Snow666

EDITORA O'REILLY P.24
Os pensamentos mais polêmicos do editor-chefe da editora

CEZAR TAURION P.26
Combate à pirataria e a cópias não autorizadas

COLUNA DO MADDOG P.27
Direitos autorais interferem na preservação de músicas

LINUX NEW MEDIA
The Pulse of Open Source

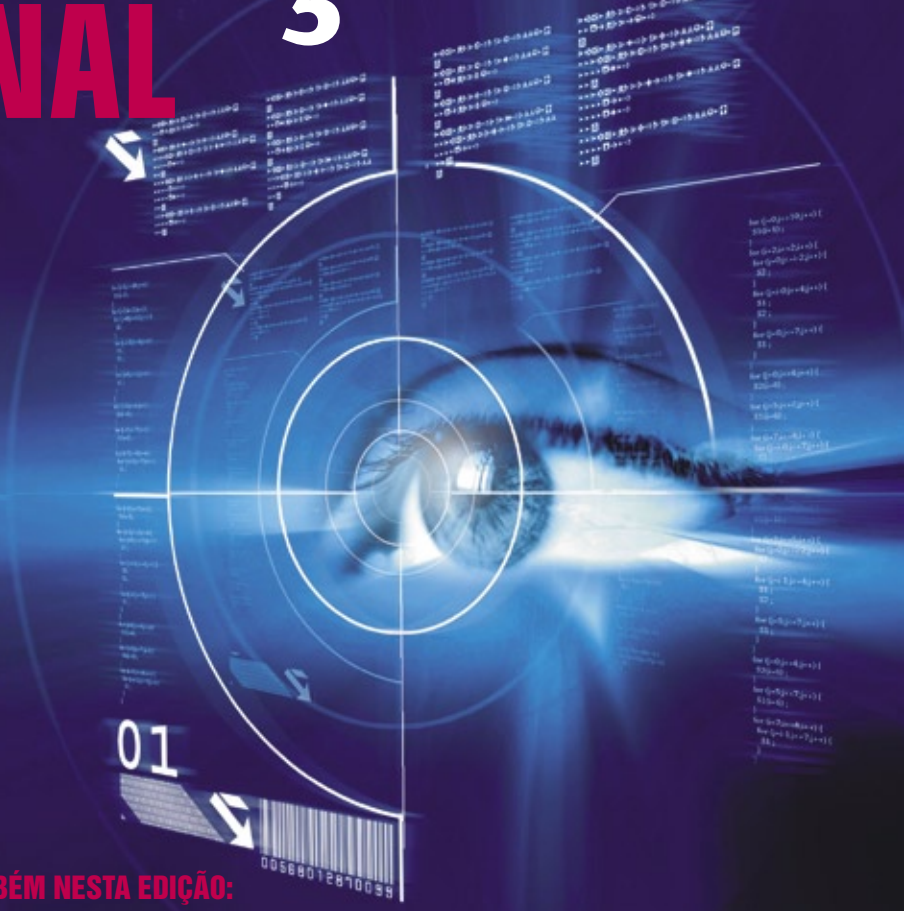
47 Outubro 2008

LINUX MAGAZINE

A REVISTA DO PROFISSIONAL DE TI

SEGURANÇA PROFISSIONAL

NA ERA DA INSEGURANÇA DA INFORMAÇÃO É FUNDAMENTAL MANTER SUA REDE A SALVO DE INVASORES p.33



SEGURANÇA: ROOTKIT VIRTUAL p.62

A virtualização nem sempre nos ajuda. Rootkits virtuais passam inteiramente despercebidos e tomam conta do sistema

REDES: CLUSTER APACHE p.58

Alta disponibilidade no Apache com seus próprios recursos internos

VEJA TAMBÉM NESTA EDIÇÃO:

9 771806 942009 00047
#47 10/08
R$ 13,90
€ 7,50

© Linux New Media do Brasil Editora Ltda.

Snow666

Open Energy™

Open Access

Acesso a componentes de software validados e testados pela Bull

1

Open Service

Desenvolve e gerencia Projetos de Software Livre utilizando o ferramental Bull de Fábrica de Sistemas

3

2

Substitui suas tecnologias existentes nos atuais ambientes de desenvolvimento por alternativas de Software Livre

Open Exchange

4

Implementa processos empresariais utilizando com total segurança soluções de Software Livre

Open Enterprise

Nós implementamos um modelo industrial para o mundo do Software Livre

"Open Energy", a familia Bull de Serviços para Software Livre. Nossas soluções respondem a todas as necessidades para o desenvolvimento, integração, interoperabilidade e manutenção de sistemas requeridas por todos os tipos de organizações que tomam o rumo do Software Livre. Estabelecida sobre os fortes alicerces da ampla infraestrutura Bull de Integração, Serviços e Centros de Competência Internacionais, a "Open Energy" lhe da acesso aos melhores especialistas e comunidades de desenvolvimento.

BULL

Architect of an Open World™

www.bull.com

© Linux New Media do Brasil Editora Ltda.

Expediente editorial

Diretor Geral
Rafael Peregrino da Silva
rperegrino@linuxmagazine.com.br

Editor
Pablo Hess
phess@linuxmagazine.com.br

Revisão
Aileen Otomi Nakamura
anakamura@linuxmagazine.com.br

Editora de Arte
Paola Viveiros
pviveiros@linuxmagazine.com.br

Centros de Competência
Centro de Competência em Software:
Oliver Frommel: ofrommel@linuxnewmedia.de
Kristian Kißling: kkissling@linuxnewmedia.de
Peter Kreussel: pkreussel@linuxnewmedia.de
Marcel Hilzinger: hilzinger@linuxnewmedia.de
Centro de Competência em Redes e Segurança:
Achim Leitner: aleitner@linuxnewmedia.de
Jens-Christoph B.: jbrendel@linuxnewmedia.de
Hans-Georg Eßer: hgesser@linuxnewmedia.de
Thomas Leichtenstern: tleichtenstern@linuxnewmedia.de
Max Werner: mwerner@linuxnewmedia.de
Markus Feilner: mfeilner@linuxnewmedia.de
Nils Magnus: nmagnus@linuxnewmedia.de

Anúncios:
Rafael Peregrino da Silva (Brasil)
anuncios@linuxmagazine.com.br
Tel.: +55 (0)11 4082 1300
Fax: +55 (0)11 4082 1302

Petra Jaser (Alemanha, Áustria e Suíça)
anzeigen@linuxnewmedia.de

Penny Wilby (Reino Unido e Irlanda)
pwilby@linux-magazine.com

Amy Phalen (Estados Unidos)
aphalen@linuxmagazine.com

Hubert Wiest (Outros países)
hwiest@linuxnewmedia.de

Gerente de Circulação
Claudio Bazzoli
cbazzoli@linuxmagazine.com.br

Na Internet:
www.linuxmagazine.com.br – Brasil
www.linux-magazin.de – Alemanha
www.linux-magazine.com – Portal Mundial
www.linuxmagazine.com.au – Austrália
www.linux-magazine.ca – Canadá
www.linux-magazine.es – Espanha
www.linux-magazine.pl – Polônia
www.linux-magazine.co.uk – Reino Unido
www.linux-magazin.ro – Romênia

Apesar de todos os cuidados possíveis terem sido tomados durante a produção desta revista, a editora não é responsável por eventuais imprecisões nela contidas ou por conseqüências que advenham de seu uso. A utilização de qualquer material da revista ocorre por conta e risco do leitor.

Nenhum material pode ser reproduzido em qualquer meio, em parte ou no todo, sem permissão expressa da editora. Assume-se que qualquer correspondência recebida, tal como cartas, emails, faxes, fotografias, artigos e desenhos, sejam fornecidos para publicação ou licenciamento a terceiros de forma mundial não-exclusiva pela Linux New Media do Brasil, a menos que explicitamente indicado.

Linux é uma marca registrada de Linus Torvalds.

Linux Magazine é publicada mensalmente por:

Linux New Media do Brasil Editora Ltda.
Av. Fagundes Filho, 134
Conj. 53 – Saúde
04304-000 – São Paulo – SP – Brasil
Tel.: +55 (0)11 4082 1300 – Fax: +55 (0)11 4082 1302

Direitos Autorais e Marcas Registradas © 2004 - 2008:
Linux New Media do Brasil Editora Ltda.

Impressão e Acabamento: Parma
Distribuída em todo o país pela Dinap S.A.,
Distribuidora Nacional de Publicações, São Paulo.

Atendimento Assinante

www.linuxnewmedia.com.br/atendimento
São Paulo: +55 (0)11 3512 9460
Rio de Janeiro: +55 (0)21 3512 0888
Belo Horizonte: +55 (0)31 3516 1280

ISSN 1806-9428 **Impresso no Brasil**

INSTITUTO VERIFICADOR DE CIRCULAÇÃO

Comunidade como valor

Caros leitores,

Surgidos a partir de movimentos estritamente comunitários, o projeto GNU e o kernel Linux alcançaram grande importância nas empresas, como todos bem sabemos. Corporações de diversos tamanhos começaram a usar o sistema de código aberto ou apenas alguns softwares, naturalmente como simples usuárias a princípio. As comunidades de cada software eram tudo, nessa época: gestoras, desenvolvedoras e promotoras.

Já na década atual, observamos empresas oferecendo, elas próprias, soluções de código aberto — algumas abertas desde o início, outras transformadas em Software Livre. Participaram desse processo as poucas empresas que compreendiam verdadeiramente as vantagens do desenvolvimento colaborativo. O mercado do SL/CA alcançava um novo patamar de maturidade. O Open Source Development Lab (OSDL) reuniu diversas empresas para financiar e gerir o desenvolvimento do kernel Linux. A comunidade era tratada como algo desvinculado de qualquer uma das empresas envolvidas com dado software e parecia perder importância frente aos recursos financeiros e humanos cedidos pelas companhias.

No entanto, sempre se pode amadurecer um pouco mais. As recentes ações da Sun na direção de formar e manter uma comunidade em torno de seus produtos de código aberto têm significado marcante. A empresa é de fato uma das que mais cedem linhas de código ao Código Aberto como um todo, o que teoricamente bastaria para torná-la uma potência no mercado. Entretanto, falta-lhe uma comunidade capaz de promover o uso de todo esse código contribuído — falta-lhe uma presença significativa na comunidade.

A comunidade dos projetos de SL/CA, atualmente, é um de seus valores. Pouco adianta uma empresa abrir o código de um projeto se ele não for abraçado pela comunidade. Somente formando uma comunidade é possível desfrutar dos benefícios do desenvolvimento colaborativo.

Associe sua empresa à comunidade dos softwares que ela usa. Isso aumenta o valor do software.

Pablo Hess
Editor

© Linux New Media do Brasil Editora Ltda.

Snow666

CAPA

© Linux New Media do Brasil Editora Ltda.

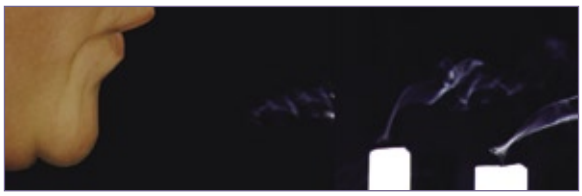
COLUNAS

NOTÍCIAS

CORPORATE

TUTORIAL

ANÁLISE



REDES

SEGURANÇA

PROGRAMAÇÃO

SERVIÇOS

© Linux New Media do Brasil Editora Ltda.

Emails para o editor

Permissão de Escrita

Se você tem dúvidas sobre o mundo Linux, críticas ou sugestões que possam ajudar a melhorar a nossa revista, escreva para o seguinte endereço: ***cartas@linuxmagazine.com.br****. Devido ao grande volume de correspondência, torna-se impossível responder a todas as dúvidas sobre aplicativos, configurações e problemas de hardware que chegam à Redação, mas garantimos que elas são lidas e analisadas. As mais interessantes são publicadas nesta seção.*

Matérias sobre Moodle

Amigos da Linux Magazine, tornei-me um leitor assíduo da revista Linux Magazine e a partir daí meu conhecimento e familiaridade com Linux e sistemas abertos em geral foram alavancados, passando a um patamar superior.

Gostaria de saber se está nos planos da editoria da revista um conjunto de matérias sobre os sistemas LMS, em particular o Moodle, que uso bastante. Não seria oportuno esse tipo de matéria?

Herli Menezes

Resposta

Prezado Herli, em primeiro lugar agradeço pela sugestão. Para nós, é fundamental essa participação de nossos leitores para sabermos quais assuntos são mais interessantes.

Por favor fique à vontade para sugerir autores para essas matérias, caso você conheça algum. Como dissemos em http://www.linuxmagazine.com.br/autores*, se um assunto é interessante para você, provavelmente também é interessante para muitos outros leitores.* ■

LPI

Quero agradecer à Linux Magazine pelo excelente trabalho que faz em prol do software livre. A revista é muito profissional e para mim é a melhor revista de TI no Brasil. Outras revistas são meramente comerciais, ou seja, são repletas de propagandas com pouca ênfase técnica em TI.

Consegui a Certificação LPI nível 1 em 25/08/2008. A coleção da Linux Magazine para a certificação LPI nível 1 foi essencial para o meu sucesso, pois tem informações pertinentes e um roteiro excelente.

Quero agradecer também ao Guia do Hardware, Guia Foca Linux, Guia do Linux e outras fontes de documentação disponíveis na Internet e confeccionadas pela comunidade.

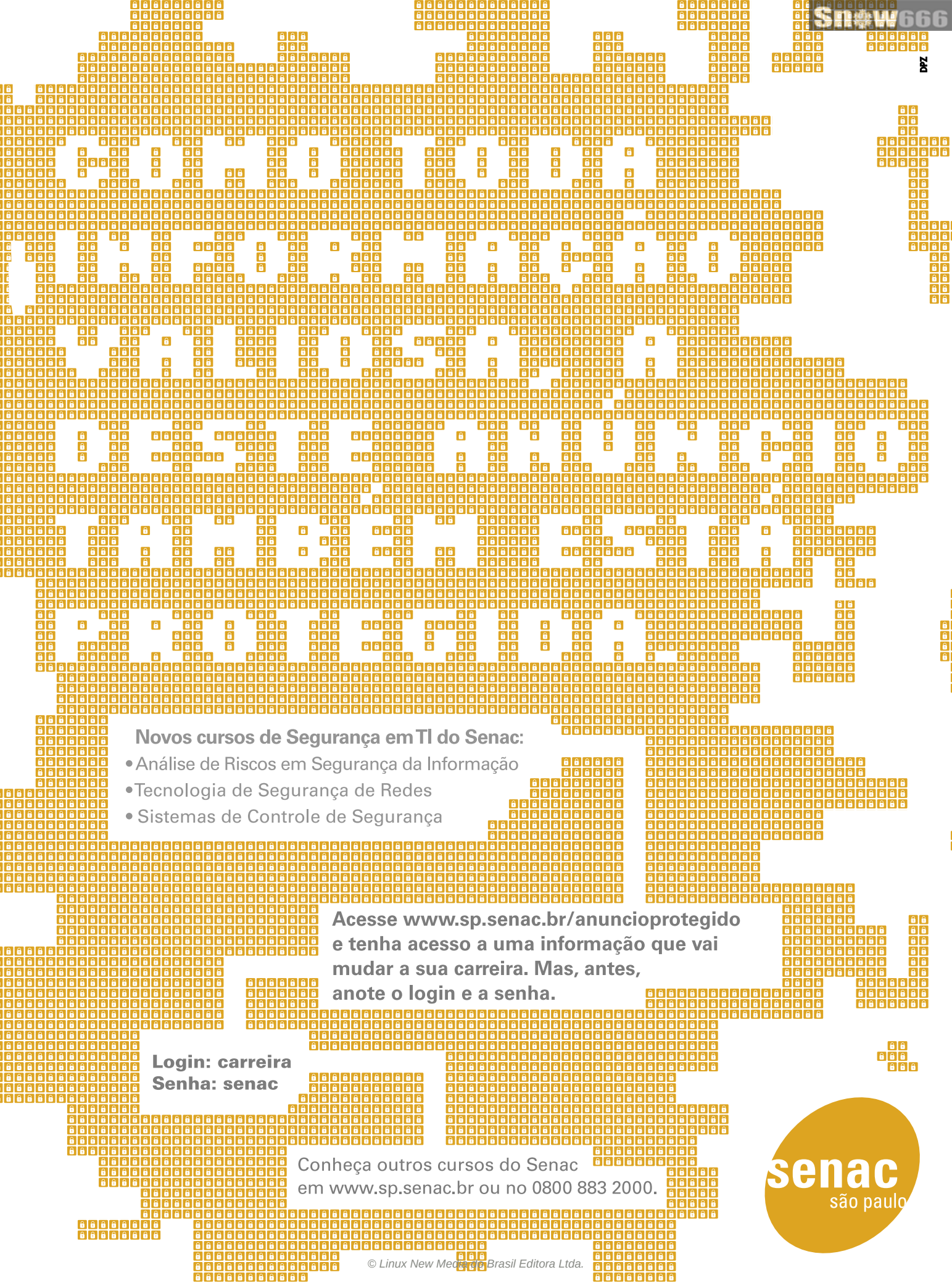
Muito obrigado!

Wanderson

Resposta

Caro Wanderson, nós que agradecemos por seu depoimento. Parabéns pelo sucesso e que seu futuro como profissional certificado pelo LPI seja brilhante. ■

© Linux New Media do Brasil Editora Ltda.



© Linux New Mediatrade Brasil Editora Ltda.

Snow666

COLUNA

Pergunte ao Klaus!

Klaus Knopper

Discos e telas podem ser difíceis de redimensionar.

por Klaus Knopper

Partições por disco

Klaus, você poderia me ajudar com a súbita mudança (em distribuições recentes) do número máximo de partições por disco, de 64 para 15?

Acabei de tentar instalar o Mythbuntu e ele simplesmente se recusou a formatar a `sdb19`. Meu sistema tem aproximadamente 30 partições em cada disco interno. Os sistemas de produção estão todos nas partições mais altas. O problema parece ser causado pela *libata* e a solução a longo prazo significa recomeçar tudo com um sistema virtualizado. Além disso, suspeito que, na prática, o mesmo se aplique ao LVM.

Você tem alguma sugestão para me ajudar a curto prazo?

Resposta

Algumas possibilidades:

- É uma gambiarra, mas é o método mais rápido: carregue o módulo `loop.ko` com `max_loop=32` e acesse as partições em `/dev/sda` com os parâmetros do `losetup` desvio (`-o`) e limite de tamanho (`-s`). Suas partições montáveis ficarão em `/dev/loop*`. Obviamente, você pode restringir isso a 16 partições, se preferir. O problema é que você precisará calcular os desvios (*offsets*) cuidadosamente pela análise da tabela de partições, além de ter cuidado para não sobrescrever partições consecutivas em caso de formatação (por isso o parâmetro de limite de tamanho).
- Voltar para IDE.
- Usar o LVM em vez de partições reais.
- Aplicar um patch no kernel.
- Para discos USB, um recurso de "usar alocação dinâmica de *minors*" permite que o *Udev* crie novos dispositivos com IDs *minor* conforme sejam necessários. Porém, acho que isso não afeta discos SCSI e SATA. Talvez seja possível ativar isso para SATA em kernels futuros também.

O uso do LVM em vez de partições provavelmente é a solução mais limpa a longo prazo, mas requer a recriação de partições, assim como um becape de todos os dados para restauração posterior.

Laptop widescreen

Acabei de comprar um Lenovo T61 com Ubuntu 8.04. A tela widescreen é ótima. O único detalhe incômodo é que como os projetores que uso ao visitar empresas não são widescreen, quando mudo a resolução para 1024x768 ou outras não-wide, a tela do laptop fica com o tamanho pedido, mas o papel de parede fica enorme. Quando uso o *Impress* do OpenOffice.org no modo de apresentação de slides, ele preenche a tela wide e o projetor mostra só uma parte da imagem, apesar de o laptop estar configurado para 1024x768.

Como consigo fazer o laptop funcionar adequadamente com projetores não-wide?

Resposta

Notebooks e projetores sempre foram um desastre. O problema que você descreveu é até um dos menores. Na verdade, é um problema de hardware: exibir uma imagem na tela do computador e no dispositivo externo ao mesmo tempo parece ser uma tarefa muito difícil para os fabricantes de chipsets.

Descobri que em vários notebooks pequenos – especificamente naqueles com baixas resoluções – a forma mais rápida e eficaz para chegar ao objetivo é desligar a tela do laptop enquanto se ajusta a resolução para um projetor externo.

O programa *Xrandr* pode ajudar nessa tarefa. Você pode executá-lo por meio de um script que, ao toque de uma tecla, reative a tela do laptop em sua resolução ideal. Por exemplo:

```
# Define o modo do projetor
xrandr --output LVDS --off
xrandr --output VGA --auto
read -p "Pressione ENTER para voltar à tela
➥ do laptop"
xrandr --output VGA --off
xrandr --output LVDS --auto
```

Ao executar esse script, a tela do laptop vai se apagar, mas retornará após o **[Enter]** ser pressionado. ■

© Linux New Media do Brasil Editora Ltda.

Snow666

© [illegible] do Brasil Editora Ltda.

COLUNA

Autenticação com pacote único

Charly Kühnast

O port knocking convencional é sujeito a sniffing e ataques de força bruta. É mais seguro e moderno enviar um pacote criptografado com uma requisição de acesso ao servidor.

por Charly Kühnast

A técnica convencional de *port knocking*, descrita na minha última coluna [1], protege o servidor contra agressores que rotineiramente verificam redes inteiras em busca de "frutos quase maduros". Um cracker mais dedicado e que registre as comunicações também consegue identificar sinais de batidas, pois as seqüências se repetem.

Em teoria, pode-se considerar o uso de listas de sinais de batida que mudem ao longo do tempo, tornando-se obsoletas logo após seu uso. Infelizmente, isso é muito complexo. Além disso, se o administrador não for suficientemente criativo, um agressor poderia simplesmente experimentar seqüências populares de batidas (7000, 8000, 9000...) para ganhar acesso.

A *Single-Packet Authentication* (autenticação por pacote único, ou SPA) é uma possível solução. O sistema cliente envia um único pacote contendo as credenciais de autenticação criptografadas – geralmente uma senha – e a requisição de abertura de uma porta específica. Uma implementação de SPA altamente funcional é o *Firewall Knock Operator*, ou *Fwknop*[2].

Além das ferramentar normais de compilação, a instalação do Fwknop requer o *Perl*, o pacote *libpcap-dev* e o módulo do CPAN *Net::Pcap*. Depois de instalar todos esses recursos, instalar o Fwknop é muito fácil, graças ao instalador em Perl.

Combinação

O Fwknop inclui o servidor `fwknopd` e o cliente `fwknop`. O servidor é configurado editando-se dois arquivos em `/etc/gwknop/`; o arquivo `fwknop.conf` contém a configuração básica. Inicialmente, será necessário apenas alterar alguns parâmetros, marcados com `__CHANGEME__`.

Os outros pontos ajustáveis possuem padrões bastante competentes. Note que é preciso sincronizar a hora do servidor e do cliente, pois, se a diferença for grande demais, o `fwknopd` vai ignorar o cliente.

As entradas em `/etc/fwknop/access.conf` definem como o `fwknopd` responde às batidas do cliente. A chave secreta usada pelo cliente para se identificar fica armazenada aqui. A linha `SOURCE` pode ser usada para restringir as redes a partir das quais o daemon aceita batidas. Para definir a porta que o sistema abre após as batidas – por exemplo `tcp/22` – deve-se usar a linha `OPEN_PORTS`. A **figura 1** mostra uma tentativa com sucesso. O cliente `fwknop` obtém a chave a partir de seu próprio arquivo `/etc/fwknop/access.conf`.

Se a conexão SSH não for aberta com a rapidez necessária, é acionado o `FW_ACCESS_TIMEOUT`. Esse valor desse tempo geralmente é de 30 segundos, mas o dobro disso pode ser mais indicado – nunca apresse um administrador de sistemas em seu trabalho! ■

```
charly@funghi:~$ fwknop -A tcp/22 -a 10.254.75.80 -k 10.254.75.80

[+] Starting fwknop client (SPA mode)...
[+] Enter an encryption key. This key must match a key in the file
    /etc/fwknop/access.conf on the remote system.

Encryption Key:

[+] Building encrypted Single Packet Authorization (SPA) message...
[+] Packet fields:

        Random data:    7842749886485723
        Username:       charly
        Timestamp:      1214918141
        Version:        1.9.5
        Type:           1 (access mode)
        Access:         10.254.75.80,tcp/22
        SHA256 digest:  evcTZFKgUbkIk/Nm28LaJwFInmIb/ENfTFiTooK5TIA

[+] Sending 182 byte message to 10.254.75.80 over udp/62201...

charly@funghi:~$
```

Figura 1 O cliente que está batendo na porta 22 pode entrar porque possui a chave certa.

Mais informações

[1] Charly Kühnast, "Knockd", na Linux Magazine 46: http://www.linuxmagazine.com.br/article/2151

[2] Fwknop: http://www.cipherdyne.org/fwknop/

Sobre o autor

Charly Kühnast é administrador de sistemas Unix no datacenter Moers, perto do famoso rio Reno, na Alemanha. Lá ele cuida, principalmente, dos firewalls.

© Linux New Media do Brasil Editora Ltda.

DPZ

Novo Librix Desktop 3.0, agora em 3D. É só instalar e sair trabalhando.

A liberdade do novo Librix Desktop 3.0 não termina no código aberto. Oferece programas cada vez mais compatíveis e comumente usados em sistemas proprietários, além de ter total compatibilidade com os equipamentos padrão de mercado. Sua plataforma está mais moderna, amigável e o sistema totalmente configurável na instalação. Assim como todo produto Itautec, já foi testado, aprovado e tem assistência técnica em todo o país. Liberdade, interoperabilidade e facilidade de uso, isso é TI. Isso é Tecnologia Itautec.

Acesse www.itautecshop.com.br ou ligue 0800 121 444.

COMPROMISSO COM A SUSTENTABILIDADE

Itautec

© Linux New Media do Brasil Editora Ltda.

Snow999

Snow666

COLUNA

Crônicas do kernel

Zack Brown

O ZFS continua causando picos de ansiedade entre os desenvolvedores.

por Zack Brown

ZFS no kernel?

A Sun liberou uma parte do código-fonte de seu sistema de arquivos ZFS sob a GPL, levando à questão: por que não incluí-lo no kernel? O código é um sistema de arquivos de 128 bits que distribui os dados armazenados ao longo de múltiplos dispositivos de bloco, fazendo checksums de todos os dados para uma melhor detecção de erros, além de manter um histórico de *snapshots* de todas as alterações numa árvore de diretórios. O ZFS havia sido liberado antes sob a CDDL, incompatível com a GPL.

A nova liberação deixou alguns desenvolvedores do kernel bastante entusiasmados, até perceberem que a versão liberada pela Sun era somente leitura e não tinha boa parte do código da versão liberada sob a CDDL.

Além disso, como Alan Cox lembrou, a Sun e a NetApp estão numa disputa legal sobre as patentes envolvidas no ZFS. Tentar incluir esses trechos patenteados no kernel exigiria permissão de ambos os combatentes, o que parece pouco provável.

Alan ainda especulou: "Só consigo interpretar as motivações da Sun de uma maneira – eles querem parecer abertos, mas sabem que o ZFS talvez seja o único aspecto que pode salvar o Solaris como produto no datacenter, então não estão realmente preparados para deixar Linus usá-lo".

Ricardo Correia também está reescrevendo o ZFS para o FUSE, e Patrick Draper relatou sucesso completo com ele até o momento (embora recomende manter becapes, assim como com qualquer sistema de arquivos). Enquanto isso, Christoph Hellwig afirma que somente leitura é melhor do que nada, e se a Sun liberou o ZFS somente para leitura, seria ótimo portar esse código para o Linux e incorporá-lo à árvore principal.

Novos drivers

Michael Buesch submeteu um driver para permitir que todos os chips Brooktree 8xx exportem todos os 24 pinos GPIO (*General Purpose Input/Output*) para a infra-estrutura GPIO do kernel. Isso permite que cada pino seja usado tanto para entrada quanto para saída, conforme desejado. Michael não conseguiu encontrar um mantenedor do GPIO para enviar esse patch, mas Andrew Morton disse que David Brownell era a pessoa certa.

David Altobelli escreveu um patch para suportar o processador de gerenciamento HP *iLO/iLO2*. Esse hardware permite que administradores controlem seus servidores sem precisarem do console. O driver de David permite que programas de espaço do usuário façam interface com o servidor e executem comandos. Andrew Morton ofereceu comentários técnicos bastante significativos, apontando várias falhas. Ele também mostrou que esse é essencialmente um driver somente para a arquitetura *x86*. Ainda não está claro se suas objeções são suficientemente sérias para pararem o projeto ou impedir sua aceitação no kernel. Geralmente os tipos de objeções técnicas levantadas por Andrew tendem a indicar uma disposição geral para considerar o patch.

Karsten Keil escreveu e submeteu um novo driver *mISDN*. Ele disse que o objetivo final desse driver seria substituir a arquitetura de drivers *I4L* para placas ISDN passivas. Ingo Molnár mostrou algumas questões técnicas, mas Tilman Schmidt disse que, como um dos mantenedores dos velhos drivers I4L que seriam substituídos pelo trabalho de Karsten, ele estava muito satisfeito em ver o mISDN avançar. Ele não conseguiu entender o substituto rapidamente e pediu uma documentação semelhante à que já existe para o I4L. ■

Sobre o autor

A lista de discussão *Linux-kernel* é o núcleo das atividades de desenvolvimento do kernel. **Zack Brown** consegue se perder nesse oceano de mensagens e extrair significado! Sua *newsletter Kernel Traffic* esteve em atividade de 1999 a 2005.

© Linux New Media do Brasil Editora Ltda.

Snow666

A Xandros é líder em inovação. De desktops a servidores e de groupware ao licenciamento de sistemas operacionais, a Xandros dita o ritmo da tecnologia com uma rede global e suas inovações em Netbooks, Moblin, utilização do processador e consumo de energia. A Xandros está redefinindo a forma como são feitos PCs e dispositivos móveis, e adaptando-se ao nosso mundo em constante mudança.

Entre em contato com a Xandros hoje para descobrir como a sua empresa pode fazer as nossas inovações funcionarem a seu favor.

xandros

jamesl@xandros.com | www.xandros.com.br
646-747-7640 | 0800-891-6799

© Linux New Media do Brasil Editora Ltda.

Proteja seu site e seus clientes

Insegurança

Aprenda mais sobre como proteger seu site com o NoScript, ModSecurity e Site Security Policy.

por Kurt Seifried

Naomi Austin – www.sxc.hu

Além de muitas falhas de segurança, a rede mundial de computadores apresenta dois conjuntos de problemas bem distintos. Por um lado, a maioria de nós usa o navegador web regularmente, e deseja evitar que os clientes web executem códigos de agressores, o que permitiria que estes se apoderassem de nossas máquinas. No outro lado estão os servidores web, que não desejamos que sejam comprometidos ou sofram ataques constantes (XSS, injeção SQL etc.). Então, qual é a resposta?

Bem, não há uma única resposta. São necessários alguns passos para proteger tanto os clientes quanto os servidores, pois não importa o grau da sua atenção à segurança, você vai interagir com servidores ou clientes menos seguros que os seus.

Figura 1 O *NoScript* bloqueia *JavaScript* por padrão e disponibiliza uma lista de opções.

JavaScript e NoScript

No navegador web Firefox, de 196 alertas de segurança, 62 listavam como solução temporária a desativação do *JavaScript*. Além disso, as vulnerabilidades baseadas em JavaScript tendem a ser aquelas que permitem a execução de código arbitrário, então qualquer medida de segurança pró-ativa que lide com elas terá um impacto significativo.

Melhorar a segurança de clientes web contra ataques é relativamente fácil; no entanto, alguns sites podem não funcionar corretamente. Desativar o JavaScript inteiramente é uma opção, mas muitos sites hoje em dia necessitam desse recurso para apresentar seu conteúdo, formulários etc.

Uma abordagem mais granular está disponível por meio do plugin *NoScript*[1] para o Firefox. O padrão é bloquear a excução de JavaScript; depois, pode-se permitir que o JavaScript rode temporária ou permanentemente. Ou então, pode-se marcar permanentemente o site como não confiável para impedir a execução de qualquer código JavaScript nele (**figura 1**).

A maior desvantagem desse plugin é que é necessário prestar atenção à barra de informações que aparece na parte de baixo da tela quando é feito o bloqueio de algum JavaScript (**figura 2**) para decidir se o permite ou não. Em caso negativo, talvez você se depare com um site quase todo vazio, ou ainda com um formulário que não funciona corretamente.

Além disso, o NoScript possui alguma proteção contra XSS – URLs com caracteres como › geram um alerta e oferecem ao usuário a chance de bloquear seu carregamento.

Segurança do servidor

Como administrador de servidores, você não pode forçar seus clientes a serem seguros, mas pode proteger seu próprio servidor e suas aplicações web de ataques. Proteger seu servidor também pode impedir que clientes defeituosos ou usuários que tenham visitado sites hostis tomem ações que possam danificar suas contas ou os dados armazenados no seu site – por exemplo, de um ataque XSS que interaja com a conta do usuário para alterar sua senha nesse site.

Apache ModSecurity

Assim como vários projetos de segurança, o *ModSecurity* começou como um projeto de código aberto licenciado sob a GPLv2 com o objetivo de adicionar uma nova camada de segurança ao servidor web *Apache*[2]. O projeto parece ter sido comercializado com sucesso; porém, assim como muitos aplicativos de

© Linux New Media do Brasil Editora Ltda.

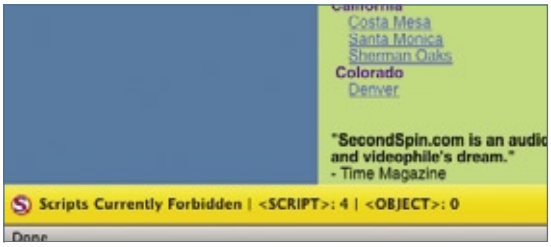


Figura 2 Uma barra de informações na parte de baixo da tela diz que o *JavaScript* foi bloqueado.

segurança de código aberto, ainda há versões gratuitas.

O principal benefício do ModSecurity é que ele pode ser usado para fornecer segurança a qualquer aplicação que esteja rodando no sistema. Porém, é preciso inserir um módulo personalizado no Apache – o que significa que é preciso ter controle do servidor – e possuir poder de processamento suficiente para lidar com o que esse módulo exige, o que pode ser significativo. O módulo ModSecurity permite que as requisições ao servidor sejam examinadas em vários estágios do processo: quando o cabeçalho de requisição é processado pela primeira vez, quando o corpo da requisição é processado, quando os cabeçalhos de resposta são criados, quando o corpo da resposta é processado e na fase de log.

Outra vantagem do ModSecurity é que ele suporta expressões regulares compatíveis com Perl (PCRE, na sigla) e as regras suportadas podem ocasionar várias ações, incluindo permitir ou bloquear uma requisição, fechar a conexão com um pacote *FIN* ou executar um programa externo. Isso permite, por exemplo, que os administradores do site filtrem caracteres como < e > das requisições – um provável indicador de ataque XSS – ou procurem informações pessoais tais como números de cartão de crédito de 12 dígitos em requisições de saída (por exemplo, disparados por um ataque de injeção SQL) e impeçam que sejam servidos dados ao agressor. Veja o **quadro 1**.

Como se pode imaginar, todo esse poder e flexibilidade têm o custo de complexidade; entretanto, isso é aliviado por causa de um poderoso conjunto padrão de regras disponível gratuitamente (licenciado sob a GPLv2), que pode ser usado como ponto de partida para a maioria dos sites.

Política de segurança de sites

A política de segurança de sites (*Site Security Policy*) é uma abordagem interessante que ainda se encontra em estágio inicial[3]. A idéia é que um servidor web hospede um arquivo que especifica como um cliente deve interagir com o servidor, evitando assim interações inseguras como ataques XSS ou de requisições cruzadas de sites. No lado do cliente, há suporte a esse padrão – ou embutido ou sob a forma de um plugin para o Firefox – que permite que o cliente baixe e examine o arquivo de política antes de interagir com o servidor web.

