em projeto, depuração e otimização de protocolos L2/L3.

Carlos Roberto Moratelli é graduado na Universidade Estadual do Oeste do Paraná (UNIOESTE) em 2004, no curso de Informática. Com mestrado em Ciências da Computação, na subárea de Microeletrônica pela Universidade Federal do Rio Grande do Sul (UFRGS). Atualmente, trabalhando como projetista de software na Digitel, onde atua na área de especificação e desenvolvimento de equipamentos para telecomunicações com experiência em tecnologias ethernet e IP. Trabalhou no desenvolvimento de software para roteadores 3G, incluindo experiência com protocolos PPP, Ethernet, DNS, IPSEC, GRE, SNMP, BGP, RIP, TFTP entre outros. Atuação em projeto de switches gerenciáveis, adquirindo conhecimento em tecnologias como VLAN, EFM, CFM, LACP, IGMP, RADIUS, 802.1x entre outros. Ampla experiência com Kernel Linux incluindo subsistema de Network. Anteriormente, atuou na Hewlett-Packard (HP Brasil) onde ajudou no desenvolvimento e manutenção de sistemas bancários baseados em linguagem C++. No mestrado, desenvolveu sua dissertação sob o título: *Estudo de técnicas para obtenção de hardware criptográfico tolerante a falhas* com orientação do Prof. Dr. Marcelo Soares Lubaszewski. Tal estudo propôs o desenvolvimento de sistemas embarcados criptográficos mais seguros com a adoção de técnicas de tolerância a falhas em processadores criptográficos. O trabalho de término de graduação foi escrito sob o título: *Desenvolvimento de um protótipo de robô para futebol de robôs* com orientação do Prof. Francisco Sérgio Sambatt, resultando em um robô controlado remotamente e capaz de atuar sobre as regras da RoboCup. Ainda na graduação foi bolsista CNPq trabalhando no desenvolvimento de software para a avaliação de sistemas de irrigação.