Um efeito colateral interessante dessa abordagem é a possibilidade de ter proxies web como o Squid com suporte a esse padrão, efetivamente protegendo todos os clientes web sob ele de ações potencialmente inseguras em sites que escolham suportar o padrão Site Security Policy.

Conclusão

A segurança web não tem uma solução simples: não importa o quanto tentemos, os malvados vão sempre ter servidores web hostis ou comprometer outros servidores web. No lado do cliente, a situação é basicamente um desastre. Se você usar Linux, no entanto, há poucas chances de você ser vitimado e muitas chances de você estar com todos os softwares atualizados, pois quase todas as distribuições fazem essas atualizações automaticamente por padrão.

Remendando os furos de segurança conforme sejam identificados e aplicando outras medidas de segurança – como o NoScript e o ModSecurity – é possível aumentar a probabilidade de manter servidores e clientes "saudáveis" como estão.

No final das contas, isso reduz o tempo e a energia gastos em limpezas repetitivas, que é o que todos querem, de qualquer forma. ■

Quadro 1: Exemplo de regra

Um exemplo simplista para detectar e bloquear qualquer número de 12 dígitos em páginas web de saída:

```
SecRule RESPONSE_BODY "[0-9]{12}" \
"phase:4,t:none,ctl:auditLogParts=+E,deny,log,auditlog,status:500,
➥ msg:'numero de 12 digitos',id:'1',tag:'LEAKAGE/
```

Mais informações

[1] NoScript: http://noscript.net/

[2] ModSecurity: http://www.modsecurity.org/

[3] Site Security Policy: http://people.mozilla.com/~bsterne/site-security-policy/

Sobre o autor

Kurt Seifried é consultor de segurança da informação especializado em redes e Linux desde 1996. Ele freqüentemente se pergunta como a tecnologia funciona em grande escala mas costuma falhar em pequena escala.

© Linux New Media do Brasil Editora Ltda.

Snow666

COLUNA

Mudando as regras do jogo

Augusto Campos

Quem sai ganhando com as trocas de plataformas livres?

por Augusto Campos

No início de setembro foi divulgado que uma das maiores instituições usuárias de código aberto no âmbito do Executivo Federal passaria a adotar e, eventualmente, até mesmo contribuir com uma distribuição Linux específica — o Debian —, o que naturalmente foi comemorado pelos demais usuários dessa distribuição e também pela comunidade brasileira de entusiastas do Linux.

Na data em que se divulgou amplamente na web brasileira a notícia desta adoção, começaram a surgir vozes diversas chamando a atenção para um fato conhecido, mas que acabou ficando de fora do comunicado que serviu como base para boa parte da cobertura nacional: a instituição já usava Linux há bastante tempo, com investimento na distribuição Fedora e bom número de equipamentos e redes já em produção com esse sistema.

> ***Não se trata de uma moléstia associada ao software de código aberto diretamente.***

Só que não se trata de uma moléstia associada ao software de código aberto diretamente. Quem atua no mercado de TI já ouviu grande número de casos de instituições que, em uma nova gestão ou após uma idéia do CIO, não amparado pelo seu corpo técnico, decidiram trocar subitamente seu fornecedor de bancos de dados, de hardware, de plataforma de desenvolvimento, de infra-estrutura de conectividade ou mesmo de sistema operacional.

Ver acontecer algo similar com uma plataforma livre, em que as diferenças de funcionalidade entre as alternativas existentes ocorrem de maneira bem específica, pode ser bastante educativo — especialmente nos casos em que há ilusão sobre a natureza da opção pelo Linux. Trata-se de uma decisão como qualquer outra, multifacetada e repleta de motivos adicionais, em qualquer instituição. Se for gerar retrabalho e custos evitáveis, como é o caso em muitas mudanças discricionárias de plataforma, é algo mais a ser considerado, mas, outra vez, só parece novidade se nos limitarmos a ver o quadro sob a perspectiva do software livre. Sob a perspectiva de plataformas de TI, esse é um filme velho.

Em uma época em que a doença original da adoção do Linux em instituições públicas brasileiras – que era a insistência em tentar criar e manter sua própria distribuição, em vez de adotar uma das boas opções disponíveis – já se aproxima da extinção, é o momento de ver surgir a possibilidade de outras moléstias e temer pela possível ocorrência de mudança e migração de distribuição ao sabor da preferência e conveniência dos gestores e de suas equipes, apontados para comandar o limitado número de instituições públicas que ativamente integram o código aberto em suas políticas de TI.

Ao mesmo tempo, está chegando o momento de procurar identificar os casos de sucesso de longa duração na adoção (e não da mera experiência na implantação) do Linux e de outros softwares livres em organizações públicas, em especial os casos em que a instituição chegou a contribuir algo de volta para os desenvolvedores ou mantenedores dos respectivos softwares. Alguém se habilita? ■

Sobre o autor

Augusto César Campos é administrador de TI e, desde 1996, mantém o site BR-linux.org, que cobre a cena do Software Livre no Brasil e no mundo.

© Linux New Media do Brasil Editora Ltda.

Snow666

TUDO SOBRE COMUNICAÇÃO IP

www.ipcomm2008.com.br

Agende!

IP COMMUNICATIONS BRASIL2008

2 a 4 de dezembro de 2008 - Centro de Convenções Rebouças - São Paulo - SP - Brasil

IPComm 2008

- Palestras - Cases - Debates - Tutoriais
- Voz - Video - TV.....over IP
- UC - Peering - Segurança - Gerenciamento
- Infraestrutura - Banda Larga

Soluções Open Source

digium | Asterisk

CommLogik Corporation

& outras

IP Expo

- Os melhores produtos e serviços para a Comunicação sobre IP

MÍDIA | LINUX MAGAZINE | Convergência DIGITAL

APOIO | AsteriskBrasil.org

REALIZAÇÃO | IP COMMUNICATIONS BRASIL

ORGANIZAÇÃO | Comunicação e Marketing

© Linux New Media do Brasil Editora Ltda.

Snow666

Google lança navegador web aberto

Após uma espera curta, mas tensa, no melhor estilo dos lançamentos da Apple, o Google liberou a versão beta pública de seu navegador web Chrome.

Utilizando código da interface do *Firefox* e o mecanismo de renderização *Webkit*, criado pela equipe do KDE e desenvolvido pela Apple, o Google Chrome contém diversas inovações tecnológicas.

Uma das novidades mais perceptíveis é a velocidade de renderização de páginas — especialmente aquelas tradicionalmente pesadas, cheias de código *Javascript* — um resultado do uso do veloz Webkit e do compilador Javascript JIT V8. Além disso, a arquitetura interna do navegador é uma novidade na área: cada aba utiliza não uma *thread*, mas um processo independente, que fica isolado de todos os outros do navegador e do sistema operacional. Com isso, caso algum aplicativo web rodando na aba saia do controle, é possível matar somente o processo da aba, minimizando as conseqüências do código defeituoso da página.

Como sempre podemos esperar do Google, o navegador traz mais do que simples mudanças tecnológicas e tem a pretensão de mudar a forma como navegamos — inclusive incentivando a separação das abas em janelas, para uso de aplicativos web em janelas independentes.

Infelizmente para usuários Linux e Mac OS, o Chrome só possui uma versão para sistemas Windows, embora a equipe de desenvolvimento planeje também versões para Linux e Mac OS. ■

Apache continua crescendo

As estatísticas de servidores web liberadas mensalmente pela Netcraft mostram que o servidor web livre *Apache* opera, agora, 1,2 milhões de sites a mais que há um mês. Isso significa que o servidor web de código aberto está por trás de 176.748.506 dos websites pesquisados, quase metade de toda a amostra. Em segundo lugar na lista está o *Microsoft IIS*, com uma fatia de 35%, seguido do *Google Web Server* (GWS) com 6% e do *Lighttpd*, também de código aberto, com 3% (1,7 milhões de sites).

Segundo a Netcraft, uma nova promessa é o servidor Nginx, de autoria do programador russo Igor Sysoev. Em sua primeira aparição na lista, ele já alcançou o quinto lugar. As estatísticas estão disponíveis no site da Netcraft. ■

The Apache Software Foundation

Biblioteca Java da Sun sob GPLv2

Três meses após sua promessa na conferência Java One, a Sun Microsystems honrou o compromisso e liberou sob a GPLv2 o *Lightweight UI Toolkit* (LWUIT), projetado para aplicativos em Java ME (*Micro Edition*).

Desde o anúncio na conferência da Sun, a biblioteca recebeu grande atenção, segundo fontes da Sun. Shai Almog e Chen Fishbein, criadores do projeto LWUIT, publicaram duas atualizações do código. Diversos desenvolvedores de aplicativos para dispositivos móveis, além de empresas, já haviam se decidido pelo uso do toolkit gráfico.

A plataforma de desenvolvimento, segundo seus promotores na Sun, é bem mais que uma coleção de ferramentas para criação de interfaces gráficas. Em especial, os desenvolvedores devem garantir que seus aplicativos em Java ME sejam capazes de rodar em diferentes dispositivos. ■

© Linux New Media do Brasil Editora Ltda.

Xen 3.3.0 lançado

Foi lançada a versão 3.3.0 do *hypervisor* de código aberto *Xen*. A nova versão melhora vários recursos e pode ser baixada no site do projeto. O Xen 3.3.0 também está disponível como parte de um pacote com um kernel 2.6.18 para atuar como *Domo* – mas os dois anos de idade dessa versão do kernel não o tornam muito atraente.

O número de sistemas operacionais hóspedes cresceu, e as melhorias gerais englobam o desempenho, os modos de gerenciamento de energia por ACPI e funções de segurança.

A migração de hóspedes entre diferentes hospedeiros também é mais fácil agora, pois não existe mais a necessidade de o hospedeiro de destino usar as mesmas tecnologias de virtualização que o de origem.

Embora a documentação da nova versão ainda não esteja pronta (a página de download do Xen 3.3.0 aponta para um README da versão 3.2.0), há um rascunho das especificações que afirma que o DomU pode ser acessado por drivers nativos (semelhante ao *PCI pass-through*), mas isso depende do suporte à tecnologia VT-d da Intel.

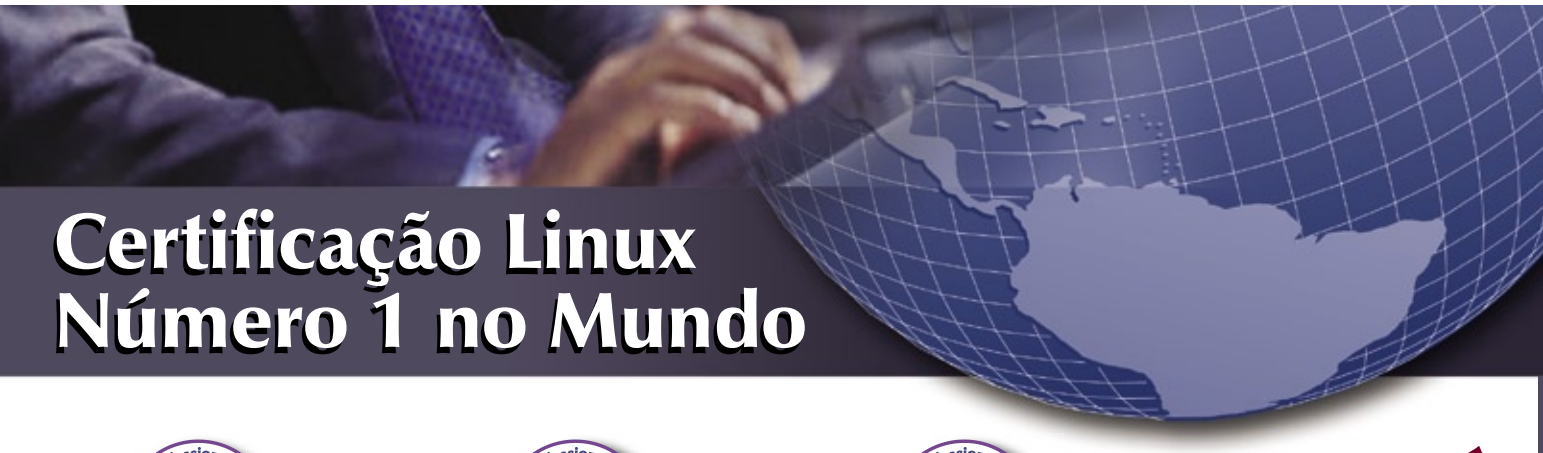
Novamente, a idade do kernel 2.6.18 pode representar um problema, pois a tecnologia VT-d só está presente em hardwares mais novos, e o kernel 2.6.18 não inclui o suporte às tecnologias surgidas nos últimos dois anos. ■

Debian Squeeze vem aí

Pouco antes da data planejada para o lançamento da próxima versão estável do Debian, codinome *Lenny*, o projeto escolheu o nome da próxima versão. Como de costume, o nome *Squeeze* remete a um personagem do longa-metragem animado *Toy Story*.

O Debian Lenny, de número 5.0, é esperado para este mês, como afirma com otimismo o gerente de versões do projeto, Luk Claes. Ele pede que a comunidade faça testes e sugestões, em particular para a atualização do atual Etch para o Lenny.

Segundo a equipe do instalador, haverá apenas um *release candidate* antes do lançamento da versão final do Lenny. Os erros críticos serão eliminados em *squashing bug parties*, de acordo com Claes. O local de encontro para os participantes será o canal `#Debian-bugs` no servidor IRC do projeto Debian (irc.debian.org). ■



LPIC-1: reconhecida no mundo todo como A certificação inicial para profissionais de Linux

LPIC-2: uma certificação avançada em Linux, largamente reconhecida como uma "HOT CERT" do mercado, que proporciona os mais altos salários entre os profissionais de Linux

LPIC-3: a primeira certificação professional enterprise-level em Linux, disponível a partir de janeiro de 2007

OSPREY: um programa único de progresso na carreira para TODOS os profissionais de Open Source

Linux Professional Institute

Saiba mais,
faça-nos uma visita
www.lpi.org/américalatina

© Linux New Media do Brasil Editora Ltda.

Snow666

Sun tenta criar comunidade

A Sun Microsystems lançou um novo site, chamado Kenai, para abrigar seus próprios projetos de código aberto. No anúncio, a empresa – que se descreve como "a maior participante do Código Aberto", participando de mais de 750 projetos de código aberto – alega que o Projeto Kenai vai servir o mundo do Código Aberto da melhor forma possível.

Atualmente em estágio beta, o projeto é baseado no framework web *Ruby on Rails* e integra repositórios *Mercurial* e *Subversion*, além do *Bugzilla* para acompanhamento de tíquetes e do *Sympa* para gerenciamento de listas de email.

Entre os projetos da empresa associados ao Kenai estão o software de virtualização *xVM Server* e o *JRuby*, implementação da linguagem Ruby em Java. As páginas de FAQ do Kenai contêm mais informações sobre a inclusão de outros projetos no futuro.

O *Chief Open Source Officer* da Sun, Simon Phipps, afirmou em seu blog: "Tenho interesse em resolver o problema da proliferação de licenças de código aberto". A empresa trabalhará junto com a Open Source Initiative (OSI) nessa seção. A OSI liberou recentemente um relatório a respeito do problema de proliferação das licenças, no qual recomenda métodos para que os autores tenham escolhas mais fáceis, para evitar licenças que "não funcionam bem em conjunto" e facilitem a compreensão da compatibilidade multi-licenças. O Kenai oferece aos desenvolvedores de novos projetos somente as licenças "recomendadas" pela OSI ("Licenças populares e amplamente usadas ou com fortes comunidades"). Phipps sugere que outras licenças não sejam necessariamente excluídas, mas seriam consideradas como parte de uma segunda fase. Até a Sun, ao liberar seu Java de código aberto, opta pela GPLv2 em vez de usar suas próprias licenças.

O conceito e o layout do site do Kenai contêm elementos de outros projetos de hospedagem, como o Google Code e o Sourceforge, mas afirma ser "mais do que apenas um *Forge*". ■

Cisco adquire Jabber

A gigante dos equipamentos de rede Cisco anunciou no final de setembro a aquisição da Jabber Inc., empresa especializada em mensagens instantâneas de padrão aberto. Analistas apontaram a ação como uma indicação de que a empresa de São Francisco, EUA, está mesmo disposta a enfrentar a Microsoft no campo da colaboração.

O valor da operação não foi revelado, e Doug Dennerline, vice-presidente sênior do grupo de softwares de colaboração da Cisco, afirmou que "com a aquisição da Jabber, vamos conseguir estender o alcance de nosso serviço atual de mensagens instantâneas e expandir as capacidades de nossa plataforma de colaboração".

A Jabber Inc. produz softwares que utilizam o protocolo Jabber (agora chamado XMPP), mas que não são, necessariamente, de código aberto. Entre seus maiores clientes de 2008 está a Marinha dos EUA.

Segundo a Cisco, a operação deve ser finalizada até janeiro de 2009.

Bolsa parada por causa da Microsoft?

Segundo uma notícia da Reuters, as negociações na bolsa de valores de Londres tiveram de ser paralizadas durante sete horas no último dia oito de setembro por problemas computacionais. A causa exata do problema ainda é desconhecida e provavelmente jamais será esclarecida. O incidente poderia ser particularmente embaraçoso para a Microsoft, que, no final de 2006, lançou uma enorme campanha afirmando que a *London Stock Exchange* havia optado pelo Windows em vez do Linux por causa de questões de confiabilidade. Uma opinião obviamente não compartilhada pela NYSE, a bolsa de valores de Nova York, que usa Linux e AIX há mais de um ano sem qualquer parada.

Essa não é a primeira vez que o sistema Infolect – baseado em .NET, MS-SQL e Windows Server – apresenta falhas. Em setembro de 2007, a bolsa de Londres foi atingida por problemas de conectividade quando três gateways Infolect não suportaram a demanda.

© Linux New Media do Brasil Editora Ltda.

WebIntegrator lançado no Linux Park

Durante a terceira edição dos seminários Linux Park, no dia 23 de setembro em Brasília, DF, a empresa ITX Tecnologia da Informação Ltda. assinou com a SLTI um acordo para disponibilização da solução WebIntegrator no Portal do Software Público, administrado pelo órgão federal. A ITX segue o caminho de outra empresa privada, a Light Infocon Tecnologia S/A, que liberou em abril deste ano as soluções *LightBase* e *GoldenDoc* no Portal.

A solução já está disponível no Portal e pode ser acessada diretamente pelos interessados. O desejo da ITX, desde a criação da empresa, era de abrir o código para a comunidade. No início, a preocupação era atender demandas por suporte técnico e ter o desafio de criar uma comunidade de usuários e desenvolvedores que pudessem apoiar a iniciativa.

O WebIntegrator é um ambiente de alta produtividade para o desenvolvimento de aplicações Web em plataforma Java, que cria facilidades de uso e acelera o aprendizado técnico dos desenvolvedores.

Algumas de suas características da solução são um ambiente 100% web, assistentes para construção de páginas e código SQL com interfaces simples, componentes pré-programados, definições em arquivos XML, geração automática de documentação e suporte a *Web Services*, entre outros.

A assinatura da carta de lançamento contou com a presença de Rodrigo Assumpção, Secretário-Adjunto de Logística e Tecnologia da Informação e Ricardo Masstalerz, Sócio Diretor da ITX.

Red Hat adquire Qumranet

A Red Hat divulgou, em um anúncio para a imprensa no dia 5 de setembro, que adquiriu os aplicativos de virtualização de código aberto *KVM* e *SolidICE* da Qumranet, Inc. O KVM (*Kernel Virtual Machine*) já faz parte do kernel Linux há dois anos. Com essa aquisição, a Red Hat prevê tempos de desenvolvimento mais curtos e garante um amplo suporte a hardware, oferecendo assim uma vantagem estratégica em relação a outras soluções.

O SolidICE é um sistema de virtualização de desktops que pode coexistir com o Linux e o Windows. De acordo com o CEO e presidente da Red Hat, Jim Whitehurst, a Red Hat se torna, assim, uma das duas empresas no planeta a oferecerem soluções completas de virtualização, enfatizando sua posição equivalente à da Microsoft como líder em tecnologia. A Red Hat esclarece qual o seu público-alvo em seu website, mostrando que vai oferecer aos usuários de desktop Windows uma plataforma escalável de virtualização. De acordo com o conceito da Red Hat, o produto gerenciará a virtualização do Windows e do Linux tanto em servidores quanto em desktops.

Como confirmou o CEO da Qumranet, Benny Schnaider, todos os funcionários da Qumranet serão transferidos para a Red Hat. "Não poderíamos ter encontrado um parceiro melhor", colocou. O valor da operação foi de US$ 107 milhões, pagos à vista pela companhia de Raleigh, EUA. A aquisição deve ser efetivada ao final do ano fiscal de 2009 da empresa, que termina no próximo mês de fevereiro, com uma expectativa de receita para o ano seguinte estimada em US$ 20 milhões.

Até o momento, a distribuição Linux havia optado pelo Xen como solução de virtualização para seu sistema corporativo RHEL. Numa conversa com o gerente da Red Hat Paul Cormier, ele confirmou que o Xen como componente do RHEL 5 será suportado pelo menos até 2014. Entretanto, o novo hypervisor, apresentado no Red Hat Summit deste ano, deverá estar disponível no final de 2008 e será baseado no KVM. A Red Hat está desenvolvendo uma infra-estrutura de gerenciamento para suportar a tecnologia KVM. ■

© Linux New Media do Brasil Editora Ltda.

Snow666

- **Multiempresa**
- **Multiplataforma**
- **Interface amigável**
- **Compatível com a legislação fiscal e tributária brasileira**
- **Independência do desenvolvedor do software**

- Gerenciamento de cadeia e fornecedores
- Análise de performance
- Contabilidade
- Financeiro
- Produção
- Logística
- Vendas
- MRP
- CRM

© Linux New Media do Brasil Editora Ltda.

Snow666

Flexibilidade e Confiabilidade

Solução de gestão integrada ADempiere:

a tecnologia utilizada por grandes empresas, agora acessível ao seu negócio, pelo melhor custo.

www.kenos.com.br • contato@kenos.com.br • (11) 4082-1305

Kenos
Sistemas de Gestão Integrada

© Linux New Media do Brasil Editora Ltda.

Entrevista com Andy Oram, editor-chefe da O'Reilly Media

Web e comportamento

Andy Oram não é um futurólogo, mas se baseia em observações do passado da Web e do comportamento das pessoas para prever o futuro da civilização online.

por Pablo Hess

A americana O'Reilly Media é uma das mais destacadas editoras de livros técnicos na área de TI na atualidade. Seus livros, cuja característica é uma xilogravura – em geral, de um animal – costumam figurar entre as obras de referência para linguagens de programação, softwares e sistemas operacionais, entre vários outros temas tecnológicos.

Em visita ao Brasil, o editor-chefe da O'Reilly, Andy Oram, conversou com a **Linux Magazine** a respeito de suas visões freqüentemente polêmicas sobre o futuro das publicações e da própria Web.

Algumas de suas colocações:

"Acho que, no futuro, a tecnologia da informação deixará de ser uma fonte de grandes lucros – e acho que essa mudança vai promover o valor da cultura e diferentes populações ao redor do mundo.

Após o hardware se tornar commodity, o valor da TI foi para o software. Com o crescimento do Código Aberto, o dinheiro agora vem da criação de serviços online, desde o SalesForce até o Google ou o Facebook.

Se a TI for comoditizada, acredito que a cultura trocada pelas pessoas passará a ser o item realmente valioso."

"Muitas pessoas ficam irritadas porque os mais jovens preferem aprender na prática e com interação do que com livros. Porém, durante séculos, todos aprenderam na prática e pela interação com outros. Os livros só vieram há poucos séculos, sempre excluíram alguns tipos de pessoas e são uma forma muito ineficiente de aprender."

Confira a entrevista:

Andy Oram, editor-chefe da O'Reilly Media.

Linux Magazine» ***Você diz que, após o hardware se tornar uma commodity, o mesmo vai ocorrer com os softwares e serviços. Você já vê algum exemplo dessa tendência?***

Andy Oram» É uma pergunta justa, e tenho que admitir que não conheço nenhum exemplo disso ainda. Os capitalistas de risco dos EUA parecem adorar as redes sociais e outros serviços. Então, este ano, o esse mercado ainda será grande.

A base da minha afirmação são duas considerações analíticas, uma de raiz lógica e a outra baseada no passado da Internet.

A história da Internet é: espaços privados, conhecidos como jardins murados, perdem para os espaços abertos. Vários observadores já mencionaram isso desde o fim do CompuServe e do Prodigy.

Pessoalmente, me incomoda ter que digitar as mesmas informações em cada rede social à qual me junto (embora o *OpenSocial* do Google possa vir a mudar isso) e também ter que visitar redes diferentes para me conectar a amigos em cada uma delas. Os efeitos de rede garantem que as pessoas vão se beneficiar mais de sistemas interconectados e interoperáveis do que de sistemas que tentem engaiolá-las.

Minha objeção lógica à tendência atual é que acho que os anúncios e outras formas de sustento para o conteúdo gratuito diminuirão em decorrência da migração das empresas para a publicidade online (mais eficaz, portanto mais barata) e também aos problemas econômicos e ambientais inerentes ao aumento do consumo.

LM» ***Você diz também que a cultura vai representar o valor no mercado de TI. O que exatamente você quer dizer com "cultura", nesse caso?***

© Linux New Media do Brasil Editora Ltda.

AO» Eu percebo que a produção e as trocas culturais hoje em dia são feitas sobre sistemas de informação, e suponho que esses sistemas serão comoditizados no futuro da mesma forma como as camadas sob eles já sofreram esse processo. Portanto, se essa camada for comoditizada, é o que se faz sobre ela que passa a deter o real valor.

Então, a próxima era da produção pertente à cultura. É uma suposição.

LM» ***Com relação ao aprendizado na prática, você critica os livros como meio de aprendizado, mas trabalha com eles. Como você pretende agir em relação a isso a partir de dentro da indústria editorial?***

AO» Passei quatro anos pesquisando como as pessoas aprendem online e tenho um fascínio pela produção colaborativa (*peer production*), com pessoas ajudando umas às outras. Tenho também uma apresentação em http://www.praxagora.com/project_education/presentation/growing_participation.pdf que ilustra como eu gostaria que as trocas de informação evoluíssem.

Basicamente, eu proponho nela que wikis são muito mais eficazes do que listas de emails ou fóruns, justamente por permitirem a interação entre a pessoa que pergunta e a que responde, assim como a participação de várias outras pessoas numa mesma resposta. O resultado tende a chegar mais rápido e a solução tem uma qualidade melhor dessa forma do que em listas de email, por exemplo.

Além disso, quando leio um livro, costumo ler entre dez e 50 páginas de cada vez. Acho arbitrário agrupar 300 páginas numa obra. Acho que materiais online refletem melhor a forma como trocamos informações com outras pessoas. As trocas são pequenas quantidades de dados em uma conversa que cresce.

Quanto ao futuro, vou continuar publicando livros (e deixá-los bons) enquanto eu puder, mas existem várias vantagens de entrar no mundo online.

A empresa onde trabalho reconhece a importância da mídia online, e exploramos essa fronteira com um serviço de assinatura online (Safari Books Online), artigos e blogs, vídeos e sites de desenvolvimento online para livros.

LM» ***Ao contrastar livros e a Web como vetores de informação, você não acha que a falta de portabilidade ainda é um dos maiores problemas das mídias eletrônicas? Ainda não há formas tão confortáveis para ler textos no ônibus, na praia ou no parque quanto um material impresso, pois smartphones têm telas pequenas demais e laptops são grandes demais.***

> *Acho que, no futuro, a tecnologia da informação deixará de ser uma fonte de grandes lucros.*

AO» Eu não uso leitores de e-books, mas acho que eles vão melhorar e ainda vou usá-los algum dia. O Kindle, da Amazon, é um grande avanço. O papel eletrônico seria ainda melhor. Porém, o mais importante, para mim, é adotar formatos abertos para termos certeza de que vamos conseguir colocar em qualquer dispositivo todos os conteúdos eletrônicos que adquirirmos.

LM» ***Certas pessoas argumentam que o próprio ato de ler está mudando profundamente, especificamente diminuindo a duração da nossa atenção em resposta a conteúdos online muito dinâmicos, enquanto outras pessoas afirmam que jamais se leu tanto quanto hoje. Você concorda com algum dos dois pontos? Quais são, para você, as mudanças mais interessantes trazidas pela Web para nosso comportamento?***

AO» Eu concordo com os dois pontos. Quando afirmam que a disponibilidade imediata de qualquer coisa após uma busca no Google pode estar nos tornando burros, eu sugiro duas perspectivas para a questão.

Primeiro, os algoritmos de busca do Google têm uma flexibilidade evolutiva contínua e, portanto, é diferente do Taylorismo. Taylor encontrou um processo único e intelectualmente nulo para cada objeto fabricado. Mas os usuários da Internet mudam seus hábitos de navegação ao longo do tempo e os sites descobrem novos truques para aumentar suas colocações nas buscas. Apesar de os algoritmos do Google serem mecânicos e discretos como os estudados por Taylor, eles se adaptam organicamente a seu ambiente – diferentemente do proposto pelo Taylorismo.

Em segundo lugar, apesar de a leitura de livros e revistas estar diminuindo, vários leitores expandem suas mentes por meio do intercâmbio online em blogs e fóruns de discussão. A contemplação solitária está cedendo lugar para uma interação comunitária de certa forma semelhante aos antigos simpósios gregos.

É verdade que perderíamos muito se a contemplação solitária fosse abandonada, mas a troca de informações com a comunidade pode ser um complemento ideal. ■

© Linux New Media do Brasil Editora Ltda.

Pirataria de software

Cezar Taurion

Pirataria ou cópia não autorizada? Independente do nome dado ao fenômeno, há diversas formas de analisá-lo e combatê-lo.

por Cezar Taurion

Um assunto que volta e meia aparece na mídia é o da pirataria de software. O próprio termo gera controvérsias. Alguns afirmam que ele é impróprio, pois uma cópia não autorizada não é um ato violento como os comumente associados aos piratas. A terminologia mais precisa seria, segundo os críticos, "cópia não autorizada".

De qualquer maneira, campanhas anti-pirataria e organizações como a BSA afirmam que a cópia não autorizada seria um dos maiores problemas da indústria de software, que causa um prejuízo de dezenas de bilhões de dólares por ano. Por outro lado, alguns analistas sugerem que esses valores estariam sendo superestimados, pois muitos usuários das cópias piratas contabilizadas no cálculo do prejuízo não usariam o software caso tivessem que pagar.

Outro argumento comumente usado nas campanhas anti-pirataria também é questionado: para muitos, o uso de cópias não autorizadas não reduziria empregos e inovações, mas geraria o efeito contrário, uma vez que usuários e pequenas empresas que usam esses softwares (que não teriam acesso caso fossem pagos) aumentam a eficiência de suas operações, melhorando o desempenho da economia, gerando mais renda e empregos.

Alguns chegam a argumentar que a pirataria seria até benéfica para alguns fornecedores de software de massa, pois cria uma base de usuários muito grande, gerando hábitos de uso e impedindo a adoção de produtos substitutos, como softwares Open Source.

Um dos motivos é a facilidade de se copiar software. Um único conjunto de CDs pode ser usado para instalar o software em dezenas de máquinas.

Mas o que impulsiona as cópias não autorizadas é o baixo valor percebido do software por parte dos seus usuários. Se olharmos um pacote de escritórios, veremos que uma grande parcela de suas funcionalidades não é usada ou não gera real valor para a maioria dos seus usuários, caracterizando o *excesso de capacidade*, pois as funcionalidades embutidas excedem a capacidade de uso por seus usuários típicos.

Outro aspecto que devemos analisar é quem são os tais "piratas". No caso dos usuários finais, provavelmente eles não causam prejuizos às empresas de software, pois não comprariam mesmo o produto. Usam porque seu custo de aquisição é zero. Por outro lado, os comerciantes ilegais usufruem financeiramente da cópia não autorizada e burlam os mecanismos fiscais. Estes devem ser combatidos.

É claro que copiar de forma não autorizada é uma violação das leis. Mas, antes de buscar punições de usuários pegos com cópias piratas, seria melhor encontrar soluções que minimizem o problema.

Uma alternativa para reduzir as cópias não autorizadas seria ajustar o preço dos softwares às necessidades dos usuários. Porém, a solução mais adequada para o mercado de softwares de massa, como suítes de escritório, seria a adoção intensiva de alternativas Open Source, como o *OpenOffice.org* ou o *Symphony* da IBM, gratuito e baseado nessa solução.

O movimento Open Source é um risco muito maior para as empresas de software de massa que adotam modelos de negócio baseados em vendas de licença do que a própria pirataria. A cópia não autorizada aumenta a externalidade de rede, enquanto que um substituto Open Source quebra esse modelo. Por isso, durante muito tempo os esforços de pressão e propaganda (FUD) de determinadas empresas de software foram muito mais direcionados a combater o crescimento do Open Source do que a combater a pirataria. ■

Sobre o autor

Cezar Taurion (ctaurion@br.ibm.com) é gerente de novas tecnologias aplicadas da IBM Brasil e editor do primeiro blog da América Latina do Portal de Tecnologia da IBM developerWorks. Seu blog está disponível em http://www-03.ibm.com/developerworks/blogs/page/ctaurion.

© Linux New Media do Brasil Editora Ltda.

Snow666

Rolos de papel para pianos automáticos

Jon 'maddog' Hall

Maddog descobre que o copyright impede a preservação de rolos de papel de pianos automáticos.

por Jon "maddog" Hall

Algumas pessoas sabem que eu coleciono instrumentos musicais automáticos: pianos, órgãos, *nickelodeons* (um tipo de *jukebox*) e outros mecanismos que usam um rolo de papel para controlar a operação do instrumento. Isso foi o resultado natural do meu fascínio por controlar hardwares com "lógica" e "software" e meu amor por música. Até criei uma palestra sobre como o Software Livre é como um piano automático, que já ministrei várias vezes, completa com ilustrações e músicas tocadas pela minha coleção para piano.

Há muitos anos eu entrei na Associação de Colecionadores de Instrumentos Musicais Automáticos (AMICA, na sigla em inglês) e recebo suas publicações. A edição deste mês falou sobre como a Yamaha está usando Linux em seus pianos automáticos para controlar a automação. Esses pianos Yamaha Disklavier Mark IV oferecem um conjunto impressionante de recursos e a capacidade de baixar atualizações de softwares musicais da Internet.

O artigo me lembrou do órgão Marshall & Olegtree Opus 1 instalado na Trinity Church em Nova York. Projetado para substituir o órgão real de tubo da Trinity, destruído durante o ataque ao World Trade Center em 2001, o Opus 1 utiliza dez PCs Linux para controlar 74 canais de som com potências entre 150 e 500 watts cada. Ele tem até um PC sobressalente para ser usado em caso de falha de hardware em algum dos dez PCs. Em seu site, os projetores do Opus 1 dizem que um certo sistema operacional popular em desktops ficou instável demais para qualquer uso com a importância de um recital de órgão e que por isso escolheram o Linux. Evidentemente, esse instrumento também está conectado à Internet e pode baixar novas vozes ou ser monitorado durante um concerto.

Mês passado foi a primeira vez, no entanto, que eu consegui participar de um capítulo local da AMICA. Em vários casos, uma das desvantagens de colecionar os velhos instrumentos operados por papel é que o papel está se desfazendo, e alguns dos membros estão tentando desesperadamente preservar suas antigas músicas copiando os rolos ou capturando a informação em arquivos MIDI ou MPEG-3. Essas pessoas depois colocam esses arquivos – canções gravadas em rolo por empresas que já fecharam as portas há tempos – à disposição na Internet para compartilhar com outros. Decidi que eu tinha várias canções na minha coleção que podiam ser compartilhadas na Internet para que outros escutassem meus instrumentos tocando. Como alguém do Software Livre e seguidor das questões do copyright, eu usaria uma melodia cujos direitos autorais já venceram há muito, como "Greensleeves". Porém, como algo no fundo da minha mente começou a me incomodar, contactei uma empresa que ainda fabrica esses rolos para pianos e pedi para falar com seu departamento legal.

Expliquei ao meu interlocutor o que eu queria fazer e perguntei se haveria problemas em usar uma música antiga que já estivesse liberada dos direitos autorais. Ele me disse que Greensleeves realmente já estava liberada dos direitos autorais, mas que a simples transferência da música para o rolo de papel também envolve copyright.

Acabamos por concordar que se eu fizesse uma gravação de menos de 30 segundos do rolo antigo de uma empresa fora de ação, usasse-a somente para fins pessoais e trancada para ninguém mais conseguir usá-la, eu provavelmente não seria processado. E eu agradeci por sua atenção. ■

Sobre o autor

Jon 'maddog' Hall é presidente da Linux International, instituição internacional dedicada a promover o Linux e o Software Livre e de Código Aberto. Maddog viaja o mundo ministrando palestras e debatendo com decisores sobre o uso do Software Livre em âmbito tanto corporativo quanto comunitário.

© Linux New Media do Brasil Editora Ltda.

Fique um passo a frente dos invasores

Prevenção

Invasores de redes têm várias formas engenhosas de escalar privilégios e ocultar sua presença no sistema. A melhor proteção é mantê-los fora.

por Tim Schürmann e Joe Casad

Logo quando você pensava que havia dominado a arte de proteção contra invasões, os cibercriminosos descobrem novas técnicas para atravessarem sua segurança. Os agressores usam qualquer vantagem possível para ficarem escondidos e ganharem controle. Então, você não deve usar tudo que estiver disponível para mantê-los do lado de fora?

As matérias de capa deste mês são dedicadas a manter os invasores fora do seu sistema. Em nosso primeiro artigo, estudamos uma técnica poderosa para manter suas portas do firewall fechadas para todos os usuários – mas ainda aberta ao tráfego de máquinas amigáveis.

E se alguém invadisse seu sistema Linux e substituísse o programa *login* por uma variante maliciosa? O novo *login* descobre o seu nome de usuário e sua senha e envia os dados coletados por meio de um buraco em seu firewall para algum servidor em outro ponto da Internet. Ninguém suspeita do novo *login*, embora o invasor saiba que um administrador de sistema atencioso pode se perguntar sobre a mudança no tamanho do arquivo.

Mas criminosos na Internet têm uma forma de cobrir seus rastros. Junto com a ferramenta *login* alterada, o meliante inclui uma variante do programa *ls*. Essa nova versão do *ls* mascara mudanças de tamanho e data do programa *login*.

O agressor também decide substituir vários outros programas do sistema que funcionam juntos para coletar informações e ocultar qualquer vestígio da intrusão. Até programas antivírus são inúteis, pois também são enganados pelas ferramentas manipuladas.

Essa situação não é, de forma alguma, ficção. Os invasores freqüentemente trazem uma coleção de ferramentas para capturar informações, abrir *backdoors* e esconder suas atividades. Esse conjunto de armas é conhecido como *rootkit*.

Um rootkit geralmente contém vários componentes que realizam várias tarefas:

- um cavalo de tróia deposita o rootkit no sistema;
- um *sniffer* analisa o tráfego de rede e obtém as credenciais de acesso;
- em alguns casos, *keyloggers* registram as teclas pressionadas para capturar senhas ou PINs antes de o sistema criptografá-las;
- uma *backdoor* fornece ao invasor acesso ao sistema.

Todas essas atividades são camufladas pela substituição de arquivos de sistema e, para rootkits atuais, redirecionamento de chamadas de API. Outros componentes depois põem o computador em uso – possivelmente para distribuir spam ou desferir ataques de negação de serviço.

Inovação

No início, os rootkits simplesmente substituíam ferramentas de sistema populares como *ls*, *passwd* e *ps*. Os especialistas em segurança rapidamente aprenderam a detectar esses rootkits básicos e os programadores de malwares aprenderam, em seguida, como atuar no kernel em si.

Se um agressor conseguir injetar código malicioso no kernel, o código ofensivo do kernel pode capturar e redirecionar qualquer requisição.

Rootkits rodando no espaço do kernel são particularmente difíceis de descobrir. No Linux, rootkits de kernel costumam ser injetados por meio de um módulo do kernel, o que explica por que são conhecidos como rootkits LKM (*loadable kernel module*).

© Linux New Media do Brasil Editora Ltda.

Os desenvolvedores de rootkits usam várias técnicas para infestar o kernel. Uma opção é manipular o diretório de memória via `/dev/kmem`.

Firmware

Rootkits de firmware oferecem um vetor alternativo de ataque. Eles infectam o firmware do PC e sobrevivem a uma reinicialização. Alguns rootkits ficam bem à vontade nas rotinas de firmware do ACPI. Um disco de recuperação limpo não é muito útil contra esse tipo de ameaça.

Pílula azul

A última tendência são os rootkits virtualizados. Um rootkit virtualizado funciona como uma máquina virtual: a primeira etapa é o rootkit modificar o processo de inicialização para que seja carregada uma máquina virtual antes do sistema operacional. O sistema operacional que você acha que está rodando no hardware na verdade roda numa máquina virtual. Isso dá ao rootkit controle total sobre o computador, sem que o sistema operacional ou o usuário perceba qualquer coisa. O nome dessa técnica – *Blue Pill*[1] – é uma alusão ao filme *Matrix*, em que a pílula azul mantém uma pessoa no mundo virtual. A técnica para detectá-la se chama *Blue Pill Detection*.

Quadro 1: Compilar ou não compilar

As instruções para os administradores sugerem a compilação dos detectores de rootkits antes do uso. Porém, como se pode imaginar, isso é um problema caso o rootkit já tenha controle do sistema. Nesse caso, o compilador talvez já esteja comprometido e crie uma versão manipulada do detector. Por esse motivo, deve-se tentar compilar o detector imediatamente após a instalação do sistema, ou usar um sistema comprovadamente limpo.

Prevenção

Rootkits são muito difíceis de detectar após sua instalação no sistema. Antes de serem instalados, no entanto, é possível identificá-los com programas antivírus e buscadores de rootkits (ou detectores de rootkit), que usam assinaturas ou heurística para identificar os culpados (veja também o **quadro 1**).

A melhor forma de parar um rootkit é não deixá-lo entrar. Os artigos desta edição discutem algumas técnicas para manter invasores do lado de fora, antes que fiquem confortavelmente instalados.

Contudo, apesar de todos os nossos esforços, você jamais ficará seguro o suficiente para ignorar a possibilidade de um rootkit se esgueirar através das suas defesas. As seções a seguir discutem algumas estratégias para remoção de rootkits.

Verificação visual

Como os resultados em sistemas em execução não são inteiramente conclusivos, deve-se desligar o suspeito e reiniciá-lo a partir de uma fonte sabidamente limpa. Ela pode ser um disco de recuperação, por exemplo. O verdadeiro teste é verificar todos os arquivos suspeitos, conferindo-os contra um sistema limpo. Isso geralmente é feito comparando-se o sistema atual com um *snapshot* do sistema obtido diretamente após a instalação. Para impedir que o rootkit afete os resultados da comparação, armazene o snapshot original numa mídia somente leitura.

Obviamente, um becape completo do sistema precisa de espaço significativo em disco. Como alternativa, pode-se apenas criar *checksums* para verificar a integridade dos arquivos. Para fazer isso, o buscador de rootkit calcula uma assinatura para cada arquivo; se o rootkit tentar alterar o arquivo, o checksum será automaticamente alterado e não coincidirá com o calculado na última atualização.

Chkrootkit

Um dos buscadores de rootkit mais populares para Linux é o *Chkrootkit*[2]. A ferramenta de Nelson Murilo e Klaus Steding-Jessen engloba uma coleção de pequenos programas em C especialmente escritos para detectar uma anomalia específica. Após descompactar o arquivo, compile os aplicativos com o sugestivo comando:

```
make sense
```

Figura 1 O *Chkrootkit* rodou uma vez e não encontrou qualquer rootkit conhecido.

© Linux New Media do Brasil Editora Ltda.

Snow666

e depois faça uma execução de teste digitando `./chkrootkit` (**figura 1**). Infelizmente, o buscador de rootkit chama vários programas binários do sistema infectado. Por isso, sempre é bom rodar essas ferramentas diretamente a partir de uma mídia limpa como um CD ou DVD:

```
./chkrootkit -p /cdrom/bin
```

para evitar uma infecção.

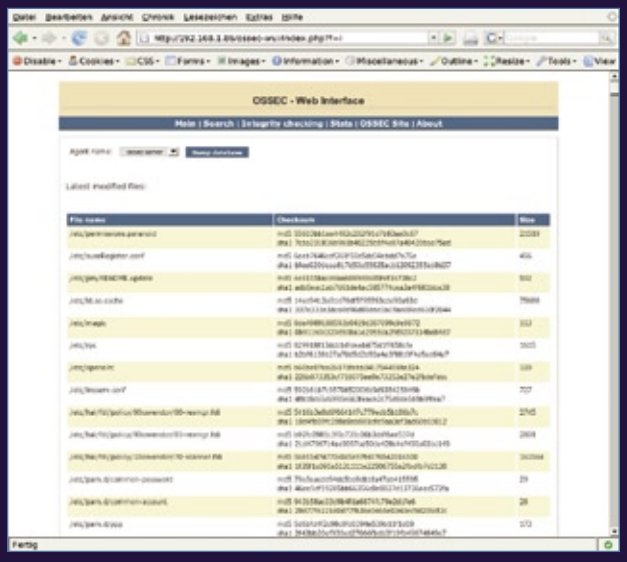


Figura 2 O projeto *OSSEC* oferece uma interface web para o agente de monitoramento.

Rootkit Hunter

Exatamente como o Chkrootkit, o Rootkit Hunter também varre o sistema infectado em busca de características específicas que indiquem a existência de um rootkit. Originalmente escrito por Michael Boelen, a ferramenta foi passada para uma equipe de desenvolvedores em 2006, e os resultados são visíveis no *SourceForge*[3]. Após baixar e descompactar o arquivo, torne-se root e digite o seguinte comando para compilá-lo:

```
./installer.sh --layou custom .
➥-- install
```

Depois, entre no subdiretório `files/` e modifique o arquivo de configuração `rkhunter.conf`. A linha de comando a seguir inicia a verificação:

```
./rkhunter --check
```

OSSEC

Diferentemente das soluções citadas acima, o *OSSEC*[4], tema da matéria "Segurança Profissional", possui uma ampla gama de funções (**figura 2**). Além de verificar automática e periodicamente a presença de rootkits, ele oferece monitoramento e análise permanentes de arquivos de log, verificações de integridade e detecção de intrusão baseada em regras. O projeto Rootcheck tem assinaturas de rootkits gratuitas para download em seu site [5].

O OSSEC permite a configuração de uma estrutura de clientes e servidores: os agentes monitoram o sistema operacional das máquinas clientes. Sempre que ocorre um evento suspeito ou atípico, é enviada uma mensagem para um servidor central de monitoramento, que depois faz a análise, tira conclusões e soa o alarme, caso necessário.

Limpeza

Depois de detectar a presença de um rootkit, a única solução eficaz é uma reinstalação completa do sistema infectado – o software malicioso vai muito fundo no sistema para permitir sua remoção com algum grau de certeza. Com isso, é bom fazer todo o possível para evitar que um rootkit comprometa o sistema. A prevenção é a única resposta real – particularmente, instalar patches de segurança. Para impedir que os malwares os explorem, é preciso certificar-se de que todas as vulnerabilidades estejam remediadas.

Recente e excelente

O princípio de usar o software mais recente vale para detectores de rootkit. O *Zeppoo*[7] tem quase dois anos de idade, por exemplo, e é incapaz de detectar qualquer um dos rootkits da safra atual por causa da total ignorância das técnicas empregadas por eles. Provavelmente é melhor deixar o Zeppoo no *Sourceforge* e usar outra ferramenta.

O próximo obstáculo eficaz é o firewall, de preferência um que controle o tráfego de rede em todas as camadas da pilha TCP/IP. Em servidores em ambientes hostis, o uso de um kernel estático e uma ferramenta de detecção de intrusão como o OSSEC são imprescindíveis.

Tenha em mente, no entanto, que humanos são, provavelmente, o principal vetor de ataque para rootkits. Se um usuário for convencido por emails de *phishing* ou vigor sexual, os melhores mecanismos de segurança vão falhar miserável e fatalmente. ■

Mais informações

[1] Blue Pill (em inglês): http://en.wikipedia.org/wiki/Blue_Pill_(malware)

[2] Chkrootkit: http://www.chkrootkit.org

[3] Rootkit Hunter: http://rkhunter.sourceforge.net

[4] OSSEC: http://www.ossec.net

[5] Assinaturas do Rootcheck: http://www.ossec.net/rootkits

[6] RK Profiler LX: http://www.trapkit.de/research/rkprofiler/index.html

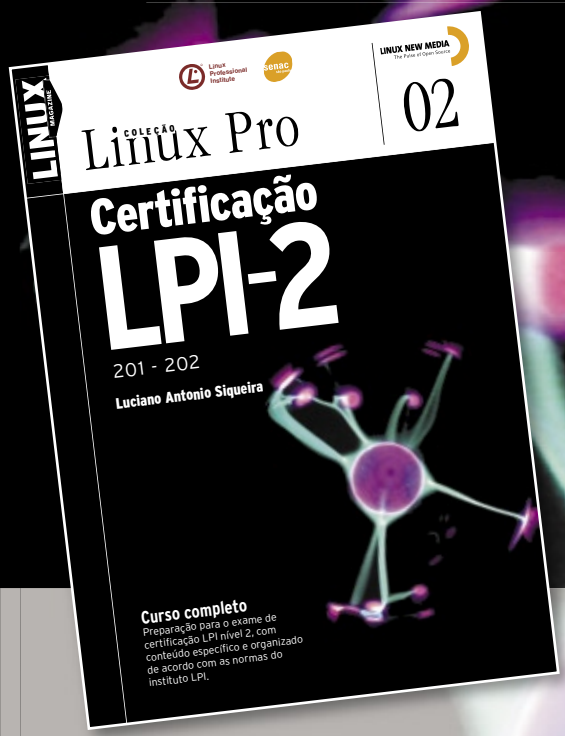
[7] Zeppoo: http://sourceforge.net/projects/zeppoo

© Linux New Media do Brasil Editora Ltda.

Snow666

Coleção Linux Pro

Prepare-se para a principal certificação profissional do mercado Linux



O primeiro volume traz informações referentes à LPI-1 e é o primeiro passo para a certificação. Estude para a prova de acordo com o conteúdo programático estabelecido pelo LPI.

Pautado conforme o roteiro estabelecido pelo próprio Linux Professional Institute e por este recomendado, o segundo volume é voltado à preparação do exame para a LPI-2.

Certifique-se para entrar em um mercado de trabalho em pleno crescimento no Brasil e no mundo.

Só a LPI garante a formação que o mercado espera para lidar com os ambientes mais diversos.
A qualidade destes volumes é atestada pelos selos do LPI e do SENAC, que os utilizam como material didático em seus cursos.

A venda nas melhores livrarias, no site www.linuxmagazine.com.br, ou pelo telefone (11) 4082-1300.

LINUX NEW MEDIA
The Pulse of Open Source

© Linux New Media do Brasil Editora Ltda.

Snow666

Acesso remoto seguro com single-packet port knocking

Toc-toc

Se você procura uma camada extra de segurança para acesso remoto, experimente o single-packet port knocking.

por Juliet Kemp

A criptografia por chave pública significa que o tráfego é seguro e que se as chaves forem corretamente verificadas, ela não fica vulnerável a ataques do tipo *man-in-the-middle*. Ocasionalmente encontram-se exploits, mas eles são rapidamente solucionados. Entretanto, seu sistema pode ainda estar vulnerável a ataques de força bruta. Se você tiver somente algumas poucas contas de usuários, se seus nomes de usuários forem incomuns e suas senhas criadas com cuidado, talvez isso não seja um problema. Mas se houver vários usuários no sistema, já fica difícil garantir a segurança de todas as senhas. Uma solução possível é usar um sistema de adivinhação de senhas como o *John the Ripper*[1], que consegue descobrir senhas fracas antes que alguém com intenções malévolas consiga explorá-las.

Além disso, também se pode criar regras de firewall para estabelecer um número máximo de tentativas de conexão a partir do mesmo IP, seja indefinidamente ou somente nos próximos poucos minutos (a última opção é melhor, por evitar problemas para usuários reais que ocasionalmente erram suas senhas repetidas vezes).

Outra alternativa é a técnica do *port knocking*. Numa configuração tradicional de port knocking, todas as portas do servidor ficam fechadas por padrão. Para alguém de fora, a rede parece inacessível. Um usuário remoto que quiser acesso tenta iniciar várias conexões com uma seqüência específica de portas fechadas. Essas conexões não têm sucesso, mas são registradas pelo servidor. Após a seqüência correta de tentativas de conexão ("batidas", "knocks"), um *daemon* no servidor edita as regras de firewall para permitir uma conexão a partir do IP de onde partiu a seqüência de batidas. Com isso, o usuário consegue acesso remoto normalmente.

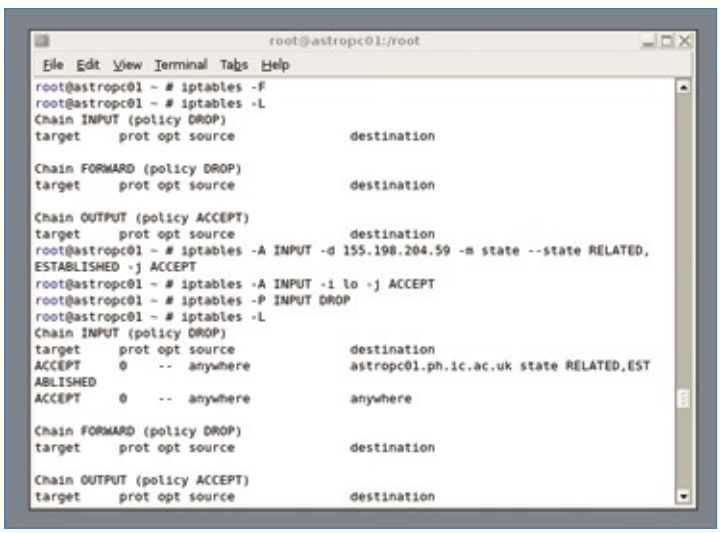


Figura 1 Configuração do *iptables*.

A porta SSH, portanto, é aberta somente em certas circunstâncias (quando o usuário demonstra que é legítimo), em vez de ficar aberta o tempo todo. Normalmente é necessária uma senha para login após a porta ser aberta, oferecendo mais uma camada de segurança.

O Port knoking tem alguns benefícios:

- É oculto – observadores externos não conseguem saber se um firewall está escutando batidas na porta (essa é a vantagem mais importante). Mesmo que um cracker tenha acesso a um exploit para SSH, ele não conseguirá chegar até o servidor SSH para usá-lo.
- É muito flexível – pode-se criar qualquer regra que se deseje.
- É à prova de falhas – se você não pode confiar nos seus usuários para criarem senhas seguras, você fica vulnerável a ataques de força bruta, assim como a qualquer furo de segurança na implementação do SSH.

Contudo, o port knocking tradicional também tem algumas desvantagens. Em primeiro lugar, o programa cliente precisa dar as batidas. Dependendo da tolerância dos seus usuários a esse tipo de exigência – e da sua própria habilidade para fazer a configuração do cliente remoto –, isso pode ser problemático. Além disso, talvez não seja possível rodar esse programa a partir de alguns locais (por exemplo, numa biblioteca ou cibercafé). Se o

© Linux New Media do Brasil Editora Ltda.

processo no lado do servidor der errado, ninguém (nem você) conseguirá se conectar à máquina.

O port knocking também aumenta o tráfego de rede, mas se sua configuração reduzir o tráfego em geral por causa da varredura de portas e tentativas de força bruta, essa redução pode compensar com vantagens o aumento de tráfego no processo de batidas. Outro problema é que o port knocking tradicional não é exatamente imbatível: pode ser possível um agressor monitorar o tráfego e detectar a seqüência de batidas.

Autorização com um pacote

Uma versão mais recente da mesma idéia básica – rodar um servidor que pareça fechado até receber a "batida secreta" – é a *Single-Packet Authorization* (SPA). Diferente do port knocking tradicional, que requer uma seqüência de várias batidas, o SPA exige, como seu nome sugere, somente um único pacote criptografado para comunicar todas as informações necessárias.

Uma sessão SSH só pode ocorrer após um pacote criptografado válido ser detectado e, assim como no port knocking, é impossível descobrir se há um *sshd* rodando do lado de dentro da rede caso você esteja fora dela.

A principal vantagem do SPA é que seus pacotes não podem ser repetidos, enquanto o port knocking tradicional permite – pelo menos em teoria – que um agressor descubra a seqüência e a replique. Além disso, o SPA é mais rápido e difícil de detectar, pois requer um único pacote.

Fwknop

A melhor ferramenta para implementar o SPA é o *fwknop*, disponível no site CipherDyne[2], que é operado pelo pesquisador de segurança Michael Rash. Apesar do fwknop também fazer o port knocking tradicional, seus autores recomendam enfaticamente o uso da solução SPA.

Instalação

Para começar a instalação, será necessário instalar o pacote *libpcap-dev* (no *Debian*) ou o pacote equivalente que forneça as bibliotecas de desenvolvimento do *PCAP*. Depois, baixe o arquivo mais recente do fwknop[3]. Após descompactá-lo, basta executar seu script de instalação:

```
./install.pl
```

Durante esse processo, selecione *server* para usar o modo local de execução e depois *pcap* para o método de aquisição de dados. Escolha sua interface de rede quando pedido e defina o endereço de email para alertas de acesso. Depois, provavelmente é necessário escolher *yes* para iniciar o fwknop a cada inicialização.

Para garantir que o fwknop rode corretamente no sistema, rode os testes após a instalação:

```
test/fwknop_test.pl
```

a partir do diretório de instalação do fwknop.

Agora, é preciso completar a configuração editando o arquivo `/etc/fwknop/access.conf`, mas não deve ser necessário editar as configurações em `/etc/fwknop/fwknop.conf`, pois todas as regras são definidas em `access.conf`. O arquivo `access.conf` de exemplo permite o acesso por SSH por 30 segundos após a chave especificada ter sido enviada com sucesso, que é um padrão razoável. A única linha que realmente precisa ser editada é a que começa com `KEY` – altere a chave nela para uma senha a sua escolha.

Antes de iniciar o fwknop, é preciso configurar o seu firewall *iptables* para que as conexões já existentes e a interface loopback sejam permitidas, enquanto todas as outras conexões são negadas com `DROP`. Tenha cuidado ao configurar regras do iptables. É muito fácil acabar trancado do lado de fora do servidor caso algo dê errado.

Se o iptables já estiver em execução, digite `iptables -F` para limpar todas as regras, defina as regras de permissão das conexões já estabelecidas e da interface loopback e, em seguida, defina a regra padrão de entrada como `DROP` (**figura 1**):

```
# iptables -A INPUT -d 1.2.3.4 \
    -m state --state \
    RELATED,ESTABLISHED -j ACCEPT
# iptables -A INPUT -i lo -j
➥ ACCEPT
# iptables -P INPUT DROP
```

Feito isso, inicie o fwknop com `/etc/init.d/fwknop start`.

Teste

Para testar sua instalação, é preciso um cliente e também um servidor. Novamente, será necessário instalar o pacote *libpcap-dev* na máquina cliente e depois instalar o fwknop como acima; porém, selecione *client* durante o processo de instalação.

Para um teste inicial, confirme a impossibilidade de login via SSH com:

```
$ ssh usuário@exemplo.com.br
```

Isso não deve gerar qualquer resposta. Então, caso haja alguma resposta, verifique a sua configuração do iptables no servidor.

Em seguida, experimente enviar o pacote de "batida":

```
$ fwknop -A tcp/22 -a \
    cliente.exemplo.com.br -D \
    servidor.exemplo.com.br
```

O parâmetro `-A` especifica qual porta do servidor se deseja acessar – e com qual protocolo. O `-a` especifica a máquina (ou IP) que o fwknop deve permitir que se conecte.

Essa opção é útil para evitar um ataque do tipo *man-in-the-middle*, ga-

© Linux New Media do Brasil Editora Ltda.

Snow666

rantindo que o endereço de origem seja duplicado no pacote criptografado, para que, caso o agressor forje os cabeçalhos do pacote, ele não tenha sucesso (porque o cabeçalho forjado não coincidiria com o pacote criptografado). A opção `-D` especifica o servidor a ser contactado.

Depois de pedida a senha definida no arquivo `access.conf` do servidor, um pacote será enviado ao servidor, e você terá 30 segundos para se conectar a ele com uma conexão SSH padrão (**figura 2**). O limite de 30 segundos é o padrão.

Assim como a senha, pode-se trocar esse intervalo no arquivo `access.conf` do servidor. Se o teste não tiver sucesso, talvez o pacote de abertura tenha sido enviado para uma porta UDP alta.

Se o cliente e o servidor se encontrarem em redes diferentes com um firewall entre elas que bloqueie essas portas, seu pacote de batida não conseguirá pasasr. Se for esse o caso, é possível editar o valor de `PCAP_FILTER` em `/etc/fwknop/fwknop.conf` no servidor para definir um valor de porta que seja permitido através do firewall, e depois usar a opção `--Server-port` no comando `fwknop`:

```
$ fwknop -A tcp/22 --Server-port \
330 -a cliente.exemplo.com.br -k \
servidor.exemplo.com.br
```

Para obter mais informações sobre o que está acontecendo, execute o daemon no servidor em modo de depuração: `fwknop -d -v`. Com isso, as tentativas de acesso serão registradas, assim como a abertura do firewall.

Note que a cadeia `FWKNOP_INPUT` do iptables não será criada até que ocorra a primeira tentativa de conexão – então, a ausência dessa cadeia no início não é um problema. Abrindo uma nova janela de terminal e executando o comando `watch -n1 iptables -L -n`, pode-se acompanhar as mudanças no iptables conforme se formem conexões.

Chave GPG

Talvez uma chave em texto puro não seja a forma ideal de autenticação. O fwknop também suporta a autenticação por GPG, mas provavelmente não é interessante usar sua chave GPG normal no servidor, já que a senha para decifrá-la precisa ser armazenada em `/etc/fwknop/access.conf`. Porém, pode-se usar uma chave GPG pré-existente no cliente caso você tenha um.

Caso negativo, vejamos como criar uma nova chave nas duas pontas. Para criar a chave do servidor, use:

```
$ gpg --gen-key
$ gpg --list-keys
```

As opções padrão para o primeiro comando (DSA e Elgamal, chave de 2.048 bits, sem expiração de chave) são boas. Como a chave precisa caber num único pacote, não se deve usar uma maior que 2.048 bits.

Dê um nome e um email apropriados para o servidor (por exemplo, *servidor.exemplo.com.br fwknop* e *fwknop@exemplo.com.br*) e tome nota da senha usada. A geração da chave levará alguns segundos. O GPG vai sugerir que você use o teclado normalmente para gerar entropia enquanto você espera – então, vá verificar seus emails nesse momento. A saída do segundo comando será semelhante a:

```
pub  1024D/AAAAAAAAA 2008-03-07
uid       servidor.exemplo.com.br
➥ fwknop <usuário@exemplo.com.br>
sub  2048g/BBBBBBBBB 2008-03-07
```

Em seguida, é preciso exportar a chave para ASCII com o comando:

```
$ gpg -a --export AAAAAAAA >
➥ server.asc
```

e seguir o mesmo processo no cliente para criar e exportar a chave:

```
$ gpg --gen-key
$ gpg --list-keys
pub   1024D/CCCCCCCC 2008-03-07
uid   test fwknop <usuário-
➥ teste@exemplo.com.br>
sub   2048g/DDDDDDDD 2008-03-07
$ gpg -a --export CCCCCCCC >
➥ client.asc
```

Em seguida, transfira os arquivos `client.asc` e `server.asc`, cada um para a outra máquina, usando o método seguro de sua preferência.

Lembre-se que, caso você já tenha configurado o fwknop no servidor, será necessário bater nas portas adequadas

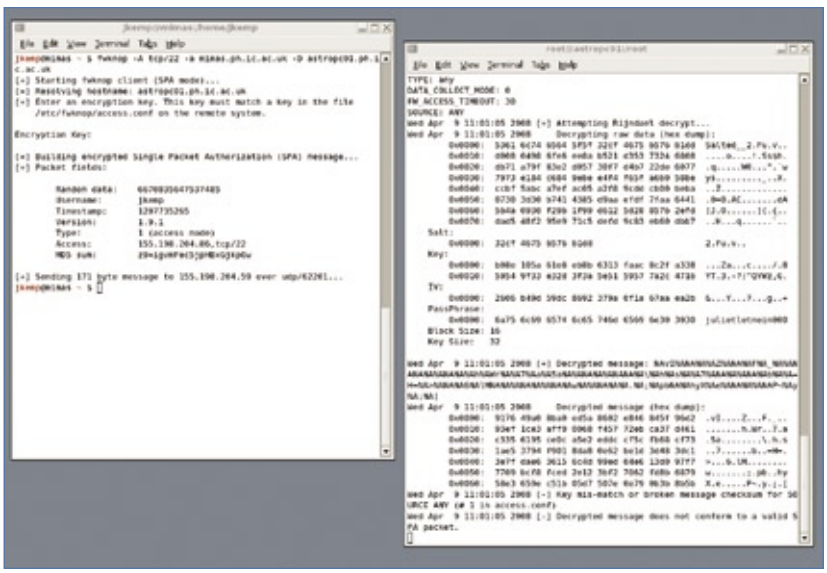


Figura 2 O *fwknop* em ação com uma senha simples.

© Linux New Media do Brasil Editora Ltda.

para conseguir fazer uma transferência por scp ou sftp agora.

Depois, importe e assine cada chave. No cliente, use:

```
$ gpg --import server.asc
$ gpg --edit-key fwknop
Command> sign
Command> save
```

e substitua fwknop na segunda linha pelo nome de usuário dono da chave ou pela ID hexadecimal da chave – AAAAAAAA, neste exemplo.

Feito isso, repita isso no servidor com a chave do cliente. Em seguida, edite o arquivo /etc/fwknop/access.conf no servidor:

```
SOURCE: ANY;
OPEN_PORTS: tcp/22;
DATA_COLLECT_MODE: PCAP;
FW_ACCESS_TIMEOUT: 30;
GPG_HOME_DIR: /root/.gnupg;
GPG_DECRYPT_ID: AAAAAAAA;
GPG_DECRYPT_PW: senhaGpg;
GPG_REMOTE_ID: CCCCCC;
```

A entrada GPG_DECRYPT_ID é o ID da chave GPG e GPG_DECRYPT_PW é a senha dessa chave. Os caracteres depois de GPG_REMOTE_ID compõem o ID da chave GPG do cliente.

Reinicie o fwknop e teste-o. A linha de comando para o cliente é:

```
$ fwknop -A tcp/22 --gpg-recip \
    AAAAAAAA --gpg-sign CCCCCCCC \
    -w -D servidor.exemplo.com.br
```

O ID da chave do servidor é --gpg-recip e o da chave do cliente é --gpg-sign. Depois de informar sua senha GPG, você receberá uma mensagem dizendo que foi enviada uma mensagem para o servidor. Feito isso, já será possível entrar novamente no servidor.

Da forma como está, somente uma chave – e portanto apenas um indivíduo – pode acessar o servidor. Para aumentar esse número, basta criar no arquivo acess.conf mais entradas do tipo:

```
GPG_REMOTE_ID: xxxxxxxx;
```

uma para cada cliente que deve ter permissão para acessar o servidor.

Entretanto, cada chave com permissão de acesso precisa ser importada e assinada no servidor. Além disso, a chave do servidor precisa ser importada e assinada pelo usuário que deseja acessá-lo. Isso pode ser um exagero em máquinas com múltiplos usuários, mas provavelmente seria aceitável para um pequeno número de administradores que precisem se conectar ao servidor (ou, evidentemente, caso se trate de uma máquina com um único usuário).

Além disso, é possível usar um nome particular de usuário ao estabelecer a conexão, com a diretiva:

```
REQUIRE_USERNAME: usuário
```

no arquivo /etc/fwknop/access.conf. De forma semelhante, pode-se exigir um sistema operacional específico ou algum endereço de origem; ou então, é possível alterar o tempo que o fwknopd manterá a porta aberta. A página de manual possui mais detalhes a esse respeito.

Apesar de ser possível usar o fwknop no modo de port knocking simples (chamado de "modo legado" na documentação), esse uso não é recomendado; portanto, não será coberto neste artigo.

Com a opção --Server-cmd comando no fwknop do cliente, é possível enviar um comando para que o servidor o execute como usuário root. É necessário editar o arquivo access.conf no servidor para incluir a diretiva ENABLE_CMD_EXEC, e pode-se restringir quais comandos são permitidos por meio da diretiva CMD_REGEX. Esse recurso pode ser útil para executar certos comandos administrativos a distância ou, talvez, para fins de becape.

Múltiplos usuários

Para implementar o SPA ou outra forma de port knocking em várias máquinas com múltiplos usuários, eles provavelmente apresentarão resistência à idéia, principalmente se estiverem acostumados a usar programas de forma transparente, como o *PuTTY*, por exemplo. Explicar os benefícios pode ajudar, assim como fornecer uma forma fácil de usar scripts apropriados na prática. Usar uma senha de texto puro em vez de chaves GPG pode ser mais fácil nesse caso, embora obviamente quanto mais pessoas souberem uma senha como essa, menos segura ela é.

Alternativamente, pode ser melhor implementar o port knocking somente em certas máquinas. Por exemplo, pode-se criar uma configuração em que certas máquinas de usuários permitam acesso externo (de fora da LAN), mas os servidores proíbam esse acesso e ainda exijam SPA para acesso interno. Isso significa que um indivíduo mal intencionado que conseguir invadir uma das máquinas dos usuários não vai ser capaz de invadir também os servidores.

O port knocking como um princípio, e em particular o SPA, é uma boa forma de restringir o acesso a máquinas vulneráveis e aumentar a segurança. Ele pode não ser adequado para todas as configurações, mas é uma técnica útil para se ter disponível em situações nas quais se deseje uma segurança mais forte. ■

Mais informações

[1] John the Ripper: http://www.openwall.com/john/

[2] CipherDyne: http://www.cipherdyne.org/fwknop/

[3] Download do fwknop: http://www.cipherdyne.org/fwknop/download/

© Linux New Media do Brasil Editora Ltda.

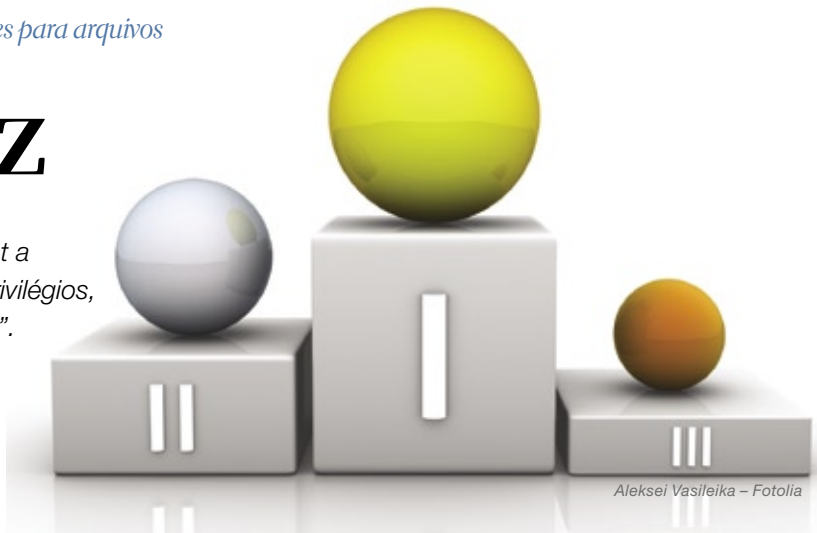
Snow666

Ajuste fino de segurança com as POSIX Capabilities para arquivos

Muito capaz

Se você não quer conceder acesso de root a qualquer programa que precise de mais privilégios, use as POSIX Capabilities em vez do bit "s".

por Ralf Spenneberg



Aleksei Vasileika – Fotolia

Sistemas Linux sempre executam processos sob o usuário que os iniciou. Isso significa que os privilégios do usuário se aplicam a todas as ações do programa. Esse modelo confiável vem enfrentando (e vencendo!) o teste da idade há 35 anos; o Linux simplesmente o herdou do Unix. Porém, esse modelo simples não funciona para todos os casos – por exemplo, se um usuário precisar alterar uma senha. A senha fica guardada em `/etc/passwd` ou `/etc/shadow`; nenhum dos arquivos é gravável por usuários normais e somente o arquivo `/etc/passwd` é legível. Permissões globais de leitura para esses arquivos seriam fatais, pois permitiriam que qualquer usuário ou aplicativo alterasse todas as senhas.

Para resolver esse dilema, Dennis Ritchie, um dos pais do Unix, inventou o bit *SUID*. Os laboratórios da Bell pediram uma patente em 1973, qual foi concedida sob o número 4135240 em 1979[1], antes de ser liberada para o domínio público. O bit SUID permite que um usuário crie um processo que exerce os privilégios do dono do arquivo (ou seja, de outro usuário; veja o **quadro 1**).

Um sistema Linux não funciona sem programas SUID. Contanto que absolutamente todos os programas com o bit *s* ativado sejam feitos e programados de forma segura, os bits *s* são uma bênção. Porém, esse bit constitui um risco considerável: um único erro num programa com ele já basta para que usuários normais consigam escalar seus privilégios e ser alçados à categoria do root. Como processos privilegiados, os programas com SUID de root passam por qualquer verificação de autorização. Entretanto, na realidade, a maioria dos aplicativos não precisa de todos os privilégios cedidos pela conta de root.

Durante muitos anos os fabricantes de sistemas Unix pesquisaram

Quadro 1: Entenda o SUID

Para exemplo do bit SUID em ação, considere o programa de senhas do Fedora 8, que possui as seguintes permissões:

```
-rwsr-xr-x 1 root root 25604 5. Apr 2007 /usr/
➥ bin/passwd
```

O arquivo do programa é de propriedade do usuário *root* e o bit SUID está ativo, como mostra o `s` em vez do `x` na quarta posição. Graças ao bit `s`, o processo resultante do `passwd` tem as permissões efetivas (UID efetivo) do usuário root (ou seja, do proprietário do programa). Ao mesmo tempo, as permissões do verdadeiro usuário são mantidas. Um UID salvo permite que o processo alterne entre esses privilégios a qualquer momento. Se o usuário *ralf* executar o comando `passwd`, pode-se ver as permissões com `ps -o cmd,pid,ppid,euser,ruser, -U ralf`:

```
CMD    PID  PPID USER  RUSER
-bash  2851 2848 ralf   ralf
passwd 2912 2851 root   ralf
```

Essa saída revela que o usuário *ralf* está logado e que esse usuário possui a shel de login `-bash` e executou o comando `passwd`. O comando está rodando com o UID real (*ruser*) *ralf* e o UID efetivo (*euser*) *root*. A ferramenta `passwd` poderia teoricamente usar qualquer um dos privilégios do root para alterar as senhas de outras pessoas ou para desativar o firewall, mas, contanto que o programa esteja funcionando bem, o `passwd` nem sonharia em fazer algo além de mudar a senha do usuário que o está usando. Para reduzir o risco, os distribuidores de Linux reduziram o uso do but SUID nos últimos anos.

© Linux New Media do Brasil Editora Ltda.

Tabela 1: Capacidades POSIX

Número	Nome	Explicação
		Capacidades do rascunho do POSIX
0	CAP_CHOWN	Definir arbitrariamente o dono de um arquivo.
1	CAP_DAC_OVERRIDE	Superar as permissões de acesso do arquivo (DAC, *Discretionary Access Control*); a marca de imutável não é afetada por isso.
2	CAP_DAC_READ_SEARCH	Ler e procurar todos os arquivos e diretórios.
3	CAP_FOWNER	Aplicar as mesmas funções a todos os arquivos cujos donos possam aplicar (como `chmod()` e `utime()`).
4	CAP_FSETID	Ativar o bit SUID para arquivos de terceiros.
5	CAP_KILL	Enviar sinais para processos arbitrários.
6	CAP_SETGID	Assumir um ID de grupo arbitrário.
7	CAP_SETUID	Assumir um ID de usuário arbitrário.
		Extensões específicas para Linux
8	CAP_SETPCAP	Atribuir suas próprias capacidades a processos de terceiros ou removê-las deles.
9	CAP_LINUX_IMMUTABLE	Modificar atributos imutáveis e somente-adição.
10	CAP_NET_BIND_SERVICE	Usar portas privilegiadas.
11	CAP_NET_BROADCAST	Enviar e receber *broadcasts*.
12	CAP_NET_ADMIN	Uma coleção de várias configurações de rede (interface, firewall, roteamento, sockets, ativar estado promíscuo etc.).
13	CAP_NET_RAW	Usar sockets (pacotes IPv4) e pacotes (Ethernet frames) crus.
14	CAP_IPC_LOCK	Bloquear segmentos de memória compartilhada.
15	CAP_IPC_OWNER	Usar IPC (comunicação entre processos) para enviar mensagens a processos arbitrários.
16	CAP_SYS_MODULE	Carregar e descarregar módulos do kernel, modificar o kernel e modificar conjuntos de atribuição de capacidades.
17	CAP_SYS_RAWIO	Usar `ioperm()` e `iopl()` junto com comunicações USB arbitrárias.
18	CAP_SYS_CHROOT	Executar o comando `chroot()`.
19	CAP_SYS_PTRACE	Monitorar e controlar processos com `ptrace()`.
20	CAP_SYS_PACCT	Configurar a contabilidade de processos.
21	CAP_SYS_ADMIN	Várias tarefas administrativas, tais como alterar o domínio e o nome da máquina, gerenciar a montagem de sistemas de arquivos, ativar e desativar o swap, deletar semáforos etc.
22	CAP_SYS_BOOT	Reiniciar o sistema via `reboot()`).
23	CAP_SYS_NICE	Usar o `nice()` para aumentar prioridades, usar o escalonamento de tempo real e alterar a afinidade da CPU de processos de terceiros.
24	CAP_SYS_RESOURCE	Exceder os limites de recursos, como cotas, espaço em disco reservado, tamanho de mensagens IPC etc.
25	CAP_SYS_TIME	Definir a hora do sistema.
26	CAP_SYS_TTY_CONFIG	Configurar dispositivos TTY.
27	CAP_MKNOD	Usar todas as funções `mknod()` na criação de arquivos de dispositivos.
28	CAP_LEASE	Liberar arquivos (veja as liberações com `fcntl()`).
29	CAP_AUDIT_WRITE	Enviar mensagens para o subsistema de auditoria.
30	CAP_AUDIT_CONTROL	Usar o `auditctl()` para configurar o subsistema de auditoria.
31	CAP_SETFCAP	Armazenar as PCaps na capacidade de atributo estendido; ou seja, definir as capacidades de arquivos POSIX.

© Linux New Media do Brasil Editora Ltda.

alternativas à confiança cega em programas. Essa busca resultou em capacidades que definem individualmente os privilégios que um processo pode receber. Até mesmo rascunhos de padrões, como o POSIX.6 e o POSIX 1003.1e, foram aprovados. Após 13 anos de trabalho dos comitês, o IEEE rejeitou o rascunho em 1998[2]. Esse episódio da história vive, hoje, nas *ACLs*[3] e em outra técnica mais poderosa conhecida como *POSIX Capabilities* (*PCaps*).

Apesar de as ACLs já estarem disponíveis para usuários Linux há algum tempo, não havia uma técnica prática para se trabalhar com as PCaps até há pouco. Novas ferramentas e uma atenção renovada a antigas ferramentas puseram as PCaps de volta sob os holofotes.

Root dividido

Desde o kernel 2.2 – para ser mais preciso, o ciclo de desenvolvimento do 2.1 –, o Linux possui privilégios de root divididos em capacidades discretas que um processo pode exercer. O Linux se refere a elas como as *Linux POSIX Capabilities*, ou PCaps, enquanto o Solaris as chama simplesmente de privilégios. No Linux, um processo pode até ativar as PCaps para seções individuais do código (marcando-as como *effective*) ou desativá-las conforme desejado. Isso significa que falhas em outras partes do código não podem utilizá-las. As *capacidades* do Linux estão listadas na **tabela 1**, na página de manual das *capabilities(7)* e nos comentários do cabeçalho ou código-fonte do kernel `/usr/include/sys/capability.h` ou `/usr/src/linux/include/linux/capability.h`.

Traçando limites

Do ponto de vista administrativo, as PCaps costumavam estar disponíveis globalmente pela interface `/proc/sys/kernel/cap-bound`. Isso permitia que o administrador retirasse capacidades globalmente em sistemas em execução e exigia uma reinicialização para restaurá-las. Sem a `CAP_SYS_MODULE`, por exemplo, é impossível carregar módulos do kernel, enquanto a ausência da `CAP_SYS_BOOT` impede a reinicialização suave.

Programas do pacote *libcap1*[4] permitiam que o administrador definisse capacidades para processos em execução. Sistemas MAC complexos como o *App Armor* e o *SELinux* fornecem meios de alterar as capacidades dos processos. Porém, se você deseja apenas atribuir as capacidades exigidas em vez de todos os privilégios de root a programas como *ping*, *passwd* e *mount*, você provavelmente preferirá uma ferramenta mais simples do que algo que crie um modelo de controle de acesso completamente novo no seu sistema. O que o Linux não tinha até o momento era uma forma prática para atribuir os privilégios exigidos (na forma das PCaps) a um programa já na sua chamada de uma forma tão fácil quanto ativar o bit *s* no sistema de arquivos. Infelizmente, o kernel não tinha a infra-estrutura para lidar com isso.

Vários *patches* anteriores para o kernel Linux introduziram essa funcionalidade ([5][6][7]). Um patch de Serge E. Hallyn[5] agora finalmente eliminou todos os obstáculos e foi incluído no kernel 2.6.24. Também faltavam as ferramentas de espaço do usuário para exibir e configurar as PCaps, mas agora Andrew Morgan e KaiGai Kohei preencheram esse espaço.

Controles no espaço do usuário

Muitos anos atrás, Andrew Morgan desenvolveu o pacote *libcap1*[4], que era incompatível com os patches de Serge E. Hallyn.

KaiGai Kohei resolveu esse problema durante um Google Summer of Code; para isso, ele adicionou alguns patches ao pacote de Morgan. No final de 2007, Andrew voltou a trabalhar na libcap e integrou a versão de Kohei.

O resultado disso foi a *libcap2*[8]. O pacote contém as ferramentas *getcap* e *setcap* para ler e definir PCaps. Atualmente, não é necessário instalar as ferramentas de espaço do usuário a partir do código-fonte.

Chris Friedhoff escreveu uma extensa documentação sobre esse tópico[9] .

Uso no Linux

Para suporte às PCaps no sistema de arquivos, é necessário um kernel Linux com a opção *File POSIX Capabilities* ativada. Para distribuições atuais, isso geralmente significa obter o kernel em `[kernel.org]`, compilá-lo e verificar as opções de segurança mostradas na **figura 1**.

Enquanto isso, também é interessante ativar a interface *Kprobes* (**figura 2**). Ela será de grande ajuda quando você precisar identificar as capacidades necessárias para um programa. Ao mesmo tempo, o sistema de arquivos usado precisa suportar atributos estendidos.

Com o novo kernel rodando, ainda serão necessárias as ferramentas de espaço do usuário correspondentes. Elas estão disponíveis como parte da biblioteca *libcap2* no kernel.org[8]. Como de costume, é preciso baixá-las, compilá-las e instalá-las, o que é feito com os comandos padrão `make && make install`, sem necessidade de configuração.

Depois disso, os comandos `setcap` e `getcap` e suas páginas de manual já devem estar disponíveis em seu sistema. Para um rápido teste de capacidades, pode-se usar os comandos `ping` e `traceroute` ou o script `quicktest.sh`.

O `ping` exige privilégios para emitir pacotes ICMP especiais pela LAN. Normalmente é o bit SUID de root que permite isso. O comando `chmod`

© Linux New Media do Brasil Editora Ltda.

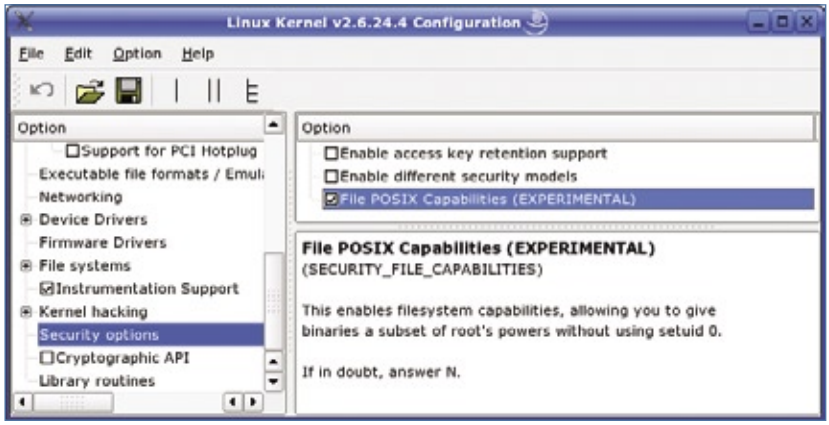


Figura 1 Ativação das PCaps na configuração do kernel.

`u-s /bin/ping` remove esses privilégios, não podendo, depois, ser executado por um usuário sem privilégios:

```
$ ping localhost
ping icmp open socket: The
➥operation is not permitted
```

Para restaurar a capacidade de rodar o `ping` com usuários normais, o comando precisará das PCaps.

Normalmente, para descobrir quais capacidades são necessárias, seria preciso investigar a forma como a ferramenta em questão funciona e depois definir as capacidades exigidas para refletir isso – capacidades demais seriam um risco desnecessário, enquanto capacidades de menos manteriam a ferramenta inútil. Parece complicado, mas a técnica mais simples é usar o módulo *Capable_Probe* de Serge E. Hallyn[10], que usa o *Syslog* para relatar processos e as PCaps de que eles precisam.

Definir necessidades

As mensagens geralmente são gravadas no log `/var/log/messages` ou em `/var/log/kern.log`. Não é uma boa idéia manter o módulo permanentemente carregado, pois ele inundará os logs com um monte de mensagens desnecessárias.

Para o diagnóstico de POSIX Capabilities, simplesmente carregue o módulo verboso; depois, use o comando `modprobe -r capable_probe` para descarregá-lo.

O **exemplo 1** mostra as mensagens que o *ping* causa. O texto puro dos números da capacidade é listado em `/usr/include/linux/capability.h`; a **tabela 1** dá um panorama.

Ganância demais

O *ping* tenta exercer as capacidades 13 (`CAP_NET_RAW`), 7 (`CAP_SETUID`) e 21 (`CAP_SYS_ADMIN`).

A manipulação do SUID (capacidade 7) não é necessária, e o poderoso 21 é um candidato improvável.

A `CAP_NET_RAW` (13) parece mais promissora; o comando a seguir concede essa capacidade ao *ping*:

```
setcap cap_net_raw=ep /bin/ping
```

Após executar o comando, qualquer usuário sem privilégios pode usar o *ping*, o qual é, obviamente, a única capacidade de que o programa precisa. Se o utilitário for incapaz de abrir esse socket, ele tentará outras ações, como revela o módulo *Capable_Probe*. Por causa desse fenômeno razoavelmente freqüente, faz sentido atribuir os privilégios de que o *ping* reclama um após o outro e pensar se faz sentido o programa exigir essa capacidade. Chris Friedhoff[9] tem listas das PCaps necessárias para vários programas.

Conjuntos de capacidades

Para permitir a herança controlada, assim como a ativação dinâmica das PCaps, os programas e processos possuem três conjuntos de capacidades. O resultado dos conjuntos depende se foram aplicados sobre um processo ou um arquivo. A **tabela 2** dá os detalhes dos privilégios *p* Permitido, *e* Efetivo e *i* Herdável. Para as PCaps de arquivos, somente as capacidades nos conjuntos *fP* e *fI* podem fazer parte de um conjunto *fE*; de forma semelhante, em PCaps de processos, somente as capacidades do conjunto *pP* podem estar nos conjuntos *pE* ou *pI*.

Se um aplicativo souber usar as PCaps, ele só vai funcionar com um conjunto Permitido. As PCaps só ativarão os privilégios necessários para

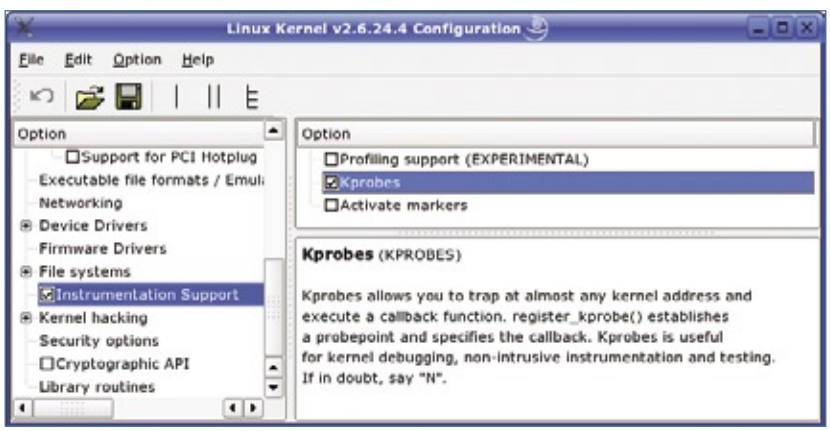


Figura 2 A interface Kprobes permite a identificação posterior de capacidades.

© Linux New Media do Brasil Editora Ltda.

Snow666

Tabela 2: Conjuntos de capacidades POSIX

Conjunto	Arquivo	Processo
P: Permitido	`fP`: O conjunto File Permitted, também conhecido como Forced, define quais PCaps o processo `exec()` resultante terá em qualquer caso.	`pP`: O conjunto Process Permitted descreve as PCaps que um processo pode acessar para ativá-las nos conjuntos Efetivo ou Herdável.
E: Efetivo	`fE`: O conjunto File Effective define quais PCaps estão ativadas imediatamente após o `exec()`. O novo processo pode usar essas capacidades imediatamente.	`pE`: O conjunto Process Effective descreve as PCaps atualmente ativas e que o kernel leva em consideração ao avaliar privilégios.
I: Herdável (Inheritable)	`fI`: O conjunto File Inheritable define quais PCaps o novo processo pode herdar do velho após uma chamada a `exec()`.	`pI`: O conjunto Process Inheritable descreve quais PCaps o processo antigo pode passar para o novo processo a fim de definir o conjunto Permitted para este novo processo.

o programa, e somente quando este precisar deles. O próprio aplicativo define quais capacidades pertencem ao seu conjunto Herdável e quais ele vai potencialmente passar adiante para novos processos. Os programas que não usam PCaps precisam de um conjunto Efetivo por sobre o conjunto Permitido.

O conjunto Efetivo para arquivos dá a um arquivo os privilégios requeridos:

```
setcap cap_net_raw=ep /bin/ping
```

Há três operadores para definir as capacidades POSIX de arquivos: `=`, `+` e `-`. Esses operadores funcionam como aqueles usados pelo comando `chmod`, embora a ordem seja incomum: `cap_net_raw=ep` ativa uma capacidade nos conjuntos Efetivo e Permitido e a deleta do conjunto Herdável. Os operadores `+` e `-` fazem o `setcap` adicionar ou subtrair capacidades dos conjuntos especificados sem alterar os outros conjuntos.

O comando `getcap` lista as PCaps de um arquivo. A listagem normal de um diretório não mostra arquivos com atributos estendidos – esse problema é comum a todos os atributos estendidos e, portanto, também às PCaps. Pelo menos a ferramenta de atributos estendidos `attr -l` mostra que um programa tem um atributo chamado *capability* (ou seja, possui capacidades de arquivo atribuídas).

Só freqüentemente

O comando `passwd` também funciona com as capacidades de sistemas de arquivos – isso elimina a necessidade do bit SUID. O programa requer a possibilidade de contornar os controles de acesso a arquivos. A **tabela 1** mostra qual capacidade lida com isso:

```
setcap cap_dac_override=ep \
➥/usr/bin/passwd
```

Esses privilégios talvez não sejam suficientes num ambiente de produção, principalmente se o programa `passwd` depender do PAM (*Pluggable Authentication Modules*), pois ele precisará de privilégios estendidos para outras tarefas.

Exemplo 1: Capable probe

```
01 Feb 14 12:34:49 station1 kernel: cr_capable: asking for ➥
➥capability 13 for ping
02 Feb 14 12:34:49 station1 kernel: cr_capable: asking for
➥capability 7 for ping
03 Feb 14 12:34:49 station1 kernel: cr_capable: asking for
➥capability 21 for ping
```

Muitos serviços que não possuem o bit SUID mas que são executados pela conta root podem ser induzidos a rodar sob contas sem privilégios com uso das PCaps. A página de Chris Friedhoff[9] dá exemplos disso para *Apache*, *Bind*, *Samba* e o servidor DHCP. Curiosamente, ele combina as técnicas de PCaps e SUID, usando IDs de usuários e grupos separadas para cada serviço. Privilégios de acesso baseados em ACLs para arquivos com PCaps impedem que usuários sem privilégios rodem o Apache ou o Bind com opções indesejáveis.

UID 0 é falha

Infelizmente, as File POSIX Capabilities não conseguem substituir as técnicas de root do SUID em algumas situações. Vários aplicativos simplesmente presumem que estejam sendo executados como root e verificam seu UID. Um teste simples revela essa limitação:

- Dê ao programa um conjunto completo de PCaps;
- Remova o bit SUID ou inicie serviços privilegiados como usuário normal.

Num sistema Fedora 8, por exemplo, o servidor NTP não lida bem com a mudança. Graças ao conjunto completo de capacidades, o programa efetivamente possui os privilégios

© Linux New Media do Brasil Editora Ltda.

que teria como root, apesar de estar rodando sob o usuário *nobody*. Entretanto, o servidor não percebe isso e pede UID=0.:

```
# setcap all=ep /usr/sbin/ntpd
# su - nobody
$ /usr/sbin/ntpd
must be run as root, not uid 99
```

É fácil atribuir todas as PCaps de uma vez com `all=ep`. Essa opção está disponível na libcap 2.08; versões anteriores são afetadas por uma falha na conversão de um valor de 64 bits para 32 bits. Se você usar uma versão anterior, será necessário ativar cada capacidade individualmente. Além dos nomes, o `setcap` felizmente entende números, e uma expressão como `setcap `seq -s, 0 31`=ep /usr/sbin/ntpd` substituirá tudo.

Quando o programa estiver rodando com todas as PCaps e sob a conta de um usuário sem privilégios, pode-se começar a remover as capacidades desnecessárias uma a uma.

Mais privilégios

Capacidades de sistemas de arquivos dão aos usuários sem privilégios acesso a funções que antes precisavam do bit SUID e, portanto, consistem em um risco crítico de segurança. Por exemplo, o administrador do sistema pode atribuir as capacidades pedidas para o programa `ntpdate`, permitindo que usuários normais sincronizem a hora do sistema com um servidor central.

O seguinte comando cuida das capacidades:

```
setcap cap_net_bind_service,cap_
➥sys_time=U ep /usr/sbin/ntpdate
```

Depois de executar esse comando, usuários sem privilégios podem rodar a ferramenta de sincronização. Contudo, essa técnica expõe o sistema a novos perigos: um usuário poderia apontar o comando `ntpdate` para um servidor de hora malicioso e inadvertidamente alterar a hora do sistema da forma errada.

Armadilhas

Enquanto um administrador talvez espere esse tipo de ataque, o software não esperará ter mais privilégios do que o usuário que o executou. Nesse cenário, os programas precisam trabalhar com muito cuidado para evitar que usuários muito criativos exponham falhas.

A única forma genuína de descobrir se um programa está seguro é com uma auditoria intensiva. Na pior das hipóteses, os privilégios do agressor serão definidos pelas capacidades concedidas ao programa comprometido, mas isso ainda é melhor do que um programa todo-poderoso com SUID de root comprometido.

Finalmente

Finalmente as POSIX Capabilities permitem que os administradores gerenciem as PCaps implementadas no kernel por meio do sistema de arquivos. Além disso, o sistema funciona da forma como um administrador Unix/Linux esperaria: o comando `setcap privilégios arquivo` define as PCaps granulares exatamente como `chown root arquivo && chmod u+s arquivo` atribui privilégios de root globalmente. É uma pena que utilitários como o `ls` não entendam essas permissões adicionais, embora o `getcap` esclareça a situação.

Substituir o SUID de root pelas PCaps não elimina qualquer falha de segurança e pode até adicionar vulnerabilidades em programas que não entendam as PCaps. Na pior hipótese, as capacidades ajudarão a reduzir o efeito: em vez de escalar para root, o agressor apenas ganha alguns privilégios adicionais que restringem os danos que seu ataque pode causar no sistema [11]. ■

Mais informações

[1] Patente de Dennis Ritche do SUID (em inglês): http://www.freepatentsonline.com/4135240.html

[2] "Resumo do POSIX.1e" (em inglês): http://wt.xpilot.org/publications/posix.1e/

[3] "Auxílio à lista – Como e por que usar ACLs no sistema de arquivos": http://www.linuxmagazine.com.br/article/2164

[4] libcap1: http://www.kernel.org/pub/linux/libs/security/linux-privs/libcap1/

[5] Patch de Serge E. Hallyn: http://lkml.org/lkml/2006/11/27/170

[6] Patch de Olaf Dietsche: http://www.olafdietsche.de/linux/capability/

[7] Patch de David A. Madore: http://www.madore.org/~david/linux/newcaps/

[8] libcap2: http://www.kernel.org/pub/linux/libs/security/linux-privs/libcap2/

[9] "POSIX Capabilities & File POSIX Capabilities" (em inglês): http://www.friedhoff.org/posixfilecaps.html

[10] Capable_Probe: http://www.friedhoff.org/posixfilecaps/capable_probe.tar.bz2

[11] "POSIX File Capabilities: Parceling the Power of Root", de Serge E. Hallyn (em inglês): http://www.ibm.com/developerworks/linux/library/l-posixcap.html

© Linux New Media do Brasil Editora Ltda.

Proteja sua rede inteira com o IDS OSSEC

Impenetrável

O sistema de detecção de intrusão OSSEC, de origem brasileira, tem destaque internacional por sua grande competência. Aprenda a instalá-lo e a configurá-lo para a sua rede.

por Marcos Aurelio Rodrigues e Rodrigo Montoro

Ao fazer um projeto de rede, um dos pontos fortes da implementação segura é a segurança em profundidade e a diversificação de controles. Uma rede considerada segura possui muitas camadas de defesa, e um dos principais mecanismos de defesa atuais são os IDSs (*Intrusion Detection Systems*). Sistemas IDS analisam exatamente o que um firewall não consegue: eles são capazes de detectar eventos baseados em regras ou anomalias, executando verificações mais inteligentes que um firewall convencional faria.

O uso de mecanismos IDS se tornaram mais conhecidos por causa do sistemas NIDS (*Network Intrusion Detect Systems*), que são capazes de detectar ataques à rede por meio de sensores instalados em um segmento da rede que receba todo o tráfego (hub, tap ou espelhamento de porta no switch). Apesar de muito eficazes, os NIDSs possuem algumas características que ainda permitem que ataques sejam eficazes. Alguns exemplos de evasão são: ataque *DoS*, *session splicing*, fragmentação de pacotes e codificação de caracteres. Além de algumas técnicas de evasão, os NIDSs também não conseguem detectar ataques que estejam trafegando criptografados.

Para complementar o NIDS, existe também o HIDS (*Host Intrusion Detect System*). O HIDS consegue detectar eventos na estações ou servidores e gerar alertas de forma similar a um sistema NIDS, porém o HIDS roda localmente na máquina, conseguindo detectar eventos que o NIDS não perceberia. O HIDS consegue fazer diversas análises, como da integridade do sistema de arquivos, monitoramento de log, monitoramento de registro, detecção de rootkits e resposta ativa. Para suprir a necessidade desse tipo de defesa foi criado o *OSSEC*[1].

O OSSEC é um sistema HIDS multiplataforma de código-fonte aberto. É uma poderosa ferramenta de análise e integração de logs, verificação de integridade de arquivos, detecção de rootkits, alerta em tempo real e resposta ativa. Suporta diversos tipos de logs e pode ser instalado em vários sistemas operacionais, incluindo Linux, OpenBSD, FreeBSD, MAC OS X, Sun Solaris e Microsoft Windows. Mesmo com a aquisição do projeto OSSEC pela empresa Third Brigade[2], o OSSEC continuará a ser Software Livre. Ele é distribuído sob a licença GNU GPL versão 3, conforme publicada pela Free Software Foundation[3].

O projeto dispõe de vários colaboradores ao redor do mundo e possui como principal desenvolvedor o brasileiro Daniel Cid. Seu desenvolvimento é eficiente não so-

Exemplo 1: Download e verificação do OSSEC

```
# wget http://www.ossec.net/files/ossec-hids-1.5.1.tar.gz
# wget #http://www.ossec.net/files/ossec-hids-1.5.1_checksum.txt
# cat ossec-hids-1.5.1_checksum.txt
# md5 ossec-hids-1.5.1.tar.gzMD5 (ossec-hids-1.5.1.tar.gz) = 1269e02
74cb0debce0d4d30b32fba083
# sha1 ossec-hids-1.5.1.tar.gz
SHA1 (ossec-hids-1.5.1.tar.gz) = 4cc2d8d01a59d81f7e8a8c66fb800152d7f
➥2c15c
```


© Linux New Media do Brasil Editora Ltda.

```
Arquivo  Editar  Ver  Terminal  Abas  Ajuda
# tar -zxvf ossec-hids-1.5.1.tar.gz
# cd ossec-hids-1.5.1

# ./install.sh
  ** Para instalação em português, escolha [br].
  ** 要使用中文进行安装, 请选择 [cn].
  ** Fur eine deutsche Installation wohlen Sie [de].
  ** Για εγκατάσταση στα Ελληνικά, επιλέξτε [el].
  ** For installation in English, choose [en].
  ** Para instalar en Español , eliga [es].
  ** Pour une installation en français, choisissez [fr]
  ** Per l'installazione in Italiano, scegli [it].
  ** 日本語でインストールします・選択して下さい・[jp].
  ** Aby instalować w języku Polskim, wybierz [pl].
  ** Для инструкций по установке на русском ,введите [ru].
  ** Za instalaciju na srpskom, izaberi [sr].
  ** Türkçe kurulum için seçin [tr].
  (en/br/cn/de/el/es/fr/it/jp/pl/ru/sr/tr) [en]: br
```

Figura 1 A instalação do OSSEC é multilíngüe.

mente quanto a correções de falhas, mas também para lançamentos de novos recursos.

Operação

Antes de começar a instalação do OSSEC, é necessário entender as diferenças entre seus modos de operação. A escolha dependerá da quantidade de máquinas a serem analisadas na sua rede:

- Local: usado para proteger uma única máquina;
- Agente: usado para proteger máquinas e relatar eventos a um servidor OSSEC;
- Servidor: usado para fazer a coleta de diversos agentes e também eventos de *Syslog* de outros dispositivos (roteadores, firewalls etc.).

Antes de fazer a instalação do OSSEC em um sistema em produção, é recomendável ter certeza de que o sistema não contenha nenhum rootkit instalado. Para tal tarefa, você pode utilizar softwares como *Rootkit Revealer*[4] ou *Chkrootkit*[5].

Para este artigo, foi utilizada a instalação dos modos *Servidor* e *Agente*. O servidor foi instalado em um sistema OpenBSD 4.3 e o agente em um CentOS 5.2.

Instalação

Para começar a instalação, vamos instalar o OSSEC Server. Faça o download do código-fonte e do arquivo de checksum e verifique a integridade do arquivo baixado (**exemplo 1**).

Para a instalação, vamos descompactar o pacote e executar o script de instalação. A instalação é bem intuitiva, e o OSSEC possui suporte a diversos idiomas. A **figura 1** mostra a primeira tela da instalação, e devemos escolher o português brasileiro (`br`). Depois disso, ele mostrará informações sobre a máquina, como nome da máquina, usuário e versão do sistema operacional (**exemplo 2**).

Após confirmar o tipo de sistema operacional e as demais informações da máquina, vamos optar por fazer a instalação do tipo servidor e decidir qual o diretório de instalação do OSSEC (**exemplo 3**).

O OSSEC pode enviar alertas por email. Para isso, devemos configurar a conta de email a ser utilizada e especificar o servidor SMTP para o envio das mensagens. O endereço de email a ser utilizado é o endereço de destino, ou seja, aquele que receberá as notificações do OSSEC.

Para facilitar a operação, o OSSEC tenta descobrir o endereço do servidor SMTP na rede em que está sendo instalado. Neste artigo, o servidor é *exemplo.org*.

Além de o OSSEC fazer o correlacionamento de logs, ele também pode atuar detectando rootkits e verificando a integridade do sistema de arquivos. Para isso, basta habilitar os mecanismos (*engines*) correspondentes, como mostra o **exemplo 5**.

O OSSEC também é capaz de, ao detectar um evento, realizar uma ação e gerar um alerta; porém, além disso, também usaremos sua resposta ativa (**exemplo 6**). A resposta pode ser ativa tanto do lado do servidor quanto do lado do agente, e pode consistir em bloquear um endereço IP ou desabilitar o acesso de um usuário, por exemplo. Para realizar

Exemplo 2: Exibição de detalhes da máquina

```
OSSEC HIDS v1.5.1 Script de instalação - http://www.ossec.net
   Você está iniciando o processo de instalação do OSSEC HIDS.
   Você precisará de um compilador C pré-instalado em seu sistema.
   Qualquer dúvida, sugestões ou comentários, por favor, mande um
e-mail para
   dcid@ossec.net (ou daniel.cid@gmail.com).
    - Sistema: OpenBSD bolivia.exemplo.org 4.3
    - Usuário: root
    - Host: bolivia.exemplo.org
    -- Aperte ENTER para continuar ou Ctrl+C para abortar. --
```


© Linux New Media do Brasil Editora Ltda.

Exemplo 3: Escolha da instalação do tipo servidor

```
1- Que tipo de instalação você deseja (servidor, cliente, local ou
➥ajuda)? servidor
    - Escolhida instalação servidor.
2- Configurando o ambiente de instalação.
   - Escolha onde instalar o OSSEC HIDS [/var/ossec]:
     - A instalação será feita no diretório /var/ossec .
```

Exemplo 4: Configuração do email

```
3- Configurando o OSSEC HIDS.
    3.1- Deseja receber notificações por e-mail? (s/n) [s]: s
     - Qual é o seu endereço de e-mail? marcos@exemplo.org
     - Seu servidor SMTP foi encontrado como: exemplo.org.
     - Deseja usá-lo? (s/n) [s]: n
     - Qual é o ip/host de seu servidor SMTP? 192.168.1.2
```

Exemplo 5: Ativação de mecanismos adicionais

```
3.2- Deseja habilitar o sistema de verificação de integridade?
➥(s/n) [s]: s
     - Syscheck (Sistema de verificação de integridade) habilitado.
    3.3- Deseja habilitar o sistema de detecção de rootkits?
➥(s/n) [s]: s
     - Rootcheck (Sistema de detecção de rootkits) habilitado.
```

a ação, ele pode utilizar utilitários como *iptables*, *ipfilter* ou *tcp wrappers*. Além disso, para diminuir os falsos positivos, é possível criar uma lista amiga (*white list*) a fim de evitar que alguns endereços sejam bloqueados.

Em uma instalação do tipo servidor, o OSSEC pode receber alertas por meio de de um canal seguro (porta 1514) ou então do uso do Syslog (**exemplo 7**) para facilitar a integração de logs de outros dispositivos (note, no entanto, que o Syslog utiliza um canal não criptografado como meio de transmissão).

Após respondermos todas as questões do instalador, o OSSEC utilizará o compilador C do sistema para fazer a instalação, como mostra o **exemplo 8**.

Uso do servidor

Agora já podemos iniciar o OSSEC com o comando `/var/ossec/bin/ossec-control star`. Com o servidor funcionando, instalaremos o agente, mas antes disso é preciso criar uma chave para cada agente que se conectará ao servidor. Essa chave é utilizada

Tabela 1: Configurações contidas em ossec.conf

Tag	Descrição
global	Opções globais utilizadas em todo o sistema.
email_alerts	Opções para envio granular de emails de alertas.
rules	Lista com as regras a serem incluídas na análise.
syscheck	Configurações relacionadas à verificação de integridade do sistema.
rootcheck	Configurações relacionadas à detecção de rootkits.
alerts	Opções de alertas para email e logs.
localfile	Arquivos de log que serão monitorados.
remote	Configurações relacionadas a conexões remotas.
client	Opções relacionadas aos agentes.
database_output	Configuração para log em banco de dados.
command	Programa que será ativado na resposta ativa.
active-response	Configurações de resposta ativa.

© Linux New Media do Brasil Editora Ltda.

para a transmissão através do canal seguro. O **exemplo 9** mostra o utilitário `manage_agents`, que, como diz o nome, gerencia os agentes e suas respectivas chaves.

Instalação do agente

Para instalar o agente do OSSEC, podemos utilizar o mesmo pacote baixado para o servidor. No agente, executamos o mesmo script

Exemplo 6: Configuração de respostas automáticas

```
3.4- Respostas automáticas permitem você executar um comando
        específico baseado nos eventos recebidos. Você pode
        bloquear um endereço de IP ou desabilitar o acesso de
        um usuário específico, por exemplo.
        Maiores informações:
        http://www.ossec.net/en/manual.html#active-response
     - Deseja habilitar o sistema de respostas automáticas?
➥(s/n) [s]: s
     - Sistema de respostas automáticas habilitado.
     - Por padrão, nós podemos habilitar o 'host-deny' e
       o 'firewall-drop'. O primeiro adicionará um host
       ao /etc/hosts.deny e o segundo bloqueará o host
       no 'iptables' (se linux) ou no ipfilter (se Solaris,
       FreeBSD ou NetBSD).
     - Eles podem ser usados para parar 'SSHD brute force
       scans', portscans e outras formas de ataque.
       Você pode também realizar  bloqueios baseados nos
       alertas do snort, por exemplo.
     - Deseja habilitar o firewall-drop? (s/n) [s]: s
       - firewall-drop habilitado (local) para níveis >= 6
     - Lista de endereços que não serão bloqueados pela
➥resposta automática:
        - 192.168.1.1
     - Deseja adicionar mais algum endereço a essa lista?
➥(s/n)? [n]:
```

Exemplo 7: Uso do Syslog para envio e análise dos logs

```
    3.5- Deseja habilitar o syslog remoto (514 udp)? (s/n) [s]: s
     - Syslog habilitado.
    3.6- Ajustando a configuração para analisar os seguintes
logs:
      -- /var/log/messages
      -- /var/log/authlog
      -- /var/log/secure
      -- /var/log/xferlog
      -- /var/log/maillog
   - Se quiser monitorar qualquer outro arquivo, modifique
     o ossec.conf e adicione uma nova entrada para o arquivo.
     Qualquer dúvida sobre a configuração, visite
➥http://www.ossec.net/hids/ .
     - Pressione ENTER para continuar -
```


Snow666

Participe do principal evento de PHP da América Latina

php Conference Brasil '08

27, 28 e 29 de Novembro
UNIFIEO • OSASCO-SP

Palestrantes Internacionais
Presenças confirmadas:

- Christopher Jones — Desenvolvimento de Produto, Oracle
- Todd Trichler — Gerente Sênior de Produto, Oracle Technology Network
- Luke Crouch — Engenheiro de Software, Sourceforge.net

Diamond
Borland · LOCAWEB Serviços de Internet · msdn · ORACLE · ScriptCase · INGRAM MICRO · IBM Premier Business Partner

Silver
dextra Coding your Business

Hospedagem
HostNET

Apoio Institucional e Infra-Estrutura
Centro Universitário FIEO UNIFIEO

Apoio
DICAS-L www.dicas-l.com.br · br-linux.org Ano 10 · DINAMIZE · HTML STAFF

Mídia Oficial
Linux Magazine

Apoio Cultural
TEMPO REAL Livraria do Profissional de Informática

Promoção e Realização
TEMPO REAL Eventos

www.phpconf.com.br

© Linux New Media do Brasil Editora Ltda.

Snow666

Exemplo 8: Instalação automatizada do sistema

```
5- Instalando o sistema
   - Executando o Makefile
...
   - O Sistema é OpenBSD.
   - O script de inicialização foi modificado para executar o OSSEC
➥HIDS durante o boot.
   - Configuração finalizada corretamente.
   - Para iniciar o OSSEC HIDS:
                /var/ossec/bin/ossec-control start
   - Para parar o OSSEC HIDS:
                /var/ossec/bin/ossec-control stop
   - A configuração pode ser vista ou modificada em /var/ossec/etc/
➥ossec.conf
      Obrigado por usar o OSSEC HIDS.
      Se você tiver alguma pergunta, sugestão ou encontrar algum
      "bug", nos contate através do e-mail contact@ossec.net ou
      utilize nossa lista de e-mail:
      ( http://www.ossec.net/main/support/ ).
      Maiores informações podem ser encontradas em
➥http://www.ossec.net
      -  Pressione ENTER para continuar  -
- Adicione as seguintes linhas na configuracao do seu firewall:
       Maiores informações em:
       http://www.ossec.net/en/manual.html#active-response-tools
        table <ossec_fwtable> persist #ossec_fwtable
        block in quick from <ossec_fwtable> to any
        block out quick from any to <ossec_fwtable>
   - Você precisa adicionar cada um dos clientes antes que estejam
➥autorizados a acessar o servidor.
     Execute o 'manage_agents' para adicioná-los ou removê-los:
     /var/ossec/bin/manage_agents
     Maiores informações em:
     http://www.ossec.net/en/manual.html#ma
```

de instalação, mudando apenas o modo de operação em relação ao usado no servidor. Nesse momento, escolhemos novamente o idioma português brasileiro, como mostra a **figura 1**.

Após isso, o script mostrará informações sobre a máquina, exatamente como fez na instalação do servidor. Na estação usada neste artigo, essa etapa é mostrada no **exemplo 10**. Dessa vez, optaremos por instalar o agente e decidiremos em qual diretório será feita a instalação (`/var/ossec/`, como mostra o **exemplo 11**).

Durante a instalação do agente, definimos o endereço do servidor OSSEC já instalado (`192.168.1.3`, no **exemplo 12**) e novamente habilitamos a verificação do sistema de arquivos, a detecção de rootkits e a resposta ativa. O OSSEC então começará novamente o processo de compilação e configuração do sistema (**exemplo 13**).

Comunicação servidor/agente

Para fazer a configuração da comunicação entre o servidor e o agente, é necessário importar a chave do servidor no agente. Para isso, abra dois terminais, um conectado ao servidor e outro ao agente. Primeiramente, vamos extrair a chave no lado servidor (**exemplo 14**) e em seguida copiá-la para o cliente (**exemplo 15**).

Ao final da importação, é necessário reiniciar tanto o servidor quanto o agente com o comando:

```
/var/osec/bin/ossec-control
➥restart
```

Configuração

Após a instalação do OSSEC, não há mais muito o que alterar para iniciar a operação. A estrutura de configuração do OSSEC é em XML, e todos os arquivos que a compõem se localizam no subdiretório `etc/` do diretório onde o OSSEC foi instalado. O arquivo principal é o `ossec.conf` e toda configuração fica dentro da seção principal `<ossec_config>`. Alguns sub-elementos dela são:

- `<global>`: opções gerais em instalações dos tipos servidor e local.
- `<alerts>`: opções de alerta do tipo email e log.
- `<remote>`: opções relacionadas a conexões remotas e de agentes (somente em instalações do tipo servidor)
- `<localfile>`: configuração relacionada a logs monitorados.

O **exemplo 16** mostra duas seções que definem o monitoramento do tipo de log syslog para alertas de segurança de um sistema Linux.

Alertas

Todo alerta possui um nível entre 0 e 15, sendo 15 o mais grave e 0 o mais trivial. Para definir quando um alerta deve ser

© Linux New Media do Brasil Editora Ltda.

Exemplo 9: Gerenciamento de agentes

```
# /var/ossec/bin/manage_agents


****************************************
* OSSEC HIDS v1.5.1 Agent manager.      *
* The following options are available: *
****************************************
    (A)dd an agent (A).
    (E)xtract key for an agent (E).
    (L)ist already added agents (L).
    (R)emove an agent (R).
    (Q)uit.
Choose your action: A,E,L,R or Q: A
- Adding a new agent (use '\q' to return to the main menu).
   Please provide the following:
    * A name for the new agent: paraguai
    * The IP Address of the new agent: 192.168.1.4
    * An ID for the new agent[001]:
Agent information:
    ID:001
    Name:paraguai
    IP Address:192.168.1.4
Confirm adding it?(y/n): y
Agent added.
```

Exemplo 10: Informações da máquina agente

```
OSSEC HIDS v1.5.1 Script de instalação - http://www.ossec.net
  Você está iniciando o processo de instalação do OSSEC HIDS.
  Você precisará de um compilador C pré-instalado em seu sistema.
  Qualquer dúvida, sugestões ou comentários, por favor, mande
➥um e-mail para
  dcid@ossec.net (ou daniel.cid@gmail.com).
   - Sistema: Linux paraguai.exemplo.org 2.6.18-53.el5
   - Usuário: root
   - Host: paraguai.exemplo.org
   -- Aperte ENTER para continuar ou Ctrl+C para abortar. --
```

Exemplo 11: Tipo de instalação cliente e local de instalação

```
1- Que tipo de instalação você deseja (servidor, cliente,
➥local ou ajuda)? cliente
    - Escolhida instalação cliente.
2- Configurando o ambiente de instalação.
   - Escolha onde instalar o OSSEC HIDS [/var/ossec]:
      - A instalação será feita no diretório /var/ossec .
```

enviado por email e quando deve ser transmitido por log, podemos alterar o conteúdo das tags `<log_alert_level>` e `<email_alert_level>`. A forma exibida na seção `<alerts>` do **exemplo 16** gera logs somente para alertas acima de 1, enquanto somente alertas acima de 7 serão enviados por email.

É necessário também citar alguns arquivos de configuração e suas respectivas sintaxes.

O diretório principal das configurações do OSSEC, considerando os caminhos de instalação escolhidos neste artigo, é `/var/ossec/etc/`, e os arquivos são:

- `ossec.conf`: arquivo principal de configuração do OSSEC. Contém as configurações descritas na **tabela 1**.
- `decoder.xml`: arquivo que contém os decodificadores dos logs recebidos pelo OSSEC. Os decodificadores separam algumas informações para a segunda parte das análises que são as regras.
- `internal_options.conf`: possui algumas informações de configuração para depuração e ajuste fino; deve ser modificado com cuidado.

Outro diretório importante é o `/var/ossec/rules/`, que contém as regras responsáveis por analisar o conteúdo dos logs. Cada arquivo dentro dele contém um conjunto de regras para definição da gravidade das mensagens no log de um aplicativo, assim como um identificador geral para cada mensagem do log.

Por exemplo, o arquivo `apache.rules.xml` define, entre outras centenas de regras, que a mensagem `authentication failed`, quando encontrada nos logs do Apache, deve receber o nível 5 e ser categorizada entre as mensagens de falha na autenticação. Enquanto isso, em `attack_rules.xml` é

© Linux New Media do Brasil Editora Ltda.

Snow666

Exemplo 12: Recursos ativos no agente

```
3- Configurando o OSSEC HIDS.
    3.1- Qual é o endereço de IP do servidor OSSEC HIDS?:
➥192.168.1.3
     - Adicionando IP do servidor 192.168.1.3
    3.2- Deseja habilitar o sistema de verificação de integridade?
➥(s/n) [s]: s
     - Syscheck (Sistema de verificação de integridade) habilitado.
    3.3- Deseja habilitar o sistema de detecção de rootkis?
➥(s/n) [s]: s
     - Rootcheck (Sistema de detecção de rootkits) habilitado.
    3.4 - Deseja habilitar o sistema de respostas automáticas?
➥(s/n) [s]: s
```

Exemplo 13: Instalação automatizada do agente

```
3.5- Ajustando a configuração para analisar os seguintes logs:
      -- /var/log/messages
      -- /var/log/secure
      -- /var/log/maillog
   - Se quiser monitorar qualquer outro arquivo, modifique
     o ossec.conf e adicione uma nova entrada para o arquivo.
     Qualquer dúvida sobre a configuração, visite
➥http://www.ossec.net/hids/ .
     - Pressione ENTER para continuar -

5- Instalando o sistema
   - Executando o Makefile
...
   - Para se comunicar com o servidor, você primeiro precisa
     adicionar este cliente a ele. Quando você tiver terminado,
     use a ferramenta 'manage_agents' para importar a chave de
     autenticação do servidor.
     /var/ossec/bin/manage_agents
     Maiores informações em:
     http://www.ossec.net/en/manual.html#ma
```

definido que mensagens que coincidirem com a expressão regular:

```
ftpd[\d+]: \S+ FTP LOGIN FROM \.+
➥0bin0sh
```

devem receber nível 14 (alto), pois caracterizam um *exploit* de estouro de buffer contra versões do servidor FTP *wu-ftpd* anteriores à 2.6, e serão agrupadas como tentativa de exploit (`exploit_attempt`).

Além desses arquivos, há dezenas de outros, abrangendo desde o servidor VoIP *Asterisk* até o monitor de máquinas virtuais *VMware*, passando por roteadores Cisco com sistema *IOS*, regras de firewall, bancos de dados *MySQL*, sistema *PAM* e servidores de email Microsoft *Exchange*.

Evidentemente, é possível e relativamente fácil criar novas regras e decodificadores, uma vez que seja compreendida a sintaxe dos arquivos.

Exemplo 14: Extração da chave do servidor

```
# /var/ossec/bin/manage_agents
**********************************
* OSSEC HIDS v1.5.1
Agent manager.       *
* The following options
are available: *
**********************************
     (A)dd an agent (A).
     (E)xtract key for an agent E).
     (L)ist already added agents (L).
     (R)emove an agent (R).
     (Q)uit.
Choose your action: A,E,L,R or Q: e
Available agents:
     ID: 001, Name: paraguai, IP:
➥192.168.1.4
Provide the ID of the agent to extract
➥the key (or '\q' to quit): 001
Agent key information for '001' is:
MDAxIHBhcmFndWFpIDE5Mi4xNjguMS40IDk
➥1YmRjMzM4Y2M2Mzk4M2ExNmI4
➥MmE4ZjE1N2RkY2EzYzBjNDA0MjNhOGJkMj
➥ljZTFiZjhjNjllOTdiMjEyYjQ=
```

Conclusão

Atualmente vivemos uma era da segurança da informação em que os ataques se tornam mais complexos e inteligentes a cada dia, sendo que a segurança em perímetro vem se tornando algo obrigatório nas empresas. Há pouco tempo atrás, era comum utilizar sistemas HIDS somente em máquinas expostas à Internet e nos servidores mais críticos. Porém, visto o aumento dos ataques a estações de trabalho e malwares com suporte a SSL, é crescente a necessidade de rodar HIDS em todas as máquinas, e o OSSEC certamente é uma das melhores soluções nesse campo, por suportar uma grande quantidade de sistemas operacionais e oferecer as vantagens do Software Livre. ■

© Linux New Media do Brasil Editora Ltda.

Exemplo 15: Importação da chave pelo agente

```
[root@paraguai ossec-hids-1.5.1]# /var/ossec/bin/manage_agents
****************************************
* OSSEC HIDS v1.5.1 Agent manager.     *
* The following options are available: *
****************************************
    (I)mport key from the server (I).
    (Q)uit.
Choose your action: I or Q: I
* Provide the Key generated by the server.
* The best approach is to cut and paste it.
*** OBS: Do not include spaces or new lines.
Paste it here (or '\q' to quit): MDAxIHBhcmFndWFpIDE5Mi4xNjguMS40IDk1Y
mRjMzM4Y2M2Mzk4M2ExNmI4MmE4ZjE1N2RkY2EzYzBjNDA0MjNhOGJkMjljZTFiZjhjNjl
lOTdiMjEyYjQ=
Agent information:
    ID:001
    Name:paraguai
    IP Address:192.168.1.4
Confirm adding it?(y/n): y
Added.
```

Exemplo 16: Monitoramento de syslog para alertas em Linux

```
<localfile>
     <log_format>syslog</log_format>
     <location>/var/log/secure</location>
</localfile>
  <alerts>
     <log_alert_level>1</log_alert_level>
     <email_alert_level>7</email_alert_level>
</alerts>
```

Mais informações

[1] Página do OSSEC: http://www.ossec.net

[2] Third Brigade: http://3rdbrigade.com

[3] FSF: http://www.com.br.com.br

[4] Rootkit Revealer: http://www.fsf.org

[5] Chkrootkit: http://tinyurl.com/465pg7

Sobre os autores

Marcos Aurelio Rodrigues (deigratia33@gmail.com) é consultor de segurança e infra-estrutura, certificado CCNA, MCSO e Comptia Security +, e possui mais de oito anos de experiência em administração e infra-estrutura de sistemas e rede, além de ser mantenedor da comunidade Snort-br.

Rodrigo "Sp0oKeR" Montoro (spooker@gmail.com) é certificado LPI, RHCE, SnortCP e MCSO, trabalha como Analista de Segurança e possui dez anos de experiência em sistemas de código aberto, em especial ferramentas de segurança, com as quais tem forte envolvimento.

Snow666

Uma empresa tão livre quanto a sua imaginação.

Pensando na sua liberdade de pensamento, a F13 Tecnologia oferece produtos, soluções e serviços em Linux e Softwares livres, como suporte técnico presencial ou remoto e cursos de formação com certificação, tais como:

- Formação Linux com ênfase na LPI (4 módulos totalizando 160 horas)
- Formação PHP (3 módulos totalizando 120 horas)
- Firewall Avançado (40 horas)
- Controle de versões com CVS, SVN e Trac (8 horas)
- Virtualização com Xen (40 horas)
- Serviço de diretórios com OpenLdap (40 horas)
- Correio Eletrônico Avançado (40 horas)
- Voip & Asterisk com ênfase em DialPlan (40 horas – Curso ministrado por instrutor com certificação OCAP)
- Administração de Bancos de Dados Livres (PostgreSQL e MySQL – 40 horas)

Linux Professional Institute Approved Training Partner

F13 TECNOLOGIA

(85) 3252.3836

www.f13.com.br

© Linux New Media do Brasil Editora Ltda.

Snow666

TUTORIAL

Becape fácil, rápido e bonito para desktops

Becape para todos

O BackupPC faz becapes pela rede para várias plataformas. Mesmo sendo destinado a desktops, ele é rápido e altamente configurável.

por David Nalley

Plataformas de becape pela rede costumam ser pouco amigáveis, em parte por causa das complexidades da lógica de agendamento e também do gerenciamento de mídias. A amistosidade pode ser um item de luxo em sistemas corporativos. O projeto *BackupPC*[1] preenche com elegância o nicho do becape, atuando agilmente pela rede e por várias plataformas e transportes.

O BackupPC segue a tradição do Unix de pequenos programas que realizam uma única tarefa muito bem. Assim como outros utilitários clássicos do Unix, o BackupPC se utiliza do poder de outros aplicativos em vez de tentar reinventar a roda. Ele suporta vários protocolos para clientes Windows e Unix (ou *Unix-like*) – desde Rsync e SMB/CIFS até tar e túneis Rsync sobre SSH. O objetivo é o agendamento eficiente e um processo facilitado de recuperação.

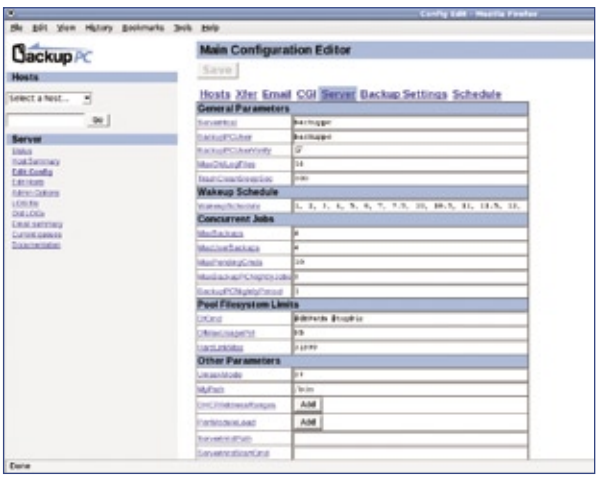


Figura 1 No editor de configuração é possível ajustar vários aspectos importantes do servidor.

O BackupPC tem uma comunidade de usuários ativa, com listas de email e um wiki gerado pelos próprios usuários. Além disso, o projeto ainda é liderado por seu autor original, Craig Barratt. Apesar de a ferramenta existir desde 2001 e ser relativamente madura, a versão mais recente – BackupPC 3.1.0 – parece estar alcançando agora a novos usuários.

Benefícios

Um dos principais recursos do BackupPC é a "desduplicação" de dados. Em sistemas tradicionais de becape, manter múltiplas cópias de arquivos que não tenham sido alterados durante o intervalo entre becapes completos exige o armazenamento da mesma informação repetidas vezes. O problema fica ainda maior quando se faz becape de vários desktops de usuários finais que estão nas mesmas listas de emails internos da empresa, possuem a mesma planilha e outros documentos em comum. O BackupPC resolve esse problema com uma verificação em duas camadas. A primeira localiza arquivos com nomes repetidos e calcula seus *hashes* para conferir se são idênticos. Em caso positivo, ela move uma única cópia do arquivo para o "pool" e cria hardlinks para

© Linux New Media do Brasil Editora Ltda.

cada instância do arquivo no becape. Os resultados são surpreendentes: num primeiro teste com oito máquinas (fazendo becapes não compactados e retenção de dois becapes completos e seis incrementais), o espaço total ocupado foi de aproximadamente 1 TB, mas a técnica do BackupPC diminuiu esse espaço para 675 GB.

O BackupPC também possui diversos recursos de agendamento interessantes, como a possibildade de priorizar becapes, por exemplo. Por padrão, o BackupPC acorda a cada hora e identifica todos os computadores que não tenham completado um becape no intervalo especificado. Ele também verifica quais máquinas estão na rede e, após cruzar essas duas listas, prioriza a lista de máquinas disponíveis com base no tempo desde o último becape. Outros fatores também podem influenciar essa lista de prioridades. Por exemplo, uma máquina que esteja 24 horas por dia na rede geralmente fica atrás de outra com presença mais esporádica.

Um recurso potencialmente positivo (mas com possibilidades de mau uso, claro) é a possibilidade de os próprios usuários finais realizarem a restauração sem precisarem comunicar qualquer coisa ao administrador do sistema. Em becapes de grande escala, gasta-se muito tempo na restauração de arquivos apagados ou danificados. Portanto, se o usuário precisar encontrar uma versão específica do arquivo, um processo de restauração que alivie o trabalho do administrador pode ser muito desejável.

Para isso, o BackupPC oferece uma interface web amigável com uma árvore de arquivos e diretórios para cada becape. Os usuários podem selecionar um único arquivo ou múltiplos na árvore para o BackupPC restaurar sem necessidade de incomodar o administrador. O BackupPC até mesmo verifica se o usuário possui as permissões de acesso necessárias para visualizar o arquivo antes de iniciar a recuperação.

Os usuários também possuem certo controle quanto ao momento de iniciar o becape (completo ou incremental) ou se desejam retirar sua máquina da lista de becape por algumas horas.

Instalação

A instalação do BackupPC é relativamente indolor, já que ele está incluído no repositório da maioria das distribuições mais populares. Porém, como eventualmente esses repositórios não contam com a versão mais recente, ou requerem alguma instalação especial, vamos abordar a instalação a partir do código-fonte.

Antes de fazer a instalação, é essencial considerar o espaço em disco e como configurá-lo. Como o BackupPC trata a "desduplicação" com a criação de hardlinks a partir da localização do arquivo no local onde os becapes estão armazenados, é necessário que o becape seja guardado num único sistema de arquivos. Isso não significa que não se possa usar LVM ou RAID (via software ou hardware), mas que deve ser usado um único sistema de arquivos para armazenar todos os becapes.

O BackupPC testa a criação desses hardlinks em cada inicialização. É preciso saber o ponto de montagem do sistema de arquivos durante a instalação.

Os dois próximos passos são realmente simples e consistem em criar um usuário para execução do BackupPC e instalar as dependências do software. Um passo não coberto é o servidor web, que já deve estar instalado:

```
# adduser backuppc
# yum install perl-Compress-Zlib
➥perl-Archive-Zip perl-XML-RSS
➥perl-File-RsyncP
```

Com as dependências resolvidas, basta baixar o código-fonte em [1], descompactá-lo e rodar o script de instalação:

```
$ tar -zxvf BackupPC-3.1.0.tar.gz
$ cd BackupPC-3.1.0
$ su -c "perl
./configure.pl"
```

Isso inicia o instalador, que faz a configuração e instalação básicas do BackupPC. As respostas padrão são boas, com algumas exceções. O diretório de dados deve ser o ponto de montagem do sistema de arquivos com os becapes (por exemplo, `/data/BackupPC/`). Além disso, pode

Exemplo 1: Scripts de inicialização

```
$ su -c "cp linux-backuppc /etc/init.d/backuppc"
$ su -c "chkconfig --add backuppc"
$ su -c "chkconfig --level 345 backuppc on"
$ su -c "chkconfig --list backuppc"
$ su -c "service backuppc start"
```

Exemplo 2: Arquivo hosts

```
host      dhcp  user   moreUsers # <- não edite
➥esta linha
nalleyt61  0    david            # <- com IP estático
host2      1    bill   jeff,fred # <- IP por DHCP
```


© Linux New Media do Brasil Editora Ltda.

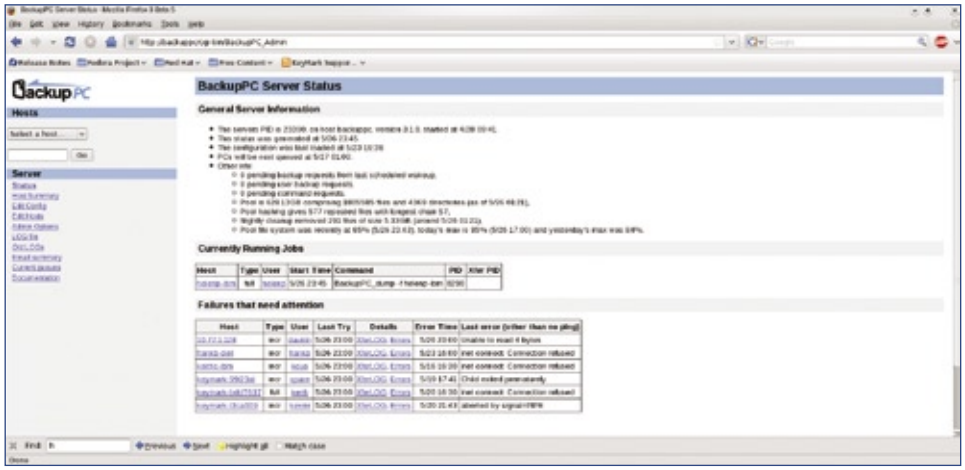


Figura 2 Verifique o status dos becapes em andamento e falhas passadas e atuais.

A configuração do BackupPC é feita em dois arquivos sob `/etc/BackupPC/`. O arquivo `hosts` detalha a identidade das máquinas a serem copiadas, enquanto `config.pl` controla a configuração do servidor.

O arquivo `hosts` lista, além dos nomes das máquinas, os usuários autorizados para ela (**exemplo 2**). A autenticação na interface web será coberta mais adiante, assim como os usuários autorizados. Os pontos principais do **exemplo 2** são as configurações nos casos de IP estático e DHCP. Se sua máquina receber o IP por DHCP, deve-se usar `0` na configuração do DHCP para que o BackupPC use DNS para encontrar a máquina. O valor `1` faz com que o programa use o *nmblookup* para encontrar a máquina via *NetBIOS*.

ser preciso entrar o caminho correto para o diretório de CGI (`/var/www/cgi-bin/`, em geral).

Para o BackupPC iniciar automaticamente, é preciso adicionar scripts de inicialização ao sistema. No subdiretório `init.d/` há scripts de inicialização para diversas distribuições. Basta copiar o script adequado à distribuição para o diretório `/etc/init.d/` e ordenar que o sistema o execute na inicialização (**exemplo 1**).

Configuração

Apesar do proesso de instalação lidar com os elementos básicos de configuração, há outras opções disponíveis pela interface web ou pela linha de comando.

Exemplo 3: Opções do arquivo config.pl

```
01 $Conf{CgiAdminUsers} = 'ke4qqq,bill'; #define os usuários administrativos como bill e ke4qqq
02 $Conf{XferMethod} = 'smb'; #define o método padrão de transferência como smb (outras opções: rsync,
➥rsyncd, or tar).
03 $Conf{SmbShareName} = 'C$'; #nome do compartilhamento smb padrão. Nesse caso, o compartilhamento do
➥administrador para a unidade C
04 $Conf{SmbShareUserName} = 'david'; #usuário do compartilhamento smb
05 $Conf{SmbSharePasswd} = 'minhasenha'; #senha do compartilhamento smb
06
07 #Se você estiver fazendo becape principalmente de máquinas Linux, as configurações a seguir também são
apropriadas.
08
09 $Conf{XferMethod} = 'rsync' #define o método padrão de transferência como rsync
10 $Conf{RsyncShareName} = '/'; #especifica o diretório a ser copiado
11 $Conf{RsyncClientRestoreCmd} = '$sshPath -q -x -l root $host $rsyncPath $argList+'; # faz o BackupPC
usar ssh e
12 # depois rsync. Você precisará configurar
13 # chaves ssh para o usuário backuppc
14 # e depois copiá-las.
```


© Linux New Media do Brasil Editora Ltda.

O arquivo `config.pl` é configurado para acordar a cada hora, procurar quais máquinas precisam ser copiadas e também para fazer um becape completo aproximadamente a cada sete dias, além de um becape incremental diário (**figura 1**).

É possível ajustar essas e outras configurações. O manual e o arquivo de configuração dão detalhes de todas as opções, mas somente umas poucas são obrigatórias para começar a fazer seus becapes no Linux ou Windows. Também é importante lembrar que é possível fazer modificações por cada máquina.

O que precisa ser configurado obrigatoriamente é o usuário administrador e como os becapes serão transferidos no ambiente (veja o **exemplo 3**). Não é recomendável utilizar o usuário root para os becapes, mas uma conta com poucos privilégios e com a permissão de usar o *sudo* para fazer o Rsync. Com o usuário *backuppc*, também é preciso entrar na máquina cliente por SSH para que ela se torne conhecida (isto é, sua chave pública seja acrescentada ao arquivo `known_hosts`).

Com o `visudo`, acrescente uma linha ao arquivo `/etc/sudoers` da máquina cliente:

```
backuppc ALL=NOPASSWD:
➥/usr/bin/rsync
```

e, em seguida, modifique os argumentos do comando para usar o sudo na chamada ao Rsync:

```
$Conf{RsyncClientCmd}
➥= '$sshPath -l backup
➥$host nice -n 19 sudo
➥/caminho/do/rsyncSend
➥$argList+';
```

As configurações devem ir além dessas, mas elas já são suficientes para fazer o becape de máquinas Linux ou Windows com compartilhamentos Samba expostos.

Apesar de ser possível configurar vários outros aspectos, como as exclusões de arquivos e diretórios e os níveis de compressão, o último item requerido é a configuração da interface web. A interface web foi automaticamente instalada, mas é preciso configurar a autenticação nele, além de ser necessária uma forma de autenticar os usuários do arquivo `hosts` e os usuários administrativos. Como o Apache é usado para a autenticação, existem diversas formas para esse processo. Por exemplo, é possível usar LDAP, *Active Directory*, autenticação básica ou qualquer outro modo suportado pelo Apache.

Apesar de a documentação de Barratt chegar até a configuração da autenticação por LDAP, este artigo usará o método de *digest*, que requer a adição da seção do **exemplo 4** ao arquivo `httpd.conf`. Em seguida, execute o comando:

```
# htpasswd -c /etc/httpd/conf/
➥passwd ke4qqq
# htpasswd /etc/httpd/conf/passwd
➥bill
```

Note que a opção `-c` só é usada na criação do arquivo de senhas e que portanto deve ser omitida nos usuá-

Exemplo 4: Modificação do arquivo httpd.conf

```
01 <Location /cgi-bin/BackupPC_Admin> # <--- altere o caminho
➥conforme sua necessidade
02    AuthType Basic
03    AuthName "login do BackupPC"
04    AuthUserFile /etc/http/conf/passwd # <--- altere o caminho
➥conforme sua necessidade
05    require valid-user
06 </Location>
```

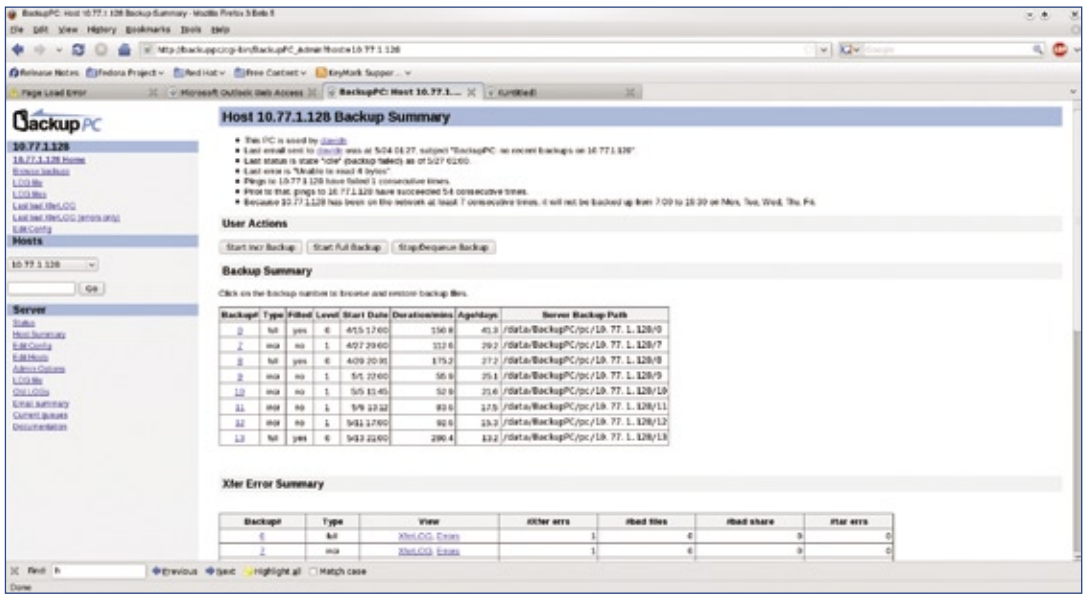


Figura 3 A página de status permite que se inicie e pare becapes, ou ainda que eles sejam retirados da fila...

© Linux New Media do Brasil Editora Ltda.

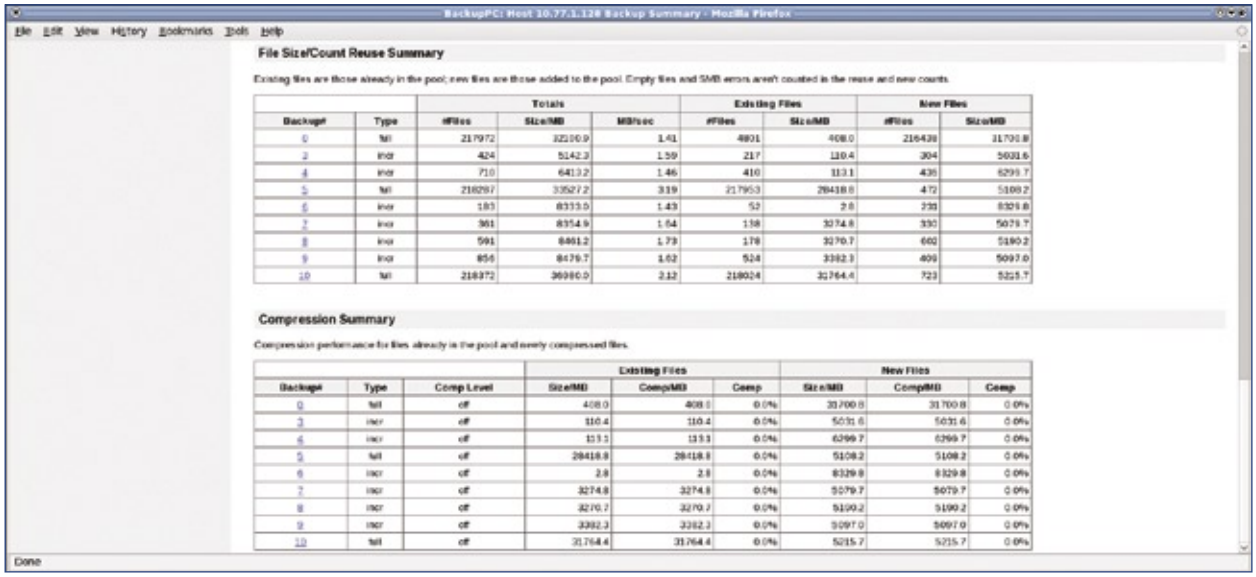


Figura 4 ... além de também ser possível verificar as estatísticas de cada um.

rios posteriores. O comando pedirá a senha do usuário *ke4qqq*.

Depois de recarregar o httpd e iniciar o BackupPC, já é possível iniciar um navegador e apontá-lo para `http://backuppcserver/cgi-bin/BackupPC_Admin/authenticate` com o usuário criado para entrar na interface web. Se você não for um usuário administrativo, só terá acesso às máquinas nas quais está listado como usuário no arquivo da máquina.

Interface web

Depois que o servidor estiver configurado, é interessante conhecer um pouco da interface do usuário. Na maioria das instalações, a URL deve ser `http://servidorbecape/cgi-bin/BackupPC_Admin`. Esse endereço mostra a página de status do servidor (**figura 2**). O usuário administrador verá todos os becapes em execução no momento e todas as falhas que requerem atenção. A maioria dessas falhas são causadas por máquinas que são desligadas pelo usuário no meio do procedimento. Além disso, são exibidas estatísticas sobre o servidor.

Um dos aspectos positivos da interface é que quase todas as referências a becapes e máquinas são clicáveis e levam ou à página de status ou à página de navegação do becape. Também se vê um link para a documentação, servido localmente.

No lado esquerdo ficam os links para navegação. O menu *drop-down* no alto mostra uma lista de máquinas, e a caixa de busca embaixo permite a busca por nome. Depois de selecionada uma máquina, a interface o leva à página de status desta, que mostra uma lista de todos os becapes já terminados, assim como estatísticas sobre cada um, como tamanho e data (**figuras 3 e 4**). Na página de status de cada máquina, também se pode iniciar e parar becapes e retirar uma máquina específica da fila durante algumas horas. O usuário

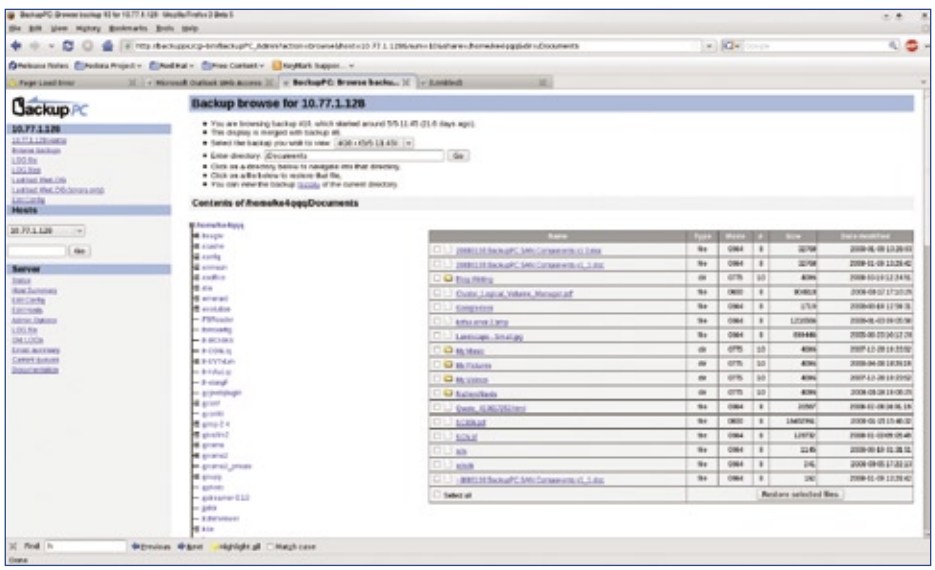


Figura 5 Na página de navegação de becapes, pode-se selecionar um arquivo individual ou múltiplos arquivos.

© Linux New Media do Brasil Editora Ltda.

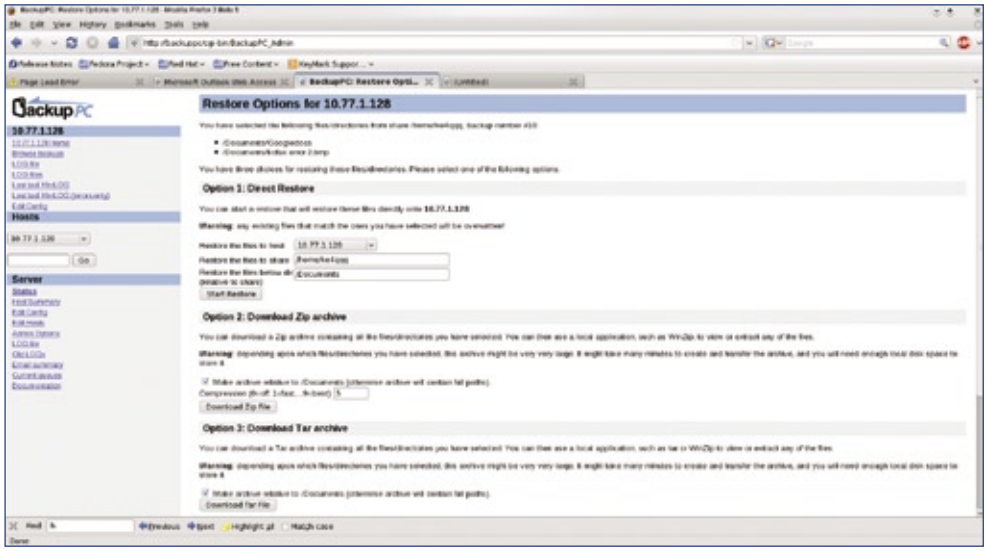


Figura 6 A interface do usuário facilita imensamente a restauração de arquivos.

final dono da máquina no arquivo `hosts` também consegue iniciar, parar e retirar da fila os becapes.

Clicar na lista de becapes apresentada na página de cada máquina leva à interface de recuperação. Na página de navegação do becape existe uma árvore de navegação no lado esquerdo do sistema de arquivos e uma lista de arquivos à direita (**figura 5**). Nessa interface, pode-se clicar num arquivo individual e baixá-lo diretamente no navegador, ou selecionar múltiplos arquivos marcando suas respectivas caixas e depois clicar em *Download selected files*.

Na página seguinte, pode-se especificar os arquivos a serem comprimidos ou se se deseja que o BackupPC restaure os arquivos diretamente na máquina pelo mesmo método de transferência usado para fazer o becape. Note que o método de transferência precisará de acesso de escrita ao sistema de arquivos (**figura 6**).

Outra página freqüentemente acessada pelo administrador é a *Host Summary*, ou resumo da máquina (**figura 7**), que fornece uma tabela colorida com todas as máquinas configuradas no servidor. Além das cores, que indicam seu status atual, é possível ver tamanho dos últimos becapes incremental e completo, a velocidade do becape e o tempo desde o último becape. Isso oferece um rápido panorama de todas as máquinas e permite a identificação de possíveis problemas e tendências.

O BackupPC oferece uma solução para becapes abrangente e também amigável. Quem procura uma solução de becape para sua empresa – principalmente para desktops de usuários finais – deve considerar seriamente essa alternativa. ■

Mais informações

[1] BackupPC: http://backuppc.sourceforge.net/

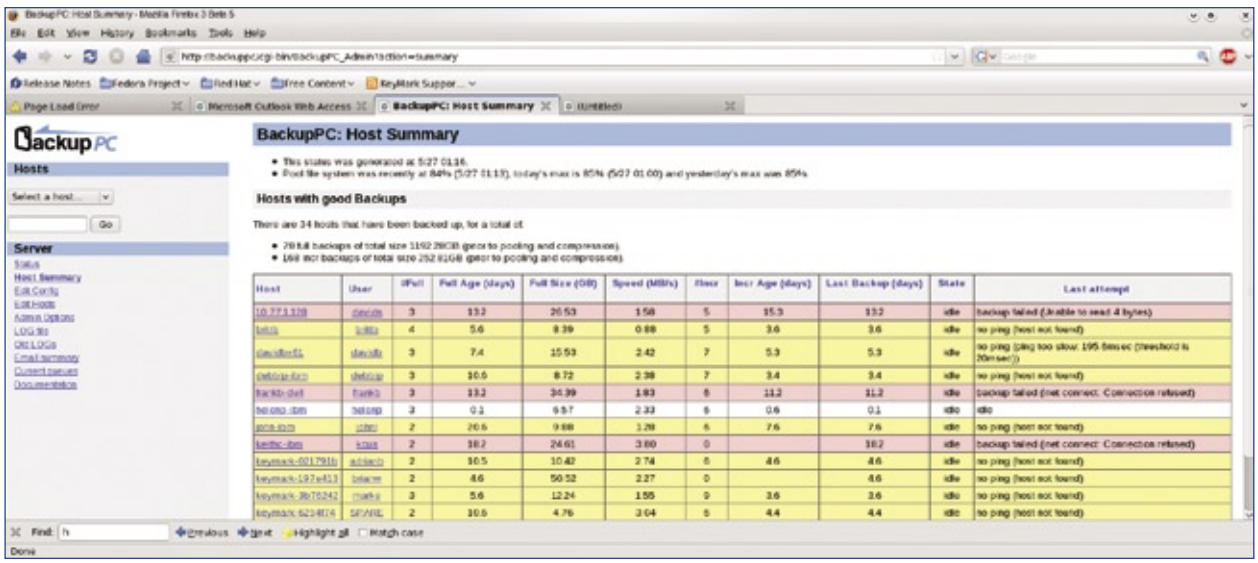


Figura 7 A página colorida de resumo fornece uma panorama bastante detalhado dos becapes.

© Linux New Media do Brasil Editora Ltda.

O projeto LessWatts ajuda o Linux a economizar energia

Economizando velas

Os sistemas Linux já consomem menos energia hoje do que há um ano. No entanto, o projeto LessWatts mostra que ainda há um longo caminho à frente.

por Sulamita Garcia

Economia de energia tem se tornado um novo foco na área de tecnologia. Talvez para você o consumo mensal de energia do seu computador não faça tanta diferença, mas para um data center com centenas ou milhares de computadores, além do custo direto da energia gasta pelos computadores, existe a energia gasta para resfriá-los, dissipando o calor que eles produzem. Ou, para habitantes de áreas distantes, significa poder usar seu computador por mais tempo, pois eles apenas têm energia para carregar a bateria três horas por dia.

O projeto *LessWatts* foi criado para reunir e fomentar iniciativas para melhoria do gerenciamento de energia no Linux. Todos os que utilizam notebooks podem confirmar que ainda existe um longo caminho a ser traçado nesse sentido. O website do projeto contém ferramentas, documentação, dicas e guias de desenvolvimento de software utilizando técnicas que evitam o gasto desnecessário de energia. Lançado há cerca de um ano, no início, o projeto consistia da ferramenta *PowerTop* e de alguns documentos. É surpreendente o crescimento do projeto, que hoje conta com 12 subprojetos e extensa documentação técnica abordando diferentes elementos envolvidos no consumo de energia no Linux. Porém, logo ficou claro que, embora o consumo de energia do kernel tenha melhorado muito, as aplicações em torno do Linux tinham muito o que melhorar para que o sistema fosse eficiente nesse quesito.

Assim, vários projetos receberam e continuam recebendo melhorias, além do trabalho de documentar e criar nos desenvolvedores uma nova mentalidade para desenvolvimento de software.

Vamos examinar alguns desses projetos, bem como algumas dicas para se ter em mente na hora de desenvolver software.

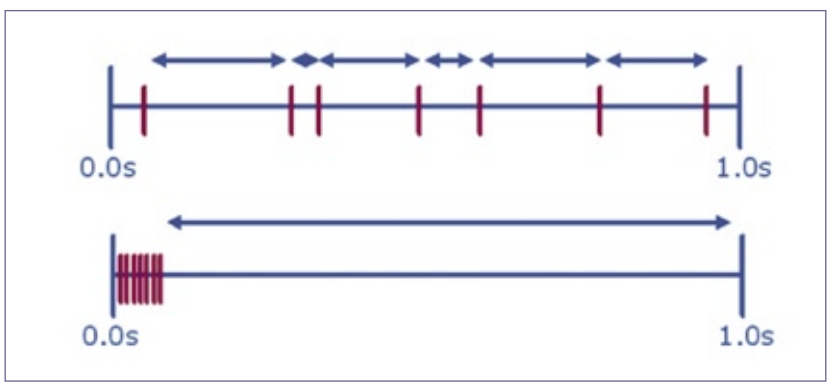


Figura 1 Ao agruparmos timers, conseguimos fazer com que o processador seja acionado durante um tempo menor e permaneça por mais tempo no estado ocioso.

Otimização economiza energia

Recentemente, a Xandros anunciou a adoção da plataforma Moblin para seus netbooks e, com isto, aumentou em cerca de 25% a duração da bateria dos dispositivos. Isso representa uma hora a mais, apenas com modificações de software disponíveis como código aberto. Vejamos algumas formas de conseguir isso:

- Evitar *polling* – a maioria das aplicações em uso hoje em dia foi implementada com base em um kernel com ciclos periódicos. Essas aplicações supõem que o kernel acordará cada CPU múltiplas vezes por segundo. Como resultado, o polling se tornou uma solução fácil e simples para muitas aplicações. Cada vez que uma aplicação

© Linux New Media do Brasil Editora Ltda.

Quadro 1: Abuso do polling

Certas aplicações realizam polling intensamente durante sua operação. Algumas dessas ações e os responsáveis por elas:

- checar uma vez por segundo se o mouse se moveu (*gnome-screensaver*)
- checar dez vezes por segundo se o volume do som foi modificado *(mixer applet*)
- checar uma vez por segundo se é hora de mostrar o próximo minuto no relógio (*clock applet*)
- checar 30 vezes por segundo (*gamin*) ou dez vezes por segundo (*Evolution*) se existe algum dado na fila interna
- checar dez vezes por segundo se existe um leitor de smartcard inserido na USB (*gdm-daemon*)
- checar mil vezes por segundo se existem dados em um pipe (*gksu*)

Polling é um recurso válido para várias aplicações, mas tenha em mente que seu uso indiscriminado pode gerar consumo desnecessário de energia.

faz uma requisição à CPU por meio do kernel, a CPU acorda do estado de inatividade, consumindo energia. Um processador atual consome 34 watts por segundo quando funcionando em carga total, e apenas 1 quando ocioso. O **quadro 1** lista exemplos de aplicações reais que abusam do polling.

- Adiantar a entrada no modo ocioso utilizando a velocidade máxima do processador: processadores tendem a economizar mais energia quando ociosos; então, geralmente é melhor executar uma operação o mais rápido possível para poder permanecer inativo por mais tempo. Vejamos um exemplo: no processador mencionado antes, decodificar um segundo de um arquivo MP3 leva meio segundo usando metade da velocidade máxima e 0,25 segundo a toda velocidade. Então, o consumo de energia para essa tarefa é:

Metade da velocidade: 0.5s × 24W + 0,5 × 1W = 12,5 Joules

Velocidade Máxima: 0,25s × 34W + 0,75 × 1W = 9,25

- Desligar dispositivos que não estejam sendo usados: esses dispositivos, além de consumirem energia desnecessariamente, impedem que o sistema entre em modo de economia de energia.
- Agrupar *timers*: muitos programas usam timers, então agrupá-los ajuda a reduzir as chamadas que tiram o processador do modo ocioso, como mostra a **figura 1**.
- Atentar ao uso de linguagens de alto nível: essas linguagens são ferramentas convenientes para a obtenção de resultados rapidamente, e freqüentemente têm funcionalidades que permitem executar tarefas complexas com o mínimo de esforço. Entretanto, esteja ciente de que algumas dessas estruturas são difíceis de implementar e algumas vezes o ambiente de execução associado à linguagem empregam polling em alta freqüência. Ao usar linguagens tais como *Java*, *Visual C#*, *Python* e *Ruby*, cheque o resultado final e evite usar algumas das estruturas de *threading* mais primitivas. E quando existir a possibilidade de escolha do ambiente de execução, avalie diferentes alternativas e versões.

PowerTop

Uma das principais ferramentas do projeto LessWatts é o PowerTop, uma aplicação simples cujo objetivo é medir os vilões de consumo de energia no sistema. Além de mostrar quais aplicações estão consumindo mais recursos, ele mostra dicas para melhorar ainda mais a economia de energia. Se você quiser, basta acionar as teclas que são listadas junto com as dicas para que o PowerTop faça as mudanças de forma autônoma.

Assim que foi lançado, o PowerTop logo começou a mostrar os aplicativos que costumavam estar no topo da lista de esbanjadores de energia. O *Pidgin*, antigo *Gaim*, foi um dos primeiros a receberem melhorias e, na versão 2.0.1, já deixava o "Muro da Vergonha". O *Firefox* ainda consta na lista, mas também vem recebendo melhorias para deixar essa condição.

O projeto LessWatts é uma comunidade que serve de ponto de troca de conhecimento entre todos os interessados em melhorar o desempenho dos sistemas de código aberto no consumo de energia. Nele, você pode encontrar diversos documentos, tutoriais, ferramentas de avaliação do seu sistema e muito mais. Além disso, pode participar e contribuir diretamente para essas melhorias. Não deixe de visitá-lo. ■

Mais informações

[1] LessWatts: http://www.lesswatts.org/

© Linux New Media do Brasil Editora Ltda.

Balanceamento de carga no Apache

Apaches unidos

As exigências atuais de desempenho e disponibilidade tornam o balanceamento de carga indispensável. Veja como o Apache já está equipado para lidar com isso sozinho.

por Erik Abele

Existem várias tecnologias que suportam o balanceamento de carga de servidores web. Balanceadores de carga vêm em todas as formas e tamanhos, desde simples técnicas baseadas em DNS até sistemas proprietários vastos e versáteis. Entretanto, há casos em que os recursos de balanceamento de carga exigidos já estão disponíveis no próprio servidor web *Apache*. Este artigo descreve algumas estratégias para o balanceamento de carga no Apache.

O esquema da **figura 1** mostra a estrutura por dentro do sistema de balanceamento baseado em software. Nesse cenário, vários balanceadores de carga aceitam requisições de usuários e as distribuem para um conjunto de servidores independentes com base num esquema pré-definido. Múltiplos sistemas individuais podem rodar em paralelo para oferecer a resistência a falhas (mostrado no fundo da **figura 1**). O Apache inclui vários módulos para suporte ao balanceamento de carga (**tabela 1**) e é preciso assegurar que os módulos desejados estejam carregados:

```
LoadModule xyz_module modules/mod_
➥xyz.so
```

Como mostra a **tabela 1**, as capacidades básicas de balanceamento de carga do Apache incluem recursos como cache, compressão, reescrita de URLs e processamento de cabeçalhos. Alguns módulos da **tabela 1** são carregados por padrão, e a melhor forma de descobrir quais é consultar a configuração do servidor.

Se forem usados servidores de aplicação compatíveis com *JServ*, como *Tomcat* e *Jetty*, o gateway também pode usar o *Apache JServ Protocol* (ajp). Para isso, basta carregar o módulo `mod_proxy_ajp` em vez de `mod_proxy_http` e alterar as URLs de `http://` para `ajp://`.

O uso desse protocolo de formato binário oferece algumas vantagens com relação ao desempenho de conexão ao *back-end* e um menor overhead dos recursos, mas essa funcionalidade traz o preço de mais conexões permanentes com os back-ends.

Contudo, é possível rodar os back-ends ajp e http ao mesmo tempo como membros de um mesmo *pool*. Para mais detalhes, a documentação do servidor http Apache [1] é uma ótima fonte.

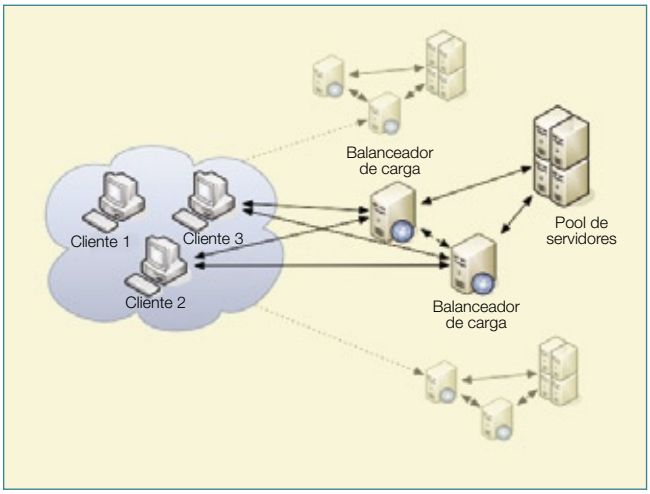


Figura 1 Visão esquemática de um sistema típico de balanceamento baseado em http.

Configuração

A configuração mostrada no **exemplo 1** inclui as definições do servidor front-end básico para balanceamento de carga no Apache.

Esse configuração começa com `Proxy-Requests` para desativar o modo normal de proxy e instalar o que se conhece como proxy reverso, ou *gateway*, para ser mais preciso. Desativar cabeçalhos `Via` (`ProxyVia`) torna o gateway invisível.

Os comandos `ProxyPreserveHost` e `ProxyErrorOverride` garantem que os cabeçalhos do host incluídos na requisição sejam repassados para os back-ends e que qualquer mensagem de erro gerada pelos back-ends seja substituída pelo balanceador de carga e, portanto, padronizada. A saída de um prazo adequado com `ProxyTimeout` finaliza a configuração básica.

A definição central, aquela de um conjunto de back-ends e seus membros, é processada por um contêiner `Proxy` e pela especificação de um esquema `balancer://`, seguida do nome do conjunto (*pool*). As instruções e parâmetros `BalancerMember` do contêiner especificam os membros

© Linux New Media do Brasil Editora Ltda.

individuais juntamente com suas propriedades.

No fim da configuração, o pool de back-ends definido anteriormente recebe um espaço de URLs separado; mais parâmetros definem a técnica genérica do balanceador de carga. Para ativar expressões regulares, é permitido usar o comando avançado `ProxyPassMatch` em vez de `ProxyPass`.

Como alternativa, o módulo de reescrita (`mod_rewrite`) e regras personalizadas permitem usar todo o poder das expressões regulares. Entretanto, nesse caso é preciso usar um `ProxySet`, pois os parâmetros do balanceador de carga não podem ser modificados pelas regras:

```
ProxySet balancer://pool1
➥ lbmethod=bytraffic
...
RewriteEngine On
RewriteRule ^/+(.*)$ balancer://
➥ pool1/$1 [P,L]
```

O **exemplo 1**, então, define dois servidores no back-end para o *pool1*. As requisições são distribuídas com base no número de requisições (veja o parâmetro `lbmethod`). O fator de carga atribui o dobro de requisições ao *servidor1* do que ao *servidor2*. As conexões são reutilizadas mas também restritas a um valor máximo. O espaço de URLs é definido como o espaço de URLs completo sob `/shop`.

A **tabela 2** mostra um resumo dos comandos `ProxyPass` e `BalancerMember` mais comuns. Para mais informações, veja a documentação do servidor http Apache [1].

Opções

O módulo de proxy do servidor Apache (`mod_proxy`) oferece uma gama inacreditável de configurações especiais. Há ferramentas para vários cenários diferentes. Por exemplo, pode-se usar o parâmetro `status` para operar um servidor com alta disponibilidade:

Tabela 1: Módulos necessários

Módulo	Função
`mod_proxy`	Módulo genérico de proxy
`mod_proxy_balancer`	Funções de balanceamento para o módulo de proxy
`mod_proxy_http`	Suporte a http para o módulo de proxy
`mod_cache`	Módulo genérico de cache
`mod_disk_cache`	Cache em arquivos para o módulo de cache
`mod_deflate`	Módulo para compressão de conteúdo
`mod_rewrite`	Módulo para processar URLs
`mod_headers`	Módulo para processar cabeçalhos http

```
BalancerMember http://
➥servidor4:1080... status=+H
```

Esse comando informa que o *servidor4* só será ativado em caso de falha de todos os membros restantes do pool.

Esse servidor é a última linha de defesa e pode ser usado para servir uma versão restrita de uma aplicação web ou para encaminhar requisições para um sistema substituto.

Sessões persistentes

Outra configuração típica é usada para suportar sessões do tipo *sticky* (persistentes):

```
BalancerMember http://
➥ servidor6:1080...
➥ stickysession=JSESSIONID
```

O parâmetro `stickysession`, combinado com o nome de um cookie suportado pelos back-ends, significa que requisições originadas por usuários individuais sempre serão enviadas para o mesmo servidor do back-end.

Esse tipo de distribuição limitada garante a persistência das requisições, mas interfere com a tarefa real do balanceamento de carga.

Para repassar informações para os servidores do back-end de uma forma

Exemplo 1: Configuração de exemplo

```
01 ProxyRequests Off
02 ProxyVia Off
03
04 ProxyPreserveHost On
05 ProxyErrorOverride On
06
07 ProxyTimeout 30
08
09 <Proxy balancer://pool>
10   BalancerMember http://servidor1:8080 min
➥=10 max=50 loadfactor=2
11
12   BalancerMember http://servidor2:8080 min
➥=5 max=25 loadfactor=1
13 </Proxy>
14
15 ProxyPass /shop balancer://pool1 lbmethod=byrequests
nofailover=Off maxattempts=3 stickysession=PHPSESSIONID
```


© Linux New Media do Brasil Editora Ltda.

Snow666

Tabela 2: Parâmetros comuns de balanceamento

Nome	Explicação
status	Status do membro do balanceador
loadfactor	Peso normalizado do membro do balanceador
lbset	Parte do cluster atribuída ao membro do balanceador
lbmethod	Método de distribuição de requisições usado pelo balanceador com base no número de requisições (byrequests) ou no volume de tráfego (bytraffic)
min	Número mínimo de conexões permanentes com o back-end
max	Número máximo de conexões permanentes com o back-end
maxattempts	Número máximo de tentativas antes de negar uma requisição
stickysession	Nome de um cookie persistente usado pelo servidor do back-end

Exemplo 2: Configuração de exemplo do mod_cache

```
01 CacheEnable disk /
02 CacheDisable /users
03 CacheRoot /var/cache/httpd
04 ...
05 AddOutputFilterByType DEFLATE text/html
```

direcionada, ou para influenciar as comunicações com o back-end, o módulo de proxy também suporta cabeçalhos http e variáveis de ambiente personalizados, que podem ser usados para restringir o número de conexões com o back-end ou para anunciar o uso de SSL:

```
SetEnv proxy-nokeepalive 1
...
RequestHeader set Front-End-Https
➥ "On"
```

Existem inúmeras variáveis padrão e cabeçalhos, tais como proxy-nokeepalive, proxy-sendcl, X-Forwarded-For ou X-Forwarded-Server, que economizam digitação e facilitam o trabalho administrativo.

Há ainda outros módulos que suportam o cache e a filtragem do conteúdo gerado pelos back-ends. Além de aumentar o desempenho, o cache também reduz o volume geral de tráfego e geralmente diminui o trabalho dos servidores do back-end. O **exemplo 2** ativa um cache simples baseado em arquivos, com compressão, para todo o espaço de URLs / (apenas para demonstrar o funcionamento do recurso de exclusão, o espaço de URLs sob /users foi excluído).

Administração

Se o módulo de status (mod_status) estiver carregado ao iniciar o servidor, o módulo de proxy também oferece uma interface web simples, porém prática. A configuração simples envolve atribuir um manipulador (*Handler*):

```
<Location "/.balancer-manager">
    SetHandler balancer-manager
</Location>
...
ProxyPass /.balancer-manager !
```

Todavia, é importante levar em consideração o controle de acesso e excluir o processamento de URLs individuais dentro do balanceador de carga, o que é feito com um comando ProxyPass negativo. Se for usado o módulo de reescrita do Apache (mod_rewrite), é possível definir uma regra separada para lidar com esse caso.

Restrições de gerenciamento

O gerenciamento é restrito à visualização do status de todos os balanceadores configurados, desativação de membros individuais do pool ou modificação de algumas configurações básicas. Porém, ele é extremamente útil caso seja encontrado um problema, ou ainda para monitorar múltiplos balanceadores de carga.

Conclusões

A versão 2.2 do Apache oferece uma forma descomplicada, eficiente, elegante e escalável para balanceamento de carga em ambiente http. Disponibilidade, uma curta curva de aprendizado e flexibilidade quase infinita favorecem esse excelente servidor. Com isso, o sistema de balanceamento de carga do Apache é uma alternativa muito razoável a outros populares programas comerciais e de código aberto. ■

Mais informações

[1] Documentação do Apache, httpd 2.2: http://httpd.apache.org/docs/2.2/pt-br/

[2] Apache Software Foundation: http://www.apache.org/

[3] HTTP 1.1: http://www.w3.org/Protocols/rfc2616/rfc2616.html

© Linux New Media do Brasil Editora Ltda.

Snow666
Gabarito

ATINGIR O MAIOR NÚMERO DE CLIENTES POTENCIAIS

DEPENDE DO
POTENCIAL
DA FERRAMENTA
QUE VOCÊ USA.

O SUCESSO DE UMA AÇÃO DE RELACIONAMENTO VIA E-MAIL, DEPENDE DE UM SERVIÇO REALMENTE DE QUALIDADE.

O UOL criou o serviço de E-mail Marketing que você precisa. Conheça todas as vantagens e diferenciais da melhor plataforma de marketing de relacionamento do mercado:

- Envio segmentado através de database marketing,
- Fila individual para o processamento das mensagens (você não concorre com disparos de outros clientes),
- Acompanhamento em tempo real das mensagens que estão sendo enviadas,
- Relatórios gráficos estatísticos.

POWERED BY: virtual target

E-mail Marketing
10 mil disparos
R$49,00 por mês

Assine: 0800 723 6000

UOL HOST
QUALIDADE EM SERVIÇOS WEB

www.uolhost.com.br

© Linux New Media do Brasil Editora Ltda.

Snow666

Rootkits virtuais e o futuro da segurança

Malware virtual

Uma nova geração de rootkits evita a detecção por meio da virtualização do sistema comprometido – e o usuário não percebe nada.

por Wilhelm Dolle e Christoph Wegener

Davorr – Fotolia

Num jogo típico de gato e rato de agressores e defensores, o objetivo do jogo é ganhar ou manter controle do sistema operacional (veja a **figura 1**). O malware legado tenta escalar privilégios e, se possível, rodar no chamado *anel 0*, o modo de kernel do sistema operacional. Assim que ele chega lá, o exploit, e portanto o agressor, consegue manipular o sistema.

A virtualização costuma ser propagandeada como um grande avanço para a segurança dos sistemas. Múltiplos sistemas virtuais podem rodar no mesmo hardware sem a possibilidade de influenciarem um ao outro. Esse isolamento evita várias técnicas de ataque padrão, mas as tecnologias atuais de virtualização também abrem uma nova frente para ataques que jamais foram possíveis no passado. Os especialistas já estão falando sobre uma nova geração de rootkits que vai explorar a característica da virtualização para evitar sua detecção.

Rootkits permite que um agressor sustente acesso privilegiado em segredo a um computador. Um rootkit pode ocultar processos, conexões de rede, arquivos e diretórios para controlar remotamente o PC da vítima, instalar backdoors, capturar pacotes de rede ou registrar as teclas pressionadas. Após o rootkit conseguir rodar no modo do kernel, ele pode filtrar e manipular valores de retorno de chamadas do sistema e esconder arquivos, diretórios e processos com grande eficácia.

Um rootkit com acesso ao modo do kernel consegue facilmente terminar aplicativos executados no modo do usuário (anel 3) por qualquer usuário normal, incluindo o root. Uma vez conquistado o kernel, o rootkit é extremamente difícil de ser identificado e removido. Obviamente, o dono legítimo do computador também pode usar o modo do kernel para criar uma linha de defesa eficaz.

A virtualização essencialmente age como outro anel com privilégios ainda mais altos que o anel 0. Qualquer um que comprometa o ambiente de virtualização praticamente controla todo o ambiente físico no qual o sistema roda. Malwares escondidos nessa camada são ainda mais difíceis de descobrir e remover do que os do modo do kernel.

Pesquisadores da Universidade de Michigan e da Microsoft Research demonstraram um rootkit de prova de conceito inicial chamado *SubVirt*[1] em março de 2006, dando à luz assim a primeira geração de rootkits que exploram a virtualização. Depois de infectar um computador, o rootkit se instala sob o sistema já existente e roda numa máquina virtual após a reinicialização.

Para permitir que isso aconteça, o SubVirt também modifica a seqüência de inicialização para que a BIOS não carregue mais diretamente o registro mestre de inicialização (MBR) do sistema operacional, mas inicie uma máquina virtual em seu lugar. A máquina virtual então executa a BIOS e inicia

© Linux New Media do Brasil Editora Ltda.

o sistema operacional copiado para o ambiente virtual pela MBR.

Enquanto os usuários continuam trabalhando em seus sistemas operacionais (virtuais), desapercebidos do que se passa uma camada abaixo, o SubVirt inicia uma segunda instância e realiza todos os tipos de truques malévolos. O rootkit não pode ser terminado ou desinstalado pelo sistema hóspede, pois rootkit controla a máquina virtual sobre a qual o sistema hóspede é executado. Pesquisadores de segurança se referem a essa técnica como rootkit baseado em máquina virtual (VMBR, na sigla em inglês).

A **figura 2** mostra a nova situação; as áreas cinzentas são ocupadas pelo rootkit. A capacidade do agressor de controlar o sistema da vítima também aumenta, pois o rootkit agora consegue usar o monitor de máquina virtual (VMM) para manipular, encaminhar ou bloquear dados arbitrários e características de hardware em rota para o sistema operacional hóspede, sem deixar o menor resquício de evidência que possa ser detectado por métodos legados.

Os pesquisadores demonstraram sua capacidade de comprometer tanto máquinas Windows XP quanto Linux, implementando ataques de prova de conceito com quatro vetores diferentes, incluindo um servidor web de phishing, um keylogger e um spyware que varre o sistema infectado atrás de dados confidenciais.

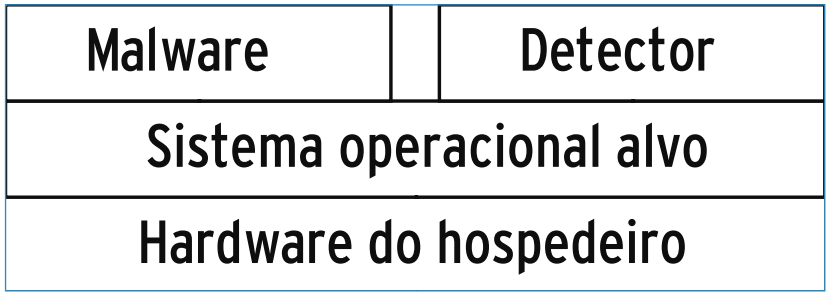


Figura 1 Softwares de detecção só conseguem identificar malwares que estejam rodando num nível igual ou mais alto que eles próprios, como o malware e o detector nesta figura.

A tecnologia usada pela versão Windows do SubVirt é baseada no software *Microsoft Virtual PC*, e a versão para Linux se baseia no *VMware*. Entretanto, são necessários privilégios administrativos para o sistema instalar o rootkit, embora um agressor possa usar várias técnicas para alcançar esse objetivo.

Descoberta

Tecnologias de virtualização como o VMware e o *Xen* estão tão difundidas que descobrir que um sistema operacional está rodando num ambiente virtual não necessariamente significa que foi encontrado um rootkit. A maioria das ferramentas de diagnóstico demonstra a existência do ambiente virtual com base em anomalias. Elas medem tempos de resposta, com a suposição de que o mesmo comando deveria levar mais tempo para completar num ambiente virtual do que nativamente – pressupondo hardwares idênticos e instalação idêntica. O efeito é causado pela máquina virtual consumindo, ela própria, ciclos de CPU.

Esse tipo de acompanhamento de tempo automatizado poderia ser suficiente para detectar uma máquina virtual legítima; porém, não elimina a existência de um rootkit, pois o rootkit também controlaria o clock interno. Além disso, a idéia de usar hardwares externos para medir tempos de resposta manualmente não escala muito bem.

O que realmente denuncia um sistema infectado são as anomalias na configuração visível do hardware, que são típicas de ambientes virtuais e particularmente válidas para placas gráficas e de rede. Ao comparar a configuração física conhecida à saída de comandos como `system-info`, `hwinfo` ou o sistema de arquivos `/proc/`, é possível descobrir as diferenças (**figura 3**). Administradores Windows precisarão usar o gerenciador de dispositivos ou ferramentas de terceiros.

O espaço livre em disco, ou a memória livre, também podem indicar a

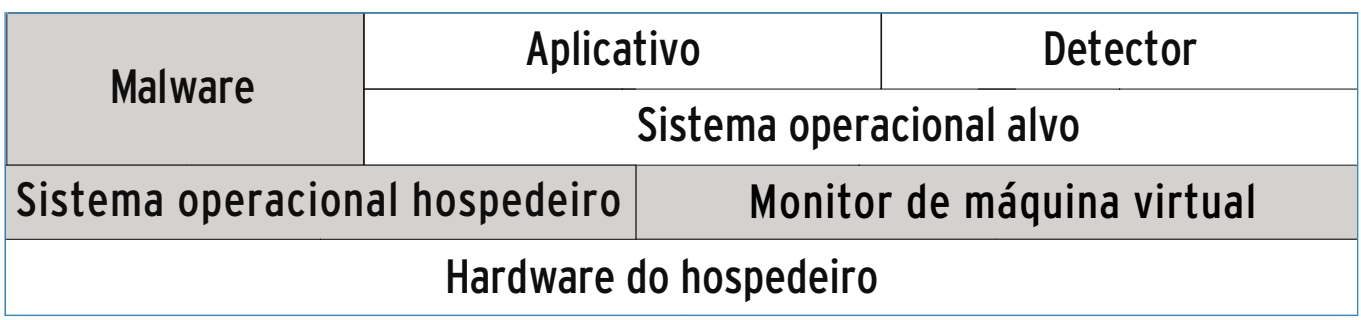


Figura 2 Um rootkit que ataca a camada de virtualização possui privilégios amplos. O sistema operacional hóspede não consegue terminar ou desinstalar o software.

© Linux New Media do Brasil Editora Ltda.

existência de um ambiente virtual. Por exemplo, se você não conseguir usar o espaço total do disco rígido, pode ser que o sistema hospedeiro da máquina virtual esteja precisando desse espaço. Porém, tenha cuidado: o sistema hospedeiro também poderia manipular esse dado, já que ele consegue controlar qualquer tipo de saída importante para o sistema hóspede.

Inicialização externa

O rootkit descoberto primeiro, o SubVirt, reside persistentemente no disco rígido; contudo, essas mudanças são muito difíceis de identificar no sistema em execução. Para identificar confiavelmente uma infecção, talvez seja necessário desligar a máquina, iniciá-la por uma mídia diferente e analisar o disco rígido – obviamente, esse método é problemático em vários servidores.

As ferramentas desenvolvidas especificamente para esse fim dão aos administradores outra abordagem para detectar a existência de um sistema operacional virtualizado. Por exemplo, Joanna Rutkowska liberou o *Red Pill*[2] no final de 2004. Ele funciona porque as instruções SIDT, SGDT e SLDT executadas por sistemas virtuais retornam valores diferentes daqueles retornados por uma CPU nativa. Como exemplo, temos a instrução SIDT retorna o endereço da tabela de interrupções.

Como alternativa, o *Scoopy doo*[3] e o *Jerry*[4] de Tobias Klein detectam um ambiente VMware. Se você tiver certeza de que está rodando um sistema sem softwares de virtualização, achados positivos dessas ferramentas são uma indicação verdadeira de um VMBR ativo.

Suporte da CPU

Essa nova geração de rootkits virtualizados pode ser perigosa, mas, como se pode esperar, essa técnica também possui alguns pontos fracos. Por exemplo, o rootkit precisa de uma reinicialização para se tornar ativo, e a reinicialização é fácil de detectar. Programadores de rootkits já criaram várias outras técnicas – algumas baseadas no desenvolvimento mais recente de virtualização baseada em hardware.

Ou o sistema inteiro fica virtualizado – caso das partições lógicas da IBM (*LPAR*[5]), por exemplo –, ou a virtualização fica restrita a componentes individuais, tais como o processador (via Intel VT[6] ou AMD-V[7]).

A virtualização de sistema ou sistema operacional depende de um VMM aceitar, dos sistemas hóspedes, instruções destinadas ao hardware. Sem suporte do processador, o VMM precisa capturar e modificar certas instruções do anel o vindas do sistema hóspede, por exemplo, para proteger seu próprio gerenciamento de memória de acessos das máquinas virtuais.

Em contraste com isso, a virtualização de processadores AMD e Intel permite que o VMM envie instruções desse tipo diretamente para o processador. A própria CPU se encarrega de manter separados os processos dos sistemas hóspedes e do VMM, pois sua lógica é inacessível até para processos do anel o. A capacidade de funcionar sem etapas de modificação da VMM ajuda o sistema a alcançar um melhor desempenho.

Alguns pesquisadores começaram a usar a virtualização baseada em hardware como um modelo para uma nova geração de rootkits os quais se beneficiam da tecnologia de processadores, que permite que eles insiram um hypervisor extra entre o hardware visível e o software. O hypervisor obtém o controle do sistema e converte imediatamente o sistema operacional num hóspede virtual, de forma dinâmica. Diferentemente da virtualização baseada em software, esse tipo de seqüestro não precisa de reinicialização, o que dificulta ainda mais a detecção da invasão.

Alguns rootkits utilizam esse tipo de aninhamento, como o *Blue Pill*[8] de Joanna Rutkowska, liberado em

```
jkleinert@laforge:~$ hwinfo --netcard
18: PCI 0a.0: 0200 Ethernet controller
  [Created at pci.281]
  UDI: /org/freedesktop/Hal/devices/pci_8086_1229
  Unique ID: rBUF.HVgIlgOrmpC
  SysFS ID: /devices/pci0000:00/0000:00:0a.0
  SysFS BusID: 0000:00:0a.0
  Hardware Class: network
  Model: "Intel EtherExpress PRO/100+ Management Adapter"
  Vendor: pci 0x8086 "Intel Corporation"
  Device: pci 0x1229 "82557/8/9 [Ethernet Pro 100]"
  SubVendor: pci 0x8086 "Intel Corporation"
  SubDevice: pci 0x000c "EtherExpress PRO/100+ Management Adapter"
  Revision: 0x08
  Driver: "e100"
  Driver Modules: "e100"
  Device File: eth0
  Memory Range: 0xdffff000-0xffffffff (rw,non-prefetchable)
  I/O Ports: 0xc800-0xc83f (rw)
  Memory Range: 0xdfe00000-0xdfefffff (rw,non-prefetchable)
  Memory Range: 0xdfd00000-0xdfdfffff (ro,prefetchable,disabled)
  IRQ: 10 (94866 events)
  HW Address: 00:02:b3:17:13:b3
  Module Alias: "pci:v00008086d00001229sv00008086sd0000000Cbc02sc00i00"
  Driver Info #0:
    Driver Status: e100 is active
    Driver Activation Cmd: "modprobe e100"
  Driver Info #1:
    Driver Status: eepro100 is not active
    Driver Activation Cmd: "modprobe eepro100"
  Config Status: cfg=new, avail=yes, need=no, active=unknown
jkleinert@laforge:~$
```

Figura 3 Ferramentas como o *hwinfo* ajudam a encontrar diferenças entre hardware físico e o hardware identificado pelo sistema operacional.

© Linux New Media do Brasil Editora Ltda.

2006 para AMD-V, ou o *Vitriol*[9], apropriado para Intel VT graças a Dino Dai Zobi. Em 2007, Rutkowska e Alexander Tereshkin relançaram o Blue Pill, reescrevendo completamente a detecção e acrescentando vários recursos [10]. Uma vez que o novo Blue Pill esteja rodando com privilégios de administrador, ele ativa o modo SVM (*Secure Virtual Machine*) nos processadores AMD mais recentes e instala o VMCB (*Virtual Machine Control Block*), que se encarrega de controlar o sistema operacional infectado em modo de hóspede.

Até a próxima reinicialização, o próprio rootkit funcionará num nível inferior à camada do hypervisor. Diferentemente do SubVirt, o Blue Pill não reside permanentemente no disco e, portanto, não sobrevive à reinicialização. Por outro lado, ele não deixa rastros que poderiam ser descobertos offline ao longo da investigação forense.

A Internet está cheia de discussões volumosas e controversas sobre a facilidade de se identificar um rootkit de segunda geração. Evidentemente, medições com comandos durante a execução como as descritas acima são menos confiáveis, pois a sobrecarga é menor (ou inexistente) graças ao sistema hospedeiro. Como Joanna Rutkowska anunciou seu Blue Pill originalmente como um "malware indetectável", as pessoas se apressaram para provar que ela estava enganada. Muitas sugestões para detectar o rootkit por meio de análises de tempo foram feitas. As técnicas conseguiram, no máximo, apenas confirmar que o sistema operacional estava rodando num ambiente virtual, o que não necessariamente significa uma infecção por rootkit.

Para evitar a descoberta, Rutkowska e Tereshkin também desenvolveram um programa chamado *Blue Chicken*[11], que detecta análises de tempo e temporariamente remove o malware da memória virtual, ocultando assim qualquer anomalia de tempo. A corrida entre a tartaruga e a lebre – ou seja, o jogo de esconder e descobrir rootkits – está a pleno vapor.

Prevenção

Em face da dificuldade para se identificar um rootkit após ele ser instalado, a necessidade de assegurar e monitorar o processo de inicialização e o VMM se torna crítica. A melhor técnica para o administrador é evitar uma infecção.

De um ponto de vista técnico, os modelos sugeridos pelo Trusted Computing Group (TCG [12]) parecem um bom início. O componente-chave é um Trusted Platform Module (TPM), um chip de hardware que implementa o modelo TCG na placa-mãe do computador. O chip TPM oferece funções e operações criptográficas utilizáveis por meio da BIOS e do sistema operacional para conferir a confiança de um sistema.

Por cima disso, o chip TPM pode ser usado para fazer testes de integridade durante o processo de inicialização. Para permitir que isso aconteça, rotinas especiais para componentes críticos do sistema calculam *hashes* criptográficos e os armazenam nos registradores de configuração de plataforma (PCRs) do chip TPM. A instância de avaliação correspondente compara os valores de hash calculados com os valores de referência guardados e declara a configuração atual do sistema como válida (*valid*), permitida (*permitted*) ou confiável (*trustworthy*).

Valores de hash incorretos indicam alterações não autorizadas ao sistema e desencadeiam respostas da instância avaliadora, como o cancelamento do processo de inicialização ou um *kernel panic*. Se a avaliação e a possível resposta ocorrerem enquanto o sistema estiver iniciando, refere-se a isso como uma inicialização segura (*secure boot*). Se isso ocorrer depois – geralmente tratado pelo sistema operacional –, é chamado como inicialização confiável (*trusted boot*).

Essas técnicas permitem que o administrador verifique a integridade de todo o sistema desde os processos de inicialização até o VMM. O chip TPM e partes da BIOS agem como uma raiz de confiança para medições. Quando o sistema inicia, o chip TPM ajuda a BIOS a verificar suas próprias partes e guarda os valores de hash nos registradores de configuração da plataforma TPM. Depois disso, a BIOS investiga a MBR do dispositivo de inicialização

Quadro 1: Neoware

Os nomes dos dois rootkits mencionados neste artigo, *Red Pill* e *Blue Pill*, foram retirados do filme "Matrix" (1999). Em uma cena, Morpheus (Laurence Fishburne) dá ao hacker Neo (Keanu Reeves) a escolha de ingerir a pílula vermelha ou a azul. A cena se passa num arranha-céu.

Morpheus: Infelizmente, não se pode contar a ninguém o que é a Matrix. Você tem que ver com seus próprios olhos. [Aproxima-se de Neo] Esta é sua última chance. Depois disso, não há volta. [Em sua mão esquerda, Morpheus mostra uma pílula azul] Escolha a pílula azul e a história termina. Você vai acordar em sua cama e acreditar no que preferir. [Uma pílula vermelha está em sua outra mão.] Escolha a pílula vermelha e continue no Mundo da Imaginação que eu vou mostrar a profundidade disso. [Pausa longa; Neo começa a escolher a pílula vermelha] Lembre-se – estou oferecendo somente a verdade, nada mais. [Neo pega a pílula vermelha e a engole com um copo d'água.] (Fonte:[13])

© Linux New Media do Brasil Editora Ltda.

Snow666

e repassa o controle para o carregador de boot.

Se o carregador de boot tiver sido instruído para tal, ele continuará a partir daí medindo valores. Alvos adequados são o arquivo de configuração do carregador de boot, o disco RAM inicial, o arquivo do kernel, o arquivo do VMM e assim por diante. Se isso for feito consistentemente, o resultado é uma corrente de confiança desde a BIOS até o VMM. O VMM é o mestre de todos os sistemas hóspedes e consegue verificar suas integridades e responder de forma apropriada, dependendo do objetivo, sem os sistemas hóspedes influenciarem o processo. Isso tudo parece ótimo na teoria, mas cuidar dos detalhes pode gastar muito tempo.

Para alcançar uma inicialização demonstravelmente segura, os valores de referência de cada elemento da cadeia de confiança precisam residir em memória confiável. O único lugar para armazenar os valores de checksum e hash no início da corrente de confiança é na NVRAM do próprio chip TPM. Depois, a BIOS precisa guardar os valores de referência da MBR em sua própria NVRAM e o carregador de boot necessita dos valores do arquivo de configuração do carregador de boot, e assim por diante – a instância verificadora garante a confiança do próximo elo da corrente, incluindo os valores de referência guardados nesse elo.

Melhores referências

A inicialização segura é um tanto restritiva quanto ao gerenciamento de valores de referência. Isso também significa que o administrador deve substituir os valores de referência guardados no chip TPM depois de atualizar a BIOS com o novo firmware, e se o código do carregador de boot mudar, o valor de referência da BIOS também precisará ser alterado, e assim por diante. Todas essas transações precisam ser seguras, o que só é possível com instâncias confiáveis e autorizadas.

Se uma ação falhar, o administrador também precisará de algum tipo de mecanismo de becape e recuperação para iniciar o sistema. E a questão dos usuários serem ou não capazes de influenciar isso é outro problema.

Toda a infra-estrutura tem nível bem baixo; por exemplo, a tecnologia VT ad Intel segue o modelo com suas implementações de "Secure Extensions" (extensões seguras) e "Secure Boot" (inicialização segura). Em razão das complexidades técnicas, o uso desses recursos em PCs de produção dificilmente será útil no futuro próximo. Sistemas embarcados, telefones móveis e PDAs, nos quais o gerenciamento de valores de referência é mais fácil de tratar, parecem candidatos mais prováveis.

Uma questão ainda permanece com relação a quem deve ser a origem da confiança: o proprietário do sistema, o fabricante ou talvez terceiros? O proprietário sempre tem o risco de perder o controle do sistema em decorrência, por exemplo, de um rootkit que comprometa todas as medições, ou de um fabricante que queira ditar os softwares que o usuário pode instalar no sistema.

A virtualização vem dando grandes passos na operação diária de servidores. O ambiente virtual oferece vários benefícios de segurança com o isolamento e com contextos paralelos, mas a virtualização também introduz novos vetores para malwares.

Conceitos iniciais demonstram de forma impressionante como rootkits podem explorar a virtualização para ocultar processos maliciosos. Podemos esperar mais implementações no futuro. Até agora, nenhum desses rootkits ultra-modernos foi lançado ao mundo, mas é bom manter-se atento à segurança do mundo virtual. ■

Mais informações

[1] Linux Virtual Server: Samuel T. King, Peter M. Chen, Yi-Min Wang, et al., "SubVirt: Implementação de Malware com Máquinas Virtuais" (em inglês): http://preview.tinyurl.com/kkgfy

[2] Joanna Rutkowska, "Red Pill... ou como detectar um VMM usando (quase) uma instrução de CPU": http://invisiblethings.org/papers/redpill.html

[3] Scoopy doo: http://preview.tinyurl.com/2oggnd

[4] Jerry: http://preview.tinyurl.com/4rahrs

[5] IBM LPAR na Wikipédia (em inglês): http://en.wikipedia.org/wiki/LPAR

[6] Intel VT: http://preview.tinyurl.com/3dljyg

[7] AMD-V: http://preview.tinyurl.com/48w32w

[8] Blue Pill (em inglês): http://preview.tinyurl.com/4wzpv3

[9] Vitriol (em inglês): http://preview.tinyurl.com/uu9fc

[10] Refazendo o Blue Pill (em inglês): http://bluepillproject.org

[11] Joanna Rutkowska e Alexander Tereshkin, "IsGameOver() Anyone?" (em inglês): http://preview.tinyurl.com/4mkhzx

[12] TCG: http://www.trustedcomputinggroup.org/home

[13] The Matrix: http://www.whysanity.net/monos/matrix3.html

© Linux New Media do Brasil Editora Ltda.

Snow666

Você está preparado para a TI virtualizada?

Aprenda a projetar e implementar infraestruturas de virtualização com Xen. Conheça outras soluções de Código Aberto, leia workshops profissionais, e maximize o desempenho em TI de sua empresa.

mais informações: www.linuxnewmedia.com.br

Coleção Linux Technical Review

LINUX NEW MEDIA
The Pulse of Open Source

© Linux New Media do Brasil Editora Ltda.

De volta ao shell – mas com janelas

Papo de botequim 2.0 Parte II

Janelas de seleção de arquivo com o Zenity.

por Julio Cezar Neves

– E aí, vamos pedir os chopes?

– Cara, estou gostando tanto deste zenity que estou mais com sede de saber do que de chope.

– Chico, traz um sem colarinho e se vira para arrumar um com saber para o meu amigo.

– E você fez o exercício que te passei na primeira aula?

– Fiz, veja no **exemplo 1**. Vamos analisar o código deste exemplo.

Primeiro você criou uma função só para fazer as perguntas, muito bom! Depois você fez um loop para ler o nome e só sai dele quando a variável `$Nome` estiver preenchida (`[ "$Nome" ] && break`) ou aborta caso tenha sido clicado em *OK* na função (`Pergunta && exit 1`).

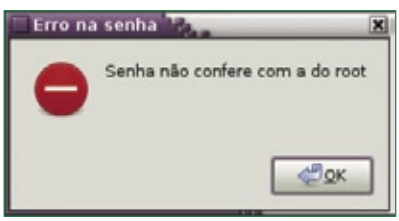


Figura 1 Janela de erro.

No primeiro bloco do loop de leitura da data de nascimento, caso tenha sido clicado em *CANCELAR* no calendário (`||`), você chama a função `Pergunta` e aborta se clicar em *OK* ou volta para o loop se clicar em *CANCELAR* (`Pergunta && exit 1 || continue`).

Ainda neste loop, você pegou a data de hoje no mesmo formato que usou para a data do calendário e testou se a diferença entre elas era maior que 10.000. Como ambas estão no formato AAAAMMDD, 10.000 significa que o último algarismo do ano atual é 1 a mais que o do nascimento. Gostei!

No loop de leitura do email, achei muito legal o uso do comando `test`, em que o `=~` significa estar usando expressões regulares (`[[ $Email =~ [[:alnum:].-]+@[[:alnum:].-] ]]`). O `[[...]]` representa o comando `test` propriamente dito, `=~` significa que você vai usar uma expressão regular e `[[:alnum:].-]` é uma lista formada por letras maiúsculas, minúsculas, números (classe *POSIX* `[:alnum:]`), ponto (`.`) e hífen (`-`). É importante notar que expressões regulares só funcionam no comando `test` a partir da versão 3.1.17 do bash.

No loop de leitura dos telefones, achei bacana a linha do `grep` (`grep -q ' ' <<< ${Tel[i]} || continue`) por duas razões: o uso de *here strings* (`<<<`) que substitui com vantagens um `echo ${Tel[@]} | tr...` e o uso da opção `-q` (*quiet*) do `grep` que o torna mudo e

Tabela 1: Opção --error

Opção	Efeito
`--text=TEXTO`	Define o texto da caixa
`--no-wrap`	Não permite quebra automática do texto

Exemplo 2: Diálogo de erro

```
1 zenity --error                \
2     --title "Erro na senha" \
3     --text "Senha não confere
com a do root"
```


© Linux New Media do Brasil Editora Ltda.

Exemplo 1: Exercício da primeira aula

```
#!/bin/bash
# Programa didático em shell+zenity
# Cadastramento
function Pergunta
{
    zenity --question     \
        --title "$Titulo" \
        --text "$Texto"
    return $?
}
while true
do
    Nome=$(zenity --entry                    \
        --title "Informações para o catálogo" \
        --text "Informe o nome da pessoa a cadastrar")
    [ "$Nome" ] && break
    Titulo="Nome não informado"
    Texto="Deseja abandonar?
        - Clique em OK para sair do programa, ou
        - Clique em CANCELAR para voltar"
    Pergunta && exit 1
done
while true
do
    Nasc=$(zenity --calendar                          \
        --title "Cadastramento de aniversários"       \
        --text "Informe a data do aniversário de $Nome" \
        --date-format "%Y%m%d") || {
            Titulo="Data não informada"
            Texto="Deseja abandonar?
         - Clique em OK para sair do programa, ou
         - Clique em CANCELAR para voltar"
            Pergunta && exit 1 || continue
         }
    Hoje=$(date +%Y%m%d)
    if  [ $((Hoje-Nasc)) -lt 10000 ]
    then
        Titulo="Provável erro de informação"
        Texto="Esta pessoa tem menos de um ano de idade.
             - Clique OK para manter esta idade ou
             - Clique CANCELAR para informar nova data"
        Pergunta || continue
    fi
    break
done
while true
do
    Email=$(zenity --entry
        --title "Informações para o catálogo" \
        --text "Informe o e-mail de $Nome")
    [ "$Email" ] || break
    [[ $Email =~ [[:alnum:].-]+@[[:alnum:].-] ]] || {
        zenity --warning             \
            --title "Erro no e-mail" \
            --text "Caracteres estranhos no endereço de e-mail"
        continue
        }
    break
done
while true
do
    let i++
    Tel[i]=$(zenity --entry                   \
        --title "Informações para o catálogo" \
```


© Linux New Media do Brasil Editora Ltda.

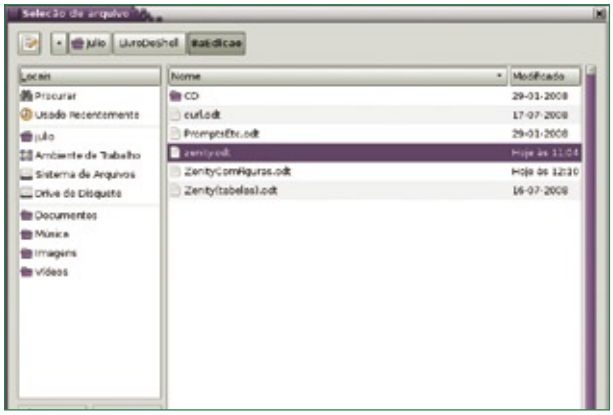


Figura 2 Janela de seleção de arquivo.

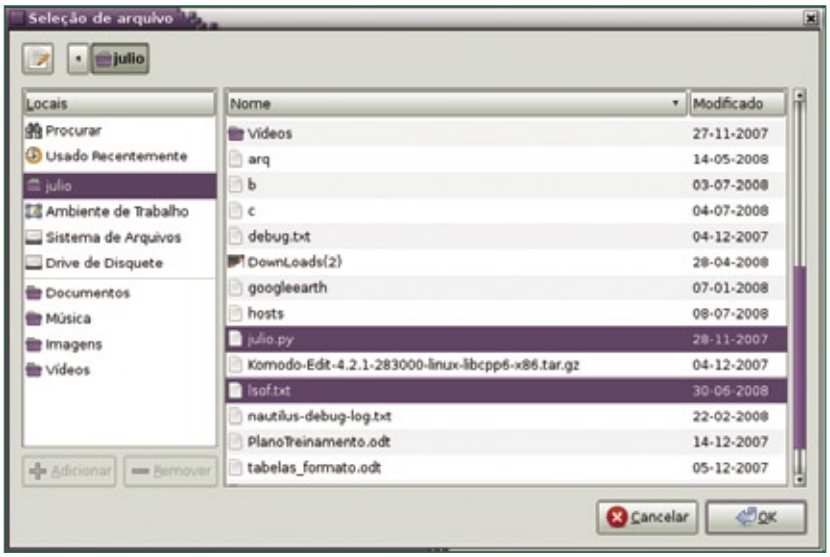


Figura 3 Janela para seleção de múltiplos arquivos.

Exemplo 3: Confirmar antes de sobrescrever.

```
1 Arquivo=$(zenity --file-selection \
2     --title "Selecão de arquivo"  \
3     --save                        \
4     --confirm-overwrite)
5 echo $Arquivo
6 /home/julio/UFs
```

Exemplo 4: Abertura de múltiplos arquivos.

```
01 Arquivos=$(zenity --file-selection \
02     --title "Seleção de arquivo"   \
03     --multiple                     \
04     --separator "^") || {
05          zenity --error            \
06              --text "Nenhum arquivo foi selecionado"
07          exit 1
08          }
09 echo $Arquivos
10 /home/julio/julio.py^/home/julio/lsof.txt
```

não manda a saída para a tela.

Estou vendo que você aprendeu tudo que te contei sobre Bash no Papo de Botequim 1.0 que foi publicado a partir da primeira **Linux Magazine**. Mas como você disse que tinha sede de saber quando chegamos aqui, vamos ver mais um pouco sobre o zenity.

A opção `--error` (**tabela 1**) exibe uma janela acusando um erro, como na **figura 1**. Para ilustrar isso, suponhamos que a senha do exemplo anterior tenha sido informada errada. Você poderá fazer como no **exemplo 2**.

Assim como na opção `--question`, caso a caixa de erro quebrasse a mensagem definida na opção `--text` e isso não fosse desejado, você poderia incluir a opção `--no-wrap`. Procedendo desta forma, a caixa ficaria mais larga para caber a mensagem inteira. Não se esqueça que o mesmo poderia ser feito aplicando a opção `--height`, que pode ser usada em qualquer caixa de diálogo.

Use essa opção para conseguir o caminho absoluto de um arquivo selecionado. É rápida e simples. Veja o trecho de programa no **exemplo 3**, que cria a janela mostrada na **figura 2**.

Podemos também selecionar múltiplos arquivos. Para isso é necessário usarmos a opção `--multiple` e é aconselhável usarmos também a opção `--separator`. Veja o **exemplo 4**, que produz a janela mostrada na **figura 3**.

Repare que os arquivos selecionados vieram separados por um circunflexo (^), que foi a opção feita pelo `--separator` (**exemplo 4**). Podemos também usar esta opção para pegar um arquivo que será sobrescrito. Usando `--confirm-overwrite` juntamente com `--save`. Caso o arquivo escolhido já exista, será automaticamente aberta uma caixa de perguntas (`--question`). Se você clicar no botão *OK*, ambas as caixas abertas serão fechadas e será devolvido o caminho completo do nome do arquivo selecionado. Se a opção feita for o botão *CANCEL*, a caixa de perguntas será fechada para que se faça uma nova escolha de arquivos.

A opção `--directory` inibe a escolha de arquivos comuns, permitindo acesso somente aos diretórios.

© Linux New Media do Brasil Editora Ltda.

Tabela 2: Opção --file-selection

Opção	Efeito
`--filename=NOME`	Define o nome do arquivo ou diretório inicial (default)
`--multiple`	Permite a seleção de diversos arquivos ou diretórios
`--directory`	Permite somente a seleção de diretórios
`--save`	Salva o arquivo selecionado
`--confirm-overwrite`	Usado com `--save`, pede confirmação caso o arquivo já exista
`--separator=SEPARADOR`	Usado com `--multiple`, especifica o separador quando retornar mais de um arquivo

Tabela 3: Opção --info

Opção	Efeito
`--text=TEXTO`	Define o texto da caixa
`--no-wrap`	Não permite quebra automática do texto

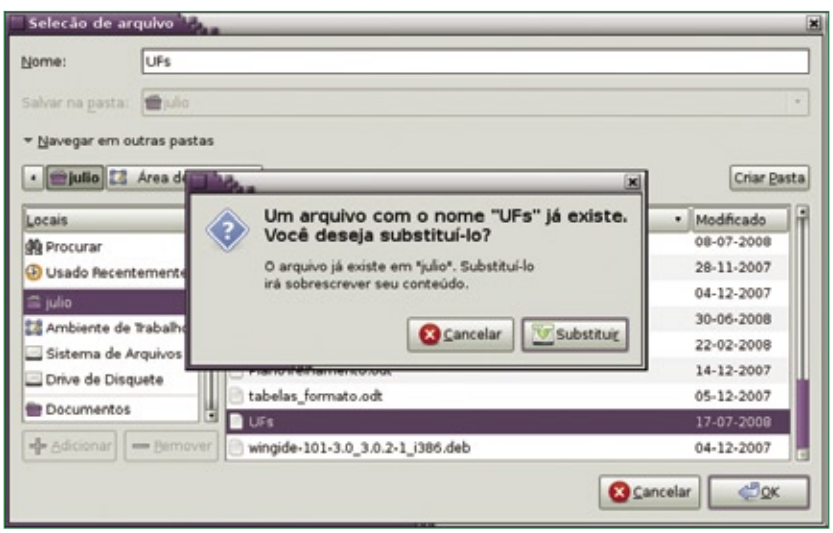


Figura 4 Janela para confirmar a sobrescrita de arquivo.

Exemplo 5: Exibir uma janela de informação.

```
1 zenity --info                \
2    --title="Fim de programa" \
3    --text="Tarefa concluída. Foram atualizados 5 arquivos."
```

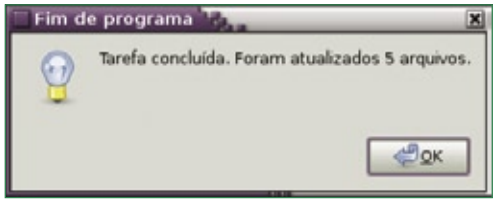


Figura 5 Janela de informação.

Temos ainda a opção `--filename="/caminho/do/arquivo"`, que posiciona a caixa de seleção de arquivos no diretório `/caminho/do` e oferece *arquivo* como padrão no campo Nome. Se `/caminho/do` fizer parte da variável `$PATH`, basta especificar apenas o nome deste.

A opção `--info`, detalhada na **tabela 3**, exibe uma janela de informação, como na **figura 5**.

Use a opção `--info` para informar algo, como no **exemplo 5**.

Caso a caixa de informações quebrasse a mensagem definida pela opção `--text` e isso não fosse desejado, você poderia incluir a opção `--no-wrap`. Procedendo desta forma, a caixa ficaria mais larga para caber a mensagem inteira, sem quebrá-la em mais de uma linha. Não se esqueça que o mesmo poderia ser feito aplicando a opção `--height`, que pode ser usada em qualquer caixa de diálogo.

– Chico, traz dois chopes que já estou de goela seca de tanto falar. Acho que se cuspir, sai em pó!

Não pense que você vai se livrar do programinha para fazer em casa. Desta vez vou passar um bem curtinho: faça um programa que permita ao operador escolher um diretório (somente diretórios) para onde serão copiados determinados arquivos. Em seguida deixe-o escolher quais arquivos do diretório corrente (todos de uma vez) serão copiados para aquele destino. Caso já exista arquivo com o mesmo nome, pedir confirmação. ■

Agradecimentos especiais ao SERPRO, empresa exemplo do uso de Software Livre no Governo Federal.

© Linux New Media do Brasil Editora Ltda.

Snow666

Programação paralela com o OpenMP

Em paralelo

O OpenMP traz o poder do multiprocessamento aos programas em C, C++ e Fortran.

por Wolfgang Dautermann

A nova safra de processadores com múltiplos núcleos está levando o mercado de desktops a um patamar de desempenho antes alcançável apenas por workstations e servidores. Da mesma forma, aqueles servidores quebra-galho que ficam na sala de servidores mas são alojados em gabinetes de PCs comuns também se beneficiam da multiplicação dos núcleos. No entanto, simplesmente ter um sistema multiprocessado não significa que todos os núcleos estejam sendo usados em todo o seu poder.

Na realidade, freqüentemente apenas um dos processadores está ocupado. A **figura 1** mostra a saída do *top* durante a execução do software *Xaos* de cálculos fractais. O programa parece usar 100% da CPU, mas a taxa real de uso é de apenas 60%.

Pressionar a tecla **[1]** lista as CPUs separadamente. Nesse modo (**figura 2**), é fácil conferir a carga de cada núcleo: uma CPU está 90% ocupada, enquanto a outra está descansando (carga de 0,3%).

O Linux recebeu suporte a sistemas multiprocessados há muitas luas, e há tempos as distribuições já instalam o kernel SMP por padrão. Portanto, o Linux já tem o que é preciso para usar o poder de múltiplos núcleos. Mas e o software?

Um programa em execução no sistema precisa estar ciente da arquitetura multiprocessada para desfrutar dos benefícios de desempenho. O *OpenMP* é uma especificação de API para "...paralelismo multi-thread com memória compartilhada" [1]. A especificação do OpenMP define um conjunto de diretivas do compilador, rotinas de bibliotecas e variáveis de ambiente para suportar ambientes multiprocessados.

Programadores C/C++ e *Fortran* podem usar o OpenMP com o intuito de criar novos programas apropriados para sistemas multiprocessados, e também de converter programas já existentes para rodarem com maior eficiência em ambientes multiprocessados.

```
top - 16:05:28 up 8 days,  1:58,  5 users,  load average: 1.01, 0.63, 0.27
Tasks: 122 total,   3 running, 119 sleeping,   0 stopped,   0 zombie
Cpu(s): 43.9%us, 12.6%sy,  0.0%ni, 41.4%id,  0.0%wa,  0.0%hi,  2.0%si,  0.0%st
Mem:   8174384k total,  4981140k used,  3193244k free,      744k buffers
Swap: 19543032k total,        0k used, 19543032k free,  3652792k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
26082 daute     25   0 94384  53m 4644 R  100  0.7   2:17.14 xaos.bin
31941 root      15   0  117m  43m 6916 S   13  0.5   0:44.09 Xorg
    1 root      18   0   804  304  244 S    0  0.0   0:02.61 init
    2 root      RT   0     0    0    0 S    0  0.0   0:00.00 migration/0
    3 root      34  19     0    0    0 S    0  0.0   0:00.00 ksoftirqd/0
    4 root      RT   0     0    0    0 S    0  0.0   0:00.00 migration/1
    5 root      34  19     0    0    0 S    0  0.0   0:00.00 ksoftirqd/1
    6 root      10  -5     0    0    0 S    0  0.0   0:00.18 events/0
    7 root      10  -5     0    0    0 S    0  0.0   0:00.17 events/1
    8 root      12  -5     0    0    0 S    0  0.0   0:00.00 khelper
    9 root      10  -5     0    0    0 S    0  0.0   0:00.00 kthread
   14 root      10  -5     0    0    0 S    0  0.0   0:00.00 kblockd/0
   15 root      10  -5     0    0    0 S    0  0.0   0:00.00 kblockd/1
   16 root      14  -5     0    0    0 S    0  0.0   0:00.00 kacpid
   17 root      14  -5     0    0    0 S    0  0.0   0:00.00 kacpi_notify
  116 root      11  -5     0    0    0 S    0  0.0   0:00.00 cqueue/0
  117 root      11  -5     0    0    0 S    0  0.0   0:00.00 cqueue/1
```

Figura 1 Por padrão, o *top* exibe a carga total das CPUs...

Multi-pista

A execução de programas de forma serial utiliza apenas um núcleo. A paralelização permite um uso mais eficiente do sistema multiprocessado.

A interface de programação do OpenMP, em constante desenvolvimento por vários fabricantes de hardware e compiladores desde 1997, oferece uma opção bem simples e portável para paralelizar programas escritos em C/C++ e Fortran.

O OpenMP é capaz de aumentar significativamente o desempenho dos programas, mas somente se a CPU realmente for exigida – e se a tarefa em questão for paralelizável, é claro. Esse costuma ser o caso em programas que dependem muito do processador.

Um, dois, vários

A API OpenMP oferece aos programadores uma opção simples para efetivamente paralelizar seus programas seriais pré-existentes por meio da especificação de algumas diretivas adicionais de compilação, que ficam semelhantes a esse código:

```
#pragma omp nome_da_diretiva
➥ [cláusulas]
```

Compiladores sem suporte ao OpenMP, como versões do GCC anteriores à 4.2, vão simplesmente ignorar as diretivas do compilador, o que significa que o código-fonte ainda pode ser compilado de forma serial:

```
$ gcc -Wall test.c
test.c: In function 'main':
test.c:12: warning: ignoring
➥ #pragma omp parallel
```


© Linux New Media do Brasil Editora Ltda.

Códigos específicos para OpenMP também podem ser compilados de forma condicional, com a diretiva `#ifdef`: a OpenMP define a macro `_OPENMP` para isso.

Um programa com OpenMP é iniciado normalmente como se fosse serial, com uma única thread. A primeira declaração OpenMP apresentada cria múltiplas threads:

```
... uma thread
#pragma omp parallel
{ ... várias threads }
... uma thread
```

A **figura 3** mostra como o programa é distribuído por múltiplas threads e depois reunido numa única.

Dividir e conquistar

Agora você criou múltiplas threads, mas elas ainda fazem a mesma tarefa. A idéia é que as threads processem dados diferentes umas das outras. Em C, há duas formas para resolver isso. Em Fortran há ainda uma terceira: "compartilhamento de trabalho em paralelo".

```
top - 16:08:00 up 8 days,  2:01,  5 users,  load average: 1.10, 0.82, 0.39
Tasks: 122 total,   3 running, 119 sleeping,   0 stopped,   0 zombie
Cpu0  :  90.0%us, 10.0%sy,  0.0%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu1  :   0.3%us, 10.3%sy,  0.0%ni, 89.4%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Mem:   8174364k total,  4987848k used,  3186536k free,      744k buffers
Swap: 19543032k total,        0k used, 19543032k free,  3653060k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
26082 daute     25   0 94384  53m 4644 R  100  0.7   4:46.56 xaos.bin
31941 root      15   0  117m  43m 6916 R   11  0.5   1:05.19 Xorg
    1 root      18   0   804  304  244 S    0  0.0   0:02.61 init
    2 root      RT   0     0    0    0 S    0  0.0   0:00.00 migration/0
    3 root      34  19     0    0    0 S    0  0.0   0:00.00 ksoftirqd/0
    4 root      RT   0     0    0    0 S    0  0.0   0:00.00 migration/1
    5 root      34  19     0    0    0 S    0  0.0   0:00.00 ksoftirqd/1
    6 root      10  -5     0    0    0 S    0  0.0   0:00.18 events/0
    7 root      10  -5     0    0    0 S    0  0.0   0:00.17 events/1
    8 root      12  -5     0    0    0 S    0  0.0   0:00.00 khelper
    9 root      10  -5     0    0    0 S    0  0.0   0:00.00 kthread
   14 root      10  -5     0    0    0 S    0  0.0   0:00.00 kblockd/0
   15 root      10  -5     0    0    0 S    0  0.0   0:00.00 kblockd/1
   16 root      14  -5     0    0    0 S    0  0.0   0:00.00 kacpid
   17 root      14  -5     0    0    0 S    0  0.0   0:00.00 kacpi_notify
  116 root      11  -5     0    0    0 S    0  0.0   0:00.00 cqueue/0
```

Figura 2 ...mas pode informar o valor de cada uma individualmente.

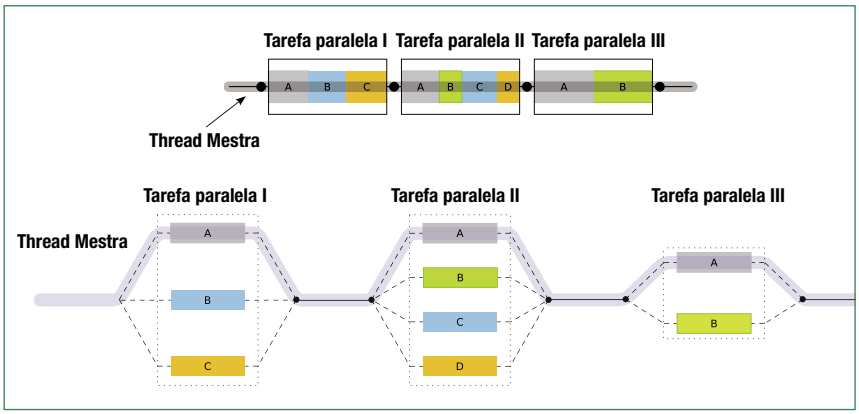


Figura 3 Método *Fork-Join* do OpenMP.

Exemplo 1: Seções e loops paralelos

```
Variação 1: Seções paralelas
.../* uma thread */
#pragma omp parallel /* várias threads */
{
#pragma omp sections
#pragma omp section
... /* Seção A do programa rodando em paralelo com B e C */
#pragma omp section
... /* Seção B do programa rodando em paralelo com A e C */
#pragma omp section
... /* Seção C do programa rodando em paralelo com A e B */
}
... /* uma thread */


Variação 2: Loops paralelos
... /* uma thread */
#pragma omp parallel [cláusulas ...]
#pragma omp for [cláusulas ...]
for (i=0;i<N;i++) {
    a[i]= i*i; /* paralelizado */
    }
... /* uma thread */
```

A primeira variação, as seções paralelas, roda seções do programa (blocos de código do programa não interdependentes) que suportam a execução simultânea em paralelo.

Para isso ocorrer, `#pragma omp parallel` define múltiplas threads. Isso significa que é possível rodar múltiplos blocos de programa independentes em threads individuais sem restrições quanto ao número de *seções* paralelas (**exemplo 1, variação 1: Seções Paralelas**). Além disso, pode-se combinar as duas diretivas de compilação, `parallel` e `sections`, para formar uma única diretiva, como em `#pragma omp parallel sections`.

A segunda variação, loops `for()` paralelos, paraleliza loops, o que é especialmente útil no caso de programas matemáticos computacionalmente intensivos (**exemplo 1, variação 2: Loops Paralelos**).

© Linux New Media do Brasil Editora Ltda.

Snow666

Tabela 1: Cláusulas

`shared(lista_de_variáveis)`	Existe apenas uma versão das variáveis, e todas as seções paralelas do programa a acessam. Todas as threads têm acesso de leitura e escrita. Se uma thread alterar uma variável, isso também afetará as outras threads. Padrão: todas as variáveis são `shared()`, exceto as de loops em `#pragma omp for`.
`private(lista_de_variáveis)`	Cada thread possui uma cópia privada não inicializada da variável. Padrão: somente variáveis de loops são privadas.
`default(shared|private|none)`	Define o comportamento padrão das variáveis: `none` significa que é preciso declarar explicitamente cada variáveis como `shared()` ou `private()`.
`firstprivate(lista_de_variáveis)`	Semelhante a `private()`; porém, neste caso, todas as cópias são inicializadas com o valor da variável antes da região ou loop paralelos.
`lastprivate(lista_de_variáveis)`	A variável recebe o valor da última thread que a alterou em processamento seqüencial após o loop ou região paralelos terem sido terminados.

A **figura 4** mostra como isso funciona. Novamente é possível combinar `#pragma omp parallel` e `#pragma omp for` para formar `#pragma omp parallel for`.

Escopo

Na programação com memória compartilhada, múltiplas CPUs podem acessar as mesmas variáveis. Isso torna o programa mais eficiente e economiza cópias. Em alguns casos, cada thread precisa de sua própria cópia das variáveis – como as variáveis do loop no caso do `for()` paralelo.

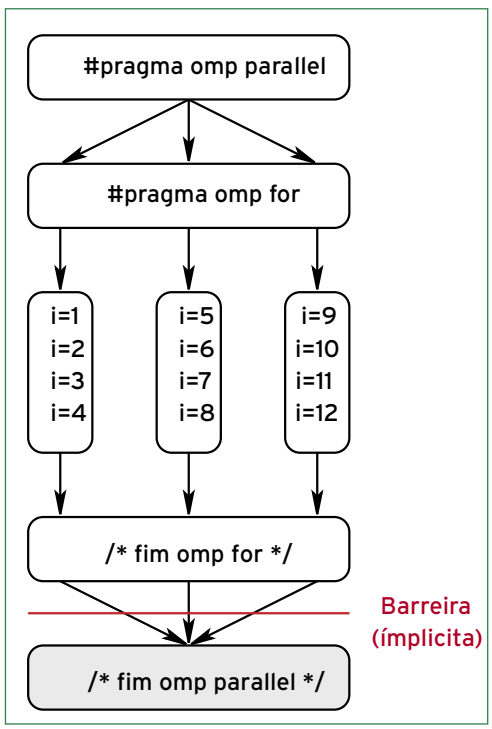


Figura 4 Loop `for()` paralelo.

As cláusulas especificadas nas diretivas OpenMP (veja as descrições na **tabela 1**) definem as propriedades dessas variáveis. Pode-se acrescentar cláusulas ao `#pragma`, por exemplo:

```
#pragma omp parallel
➥for shared(x, y) private(z)
```

Erros em declarações `shared()` e `private()` de variáveis são uma das causas mais comuns de erros na programação paralela.

Redução

Agora já mostramos como criar threads e distribuir o trabalho por múltiplas threads. Porém, como fazer todas as threads trabalharem num resultado único – por exemplo, para somar os valores de um vetor? A função `reduction()` (**exemplo 2**) cuida disso.

O compilador cria uma cópia local de cada variável de `reduction()` e inicializa-as de forma independente do operador (por exemplo, `0` para `+` e `1` para `*`). Caso múltiplas threads estejam cuidando, cada uma, de uma parte do loop, a thread mestre soma os três subtotais ao final.

Quem é mais rápido?

Depurar programas paralelos é uma forma de arte. É especialmente difícil encontrar erros que não ocorram em programas seriais e não ocorram regularmente no processamento paralelo. Essa categoria inclui o que se conhece como condições de corrida: resultados diferentes em execuções repetidas do programa com múltiplos blocos executados em paralelo, dependendo de qual thread é a mais rápida em cada execução. O código do **exemplo 3** começa preenchendo um vetor em paralelo e depois prossegue calculando a soma desses valores em paralelo.

Sem a declaração do OpenMP `#pragma omp critical(soma_total)` na **linha 13**, pode ocorrer a seguinte condição de corrida:

- a thread 1 carrega o valor atual de `soma` num registrador da CPU;
- a thread 2 carrega o valor atual de `soma` num registrador da CPU;
- a thread 2 adiciona `a[i+1]` ao valor no registrador;
- a thread 2 grava o valor do registrador de volta na variável `soma`;
- a thread 1 adiciona `a[i]` ao valor no registrador;
- a thread 1 grava o valor do registrador na variável `soma`.

Exemplo 2: `reduction()`

```
01 a=0; b=0;
02 #pragma omp parallel for
➥private(i) shared(x, y, n)
➥reduction(+:a, b)
03 for (i=0; i<n; i++) {
04      a = a + x[i] ;
05      b = b + y[i] ;
06      }
```


© Linux New Media do Brasil Editora Ltda.

Exemplo 3: Evitar condições de corrida

```
01 #ifdef _OPENMP
02 #include <omp.h>
03 #endif
04 #include <stdio.h>
05 int main() {
06     double a[1000000];
07     int i;
08     #pragma omp parallel for
09     for (i=0; i<1000000; i++) a[i]=i;
10     double soma = 0;
11     #pragma omp parallel for shared (soma) private (i)
12     for (i=0; i<1000000; i++) {
13         #pragma omp critical (soma_total)
14         soma = soma + a[i];
15     }
16     printf("soma=%lf\n",soma);
17 }
```

Como a thread 2 ultrapassa a thread 1 nessa execução, ganhando assim a "corrida", `a[i+1]` não será calculado corretamente. Apesar da thread 2 calcular a soma e guardá-la na variável `soma`, a thread 1 a sobrescreve com um valor incorreto.

A declaração `#pragma omp critical` assegura que isso não ocorra. Todas as threads executam o código crítico, mas apenas um de cada vez. Com isso, o **exemplo 3** realiza a adição corretamente sem as threads paralelas atrapalharem o resultado. Para operações elementares (por exemplo, `i++`), `#pragma omp atomic` executará um comando de forma atômica. O acesso de escrita a variáveis `shared()` também é protegido quando se usa uma declaração `#pragma omp critical`.

Todos presentes?

Em alguns casos é necessário sincronizar todas as threads. A declaração `#pragma omp barrier` cria uma barreira virtual: todas as threads esperam até que a última delas alcance a barreira antes do processamento prosseguir. Mas é preciso pensar com cuidado antes de criar uma barreira artificial – fazer threads suspenderem o processamento diminuirá o ganho de desempenho obtido pelo uso do paralelismo. As threads que esperam não fazem nenhum trabalho. O **exemplo 4** ilustra um caso em que a barreira é inevitável.

A linha `Calcula()` desse exemplo calcula o segundo argumento com referência ao primeiro. Os argumentos nesse caso podem ser vetores, e a função de cálculo pode ser uma complexa operação matemática com matrizes. Aqui, é essencial usar `#pragma omp barrier` – qualquer falha na sincronia significaria que algumas threads começariam a segunda rodada de cálculos antes de os valores do cálculo em `B` ficarem disponíveis.

Alguns esquemas OpenMP (como `parallel`, `for`, `single`) incluem uma barreira implícita que pode ser explicitamente desativada com uma cláusula `nowait`, como em `#pragma omp for nowait`. Outros mecanismos de sincronização incluem:

- `#pragma omp master {código}`: código que será executado uma única vez e somente pela thread mestra;
- `#pragma omp single {código}`: código que será executado uma única vez, mas não necessariamente pela thread mestra;
- `#pragma omp flush (variáveis)`: variáveis em cache gravadas novamente na memória garantem uma visão consistente da memória.

Esses mecanismos de sincronização ajudarão a manter o código rodando tranqüilamente em ambientes multiprocessados.

Funções de biblioteca

A **tabela 2** lista algumas outras funções do OpenMP. Para usá-las, é preciso incluir o cabeçalho `omp.h` no código C/C++. Para garantir que o programa seja compilável também sem o OpenMP, é interessante acrescentar a linha `#ifdef _OPENMP` para a compilação condicional:

```
#ifdef _OPENMP
#include <omp.h>
threads = omp_get_num_threads();
#else
threads = 1
#endif
```

Funções de bloqueio (*locking*) permitem que uma thread bloqueie um recurso reservando acesso exclusivo (`omp_set_lock()`) a ele. Outras threads então podem usar `omp_test_lock()` para conferirem se o recurso está bloqueado. Essa configuração é útil caso se deseje que várias threads

Exemplo 4: Barreira inevitável

```
01 #pragma omp parallel shared (A, B, C)
02 {
03     Calcula(A,B);
04     printf("B foi calculado a partir de A\n");
05 #pragma omp barrier
06     Calcula(B,C);
07     printf("C foi calculado a partir de B\n")
```


© Linux New Media do Brasil Editora Ltda.

Snow666

Tabela 2: Funções do OpenMP

Função	Explicação
`int omp_get_num_threads()`	Retorna o número de threads.
`int omp_get_thread_num()`	Retorna o número da thread atual.
`void omp_set_num_threads(int)`	Define o número de threads a serem usadas em regiões paralelas futuras.
Funções de bloqueio	
`void omp_init_lock(omp_lock_t*)`	Inicializa uma trava (*lock*).
`void omp_set_lock(omp_lock_t*)`	Espera e depois define uma trava; bloqueia caso a trava não esteja disponível.
`int omp_test_lock(omp_lock_t*)`	Espera e depois define uma trava; não bloqueia se a trava não estiver disponível.
`void omp_unset_lock(omp_lock_t*)`	Remove uma trava.
`void omp_destroy_lock(omp_lock_t*)`	Destrói uma trava.

gravem dados num arquivo, mas seja preciso restringir acesso a uma thread por vez. Quando são usadas funções de bloqueio, é importante atentar para *deadlocks*.

Esse fenômeno ocorre quando threads precisam de recursos mas bloqueiam uma à outra. Por exemplo, quando a thread 1 bloqueia o recurso A e precisa usar o recurso B, enquanto a thread 2 faz o contrário, ambas as threads esperam para sempre.

Exemplo 5: Hello, world

```
01 /* helloworld.c (Versão OpenMP) */
02
03 #ifdef _OPENMP
04 #include <omp.h>
05 #endif
06 #include <stdio.h>
07 int main(void)
08 {
09     int i;
10 #pragma omp parallel for
11     for (i = 0; i < 4; ++i)
12     {
13         int id = omp_get_thread_num();
14         printf("Ola, mundo da thread %d\n", id);
15         if (id==0)
16             printf("Existem %d threads\n", omp_get_num_
➥threads());
17     }
18     return 0;
19 }
```

Variáveis de ambiente

Algumas variáveis de ambiente controlam o comportamento da execução dos programas OpenMP; a mais importante é `OMP_NUM_THREADS`. Ela especifica quantas threads conseguem operar em regiões paralelas, pois threads em excesso na realidade tornam o processamento mais lento. O comando `export OMP_NUM_THREADS=1` faz com que o programa rode com somente uma thread – exatamente como um programa serial normal.

Exemplo 6: Compilação do Hello world

```
$ gcc -Wall -fopenmp
➥helloworld.c
$ export OMP_NUM_THREADS=4
$ ./a.out
Ola mundo da thread 3
Ola mundo da thread 0
Ola mundo da thread 1
Ola mundo da thread 2
Existem 4 threads
```

Exemplo 7: Notificação

```
$ icc -openmp helloworld.c
helloworld.c(8): (col. 1)
remark:
OpenMP DEFINED LOOP WAS
PARALLELIZED.
```

$$\frac{\pi}{4} = \frac{1}{1} - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \cdots + \frac{(-1)^n}{2n+1} + \cdots$$

Figura 5 Fórmula de cálculo de pi de Gregor Leibniz.

Mãos à obra

Para usar o OpenMP, é preciso o conjunto de compiladores GCC na versão 4.2 ou posterior. Outros compiladores compatíveis são o da Sun [2], gratuito, e o da Intel [3], gratuito para uso não comercial.

O **exemplo 5** mostra uma versão em OpenMP do clássico *Hello World*. Para usar o OpenMP no GCC basta usar a opção `-fopenmp` na compilação. O **exemplo 6** mostra os comandos para compilar o programa juntamente com sua saída.

Se for usado o compilador da Sun, a opção é `-xopenmp`. Com o compi-

© Linux New Media do Brasil Editora Ltda.

Exemplo 8: Cálculo de pi

```
01 /* pi-openmp.c (Versão OpenMP) */
02
03 #include <stdio.h>
04 #define STEPCOUNTER 1000000000
05 int main(int argc, char *argv[])
06   {
07   long i;
08   double pi = 0;
09   #pragma omp parallel for reduction(+: pi)
10   for (i = 0; i < STEPCOUNTER; i++) {
11     /* pi/4 = 1/1 - 1/3 + 1/5 - 1/7 + ...
12     Para evitar precisar mudar
13     continuamente o sinal (s=1; a cada
14     etapa s=s*-1), adicionamos dois
15     elementos ao mesmo tempo. */
16     pi += 1.0/(i*4.0 + 1.0);
17     pi -= 1.0/(i*4.0 + 3.0);
18   }
19   pi = pi * 4.0;
20   printf("Pi = %lf\n", pi);
21   return 0;
22 }
```

Exemplo 9: Pi paralelo

```
$ gcc -Wall -fopenmp -o
➥ pi-openmp
pi-openmp.c
$ export OMP_NUM_THREADS=1
$ time ./pi-openmp
Pi = 3.141593
real    0m31.435s
user    0m31.430s
sys     0m0.004s
$ export OMP_NUM_THREADS=2
$ time
./pi-openmp
Pi = 3.141593
real    0m15.792s
user    0m31.414s
sys     0m0.012s
```

lador da Intel, a opção é `-openmp`. O compilador da Intel até notifica o programador quando algo é paralelizado (**exemplo 7**).

Benefícios?

Para um exemplo de aumento de desempenho com o OpenMP, vejamos um teste que calcula pi [4] com uso da fórmula de Gregory Leibniz (**exemplo 8** e **figura 5**). Esse método não é, de forma alguma, o mais eficiente para calcular o valor de pi; porém, o objetivo não é eficiência, mas fazer as CPUs trabalharem.

Paralelizar o loop `for()` com o OpenMP de fato otimiza o desempenho (**exemplo 6**). O programa roda com o dobro da velocidade quando usa duas CPUs em vez de uma, pois praticamente todo o cálculo pode ser paralelizado.

Se o programa for monitorado com a ferramenta *top*, é possível verificar que as múltiplas CPUs realmente estão trabalhando e que o programa `pi-openmp` usa mesmo 200% da CPU.

Esse efeito tão positivo sobre a velocidade do programa não será sempre tão pronunciado para qualquer programa. Nesse caso, é possível retornar à execução serial de trechos do programa. Os resultados sempre seguirão a Lei de Amdahl [5] (veja o **quadro 1** para uma explicação). ■

Quadro 1: Lei da Amdahl

"Speedup" descreve o fator pelo qual um programa pode ser acelerado por paralelismo. Num caso ideal, a execução de um programa com *N* processadores levaria somente 1/*N* do tempo exigido por um programa serial. Esse caso ideal é conhecido como "speedup linear". No mundo real, o speedup linear costuma ser impossível de alcançar porque certas partes do programa não são particularmente apropriadas para a paralelização.

Dada a parte de um programa que suporte paralelismo, *P* (portanto, 1 – *P* é a parte não paralelizável) e o número de processadores disponíveis, *N*, o speedup máximo é calculado pela fórmula:

$$\frac{1}{(1-P)+\frac{P}{N}}$$

Se a parte serial do programa (1–P) for 1/4, o speedup não pode ser maior que 4. Não importa quantos processadores sejam usados.

Mais informações

[1] OpenMP: http://www.openmp.org

[2] Compilador da Sun: http://developers.sun.com/sunstudio

[3] Compilador da Intel: http://www.intel.com/cd/software/products/asmo-na/eng/compilers/clin/

[4] Cálculo do Pi (Wikipédia): http://pt.wikipedia.org/wiki/Pi#M.C3.A9todos_de_c.C3.A1lculo

[5] Lei de Amdahl (Wikipédia): http://pt.wikipedia.org/wiki/Lei_de_Amdahl

© Linux New Media do Brasil Editora Ltda.

Linux.local

O maior diretório de empresas que oferecem produtos, soluções e serviços em Linux e Software Livre, organizado por Estado. Sentiu falta do nome de sua empresa aqui? Entre em contato com a gente: **11 4082-1300** *ou* **anuncios@linuxmagazine.com.br**

Fornecedor de Hardware = 1
Redes e Telefonia / PBX = 2
Integrador de Soluções = 3
Literatura / Editora = 4
Fornecedor de Software = 5
Consultoria / Treinamento = 6

Empresa	Cidade	Endereço	Telefone	Web	1	2	3	4	5	6
Ceará										
F13 Tecnologia	Fortaleza	Rua Coronel Solon, 480 – Bairro de Fátima Fortaleza - CE - CEP: 60040-270	85 3252-3836	www.f13.com.br		✔	✔		✔	✔
Espírito Santo										
Linux Shopp	Vila Velha	Rua São Simão (Correspondência), 18 – CEP: 29113-120	27 3082-0932	www.linuxshopp.com.br	✔	✔		✔	✔	
Megawork Consultoria e Sistemas	Vitória	Rua Chapot Presvot, 389 – Praia do Canto – CEP: 29055-410 sl 201, 202	27 3315-2370	www.megawork.com.br			✔		✔	✔
Spirit Linux	Vitória	Rua Marins Alvarino, 150 – CEP: 29047-660	27 3227-5543	www.spiritlinux.com.br			✔		✔	✔
Minas Gerais										
Instituto Online	Belo Horizonte	Av. Bias Fortes, 932, Sala 204 – CEP: 30170-011	31 3224-7920	www.institutoonline.com.br				✔		✔
Linux Place	Belo Horizonte	Rua do Ouro, 136, Sala 301 – Serra – CEP: 30220-000	31 3284-0575	corporate.linuxplace.com.br		✔	✔		✔	✔
Microhard	Belo Horizonte	Rua República da Argentina, 520 – Sion – CEP: 30315-490	31 3281-5522	www.microhard.com.br	✔	✔	✔		✔	✔
TurboSite	Belo Horizonte	Rua Paraíba, 966, Sala 303 – Savassi – CEP: 30130-141	0800 702-9004	www.turbosite.com.br	✔				✔	✔
Paraná										
iSolve	Curitiba	Av. Cândido de Abreu, 526, Cj. 1206B – CEP: 80530-000	41 252-2977	www.isolve.com.br		✔	✔			✔
Mandriva Conectiva	Curitiba	Rua Tocantins, 89 – Cristo Rei – CEP: 80050-430	41 3360-2600	www.mandriva.com.br			✔	✔	✔	✔
Telway Tecnologia	Curitiba	Rua Francisco Rocha 1830/71	41 3203-0375	www.telway.com.br					✔	✔
Rio de Janeiro										
Múltipla Tecnologia da Informação	Rio de Janeiro	Av. Rio Branco, 37, 14° andar – CEP: 20090-003	21 2203-2622	www.multipla-ti.com.br	✔		✔		✔	✔
NSI Training	Rio de Janeiro	Rua Araújo Porto Alegre, 71, 4º andar Centro – CEP: 20030-012	21 2220-7055	www.nsi.com.br				✔		✔
Open IT	Rio de Janeiro	Rua do Mercado, 34, Sl, 402 – Centro – CEP: 20010-120	21 2508-9103	www.openit.com.br				✔		✔
Unipi Tecnologias	Campos dos Goytacazes	Av. Alberto Torres, 303, 1ºandar – Centro – CEP: 28035-581	22 2725-1041	www.unipi.com.br			✔	✔	✔	✔
Rio Grande do Sul										
4up Soluções Corporativas	Novo Hamburgo	Pso. Calçadão Osvaldo Cruz, 54 sl. 301 CEP: 93510-015	51 3581-4383	www.4up.com.br		✔	✔		✔	✔
Definitiva Informática	Novo Hamburgo	Rua General Osório, 402 - Hamburgo Velho	51 3594 3140	www.definitiva.com.br	✔		✔		✔	✔
Solis	Lajeado	Av. 7 de Setembro, 184, sala 401 – Bairro Moinhos CEP: 95900-000	51 3714-6653	www.solis.coop.br		✔	✔	✔	✔	✔
DualCon	Novo Hamburgo	Rua Joaquim Pedro Soares, 1099, Sl. 305 – Centro	51 3593-5437	www.dualcon.com.br	✔		✔		✔	✔
Datarecover	Porto Alegre	Av. Carlos Gomes, 403, Sala 908, Centro Comercial Atrium Center – Bela Vista – CEP: 90480-003	51 3018-1200	www.datarecover.com.br	✔		✔			
LM2 Consulting	Porto Alegre	Rua Germano Petersen Junior, 101-Sl 202 – Higienópolis – CEP: 90540-140	51 3018-1007	www.lm2.com.br			✔		✔	✔
Lnx-IT Informação e Tecnologia	Porto Alegre	Av. Venâncio Aires, 1137 – Rio Branco – CEP: 90.040.193	51 3331-1446	www.lnx-it.inf.br	✔		✔		✔	✔
Plugin	Porto Alegre	Av. Júlio de Castilhos, 132, 11º andar Centro – CEP: 90030-130	51 4003-1001	www.plugin.com.br	✔		✔		✔	
TeHospedo	Porto Alegre	Rua dos Andradas, 1234/610 – Centro – CEP: 90020-008	51 3286-3799	www.tehospedo.com.br	✔	✔				
São Paulo										
Ws Host	Arthur Nogueira	Rua Jerere, 36 – Vista Alegre – CEP: 13280-000	19 3846-1137	www.wshost.com.br	✔		✔		✔	
DigiVoice	Barueri	Al. Juruá, 159, Térreo – Alphaville – CEP: 06455-010	11 4195-2557	www.digivoice.com.br	✔	✔	✔		✔	✔
Dextra Sistemas	Campinas	Rua Antônio Paioli, 320 – Pq. das Universidades – CEP: 13086-045	19 3256-6722	www.dextra.com.br			✔		✔	✔
Insigne Free Software do Brasil	Campinas	Av. Andrades Neves, 1579 – Castelo – CEP: 13070-001	19 3213-2100	www.insignesoftware.com			✔		✔	✔
Microcamp	Campinas	Av. Thomaz Alves, 20 – Centro – CEP: 13010-160	19 3236-1915	www.microcamp.com.br				✔		✔
PC2 Consultoria em Software Livre	Carapicuiba	Rua Edeia, 500 - CEP: 06350-080	11 3213-6388	www.pc2consultoria.com	✔					✔
Savant Tecnologia	Diadema	Av. Senador Vitorino Freire, 465 – CEP: 09910-550	11 5034-4199	www.savant.com.br	✔	✔	✔			✔
Epopéia Informática	Marília	Rua Goiás, 392 – Bairro Cascata – CEP: 17509-140	14 3413-1137	www.epopeia.com.br						✔
Redentor	Osasco	Rua Costante Piovan, 150 – Jd. Três Montanhas – CEP: 06263-270	11 2106-9392	www.redentor.ind.br	✔					
Go-Global	Santana de Parnaíba	Av. Yojiro Takaoca, 4384, Ed. Shopping Service, Cj. 1013 – CEP: 06541-038	11 2173-4211	www.go-global.com.br			✔		✔	✔
AW2NET	Santo André	Rua Edson Soares, 59 – CEP: 09760-350	11 4990-0065	www.aw2net.com.br			✔		✔	✔
Async Open Source	São Carlos	Rua Orlando Damiano, 2212 – CEP 13560-450	16 3376-0125	www.async.com.br	✔				✔	✔

© Linux New Media do Brasil Editora Ltda.

Empresa	Cidade	Endereço	Telefone	Web	1	2	3	4	5	6
São Paulo (continuação)										
Delix Internet	São José do Rio Preto	Rua Voluntário de São Paulo, 3066 9º – Centro – CEP: 15015-909	11 4062-9889	www.delixhosting.com.br	✔		✔		✔	
4Linux	São Paulo	Rua Teixeira da Silva, 660, 6º andar – CEP: 04002-031	11 2125-4747	www.4linux.com.br					✔	✔
A Casa do Linux	São Paulo	Al. Jaú, 490 – Jd. Paulista – CEP: 01420-000	11 3549-5151	www.acasadolinux.com.br			✔		✔	✔
Accenture do Brasil Ltda.	São Paulo	Rua Alexandre Dumas, 2051 – Chácara Santo Antônio – CEP: 04717-004	11 5188-3000	www.accenture.com.br			✔		✔	✔
ACR Informática	São Paulo	Rua Lincoln de Albuquerque, 65 –Perdizes – CEP: 05004-010	11 3873-1515	www.acrinformatica.com.br	✔					✔
Agit Informática	São Paulo	Rua Major Quedinho, 111, 5º andar, Cj. 508 – Centro – CEP: 01050-030	11 3255-4945	www.agit.com.br	✔	✔				✔
Altbit - Informática Comércio e Serviços LTDA.	São Paulo	Av. Francisco Matarazzo, 229, Cj. 57 – Água Branca – CEP 05001-000	11 3879-9390	www.altbit.com.br	✔		✔		✔	✔
AS2M -WPC Consultoria	São Paulo	Rua Três Rios, 131, Cj. 61A – Bom Retiro – CEP: 01123-001	11 3228-3709	www.wpc.com.br			✔		✔	✔
Big Host	São Paulo	Rua Dr. Miguel Couto, 58 – Centro – CEP: 01008-010	11 3033-4000	www.bighost.com.br	✔				✔	✔
Blanes	São Paulo	Rua André Ampére, 153 – 9º andar – Conj. 91 CEP: 04562-907 (próx. Av. L. C. Berrini)	11 5506-9677	www.blanes.com.br	✔	✔	✔		✔	✔
Commlogik do Brasil Ltda.	São Paulo	Av. das Nações Unidas, 13.797, Bloco II, 6º andar – Morumbi – CEP: 04794-000	11 5503-1011	www.commlogik.com.br	✔	✔	✔		✔	✔
Computer Consulting Projeto e Consultoria Ltda.	São Paulo	Rua Vergueiro, 6455, Cj. 06 – Alto do Ipiranga – CEP: 04273-100	11 5062-3927	www.computerconsulting.com.br	✔		✔		✔	✔
Consist Consultoria, Sistemas e Representações Ltda.	São Paulo	Av. das Nações Unidas, 20.727 – CEP: 04795-100	11 5693-7210	www.consist.com.br			✔	✔	✔	✔
Domínio Tecnologia	São Paulo	Rua das Carnaubeiras, 98 – Metrô Conceição – CEP: 04343-080	11 5017-0040	www.dominiotecnologia.com.br	✔					✔
EDS do Brasil	São Paulo	Av. Pres. Juscelino Kubistcheck, 1830 Torre 4 - 5º andar	11 3707-4100	www.eds.com		✔	✔			✔
Ética Tecnologia	São Paulo	Rua Nova York, 945 – Brooklin – CEP:04560-002	11 5093-3025	www.etica.net	✔		✔		✔	✔
Getronics ICT Solutions and Services	São Paulo	Rua Verbo Divino, 1207 – CEP: 04719-002	11 5187-2700	www.getronics.com/br			✔		✔	✔
Hewlett-Packard Brasil Ltda.	São Paulo	Av. das Nações Unidas, 12.901, 25º andar – CEP: 04578-000	11 5502-5000	www.hp.com.br	✔		✔	✔	✔	✔
IBM Brasil Ltda.	São Paulo	Rua Tutóia, 1157 – CEP: 04007-900	0800-7074 837	www.br.ibm.com	✔		✔		✔	✔
iFractal	São Paulo	Rua Fiação da Saúde, 145, Conj. 66 – Saúde – CEP: 04144-020	11 5078-6618	www.ifractal.com.br			✔		✔	✔
Integral	São Paulo	Rua Dr. Gentil Leite Martins, 295, 2º andar Jd. Prudência – CEP: 04648-001	11 5545-2600	www.integral.com.br	✔				✔	
Itautec S.A.	São Paulo	Av. Paulista, 2028 – CEP: 01310-200	11 3543-5543	www.itautec.com.br	✔	✔	✔		✔	✔
Kenos Consultoria	São Paulo	Av: Fagundes Filho, 13, Conj 53 – CEP: 04304-000	11 40821305	www.kenos.com.br					✔	✔
Konsultex Informatica	São Paulo	Av. Dr. Guilherme Dumont Villares, 1410 6 andar, CEP: 05640-003	11 3773-9009	www.konsultex.com.br			✔		✔	✔
Linux Komputer Informática	São Paulo	Av. Dr. Lino de Moraes Leme, 185 – CEP: 04360-001	11 5034-4191	www.komputer.com.br	✔		✔		✔	✔
Linux Mall	São Paulo	Rua Machado Bittencourt, 190, Cj. 2087 – CEP: 04044-001	11 5087-9441	www.linuxmall.com.br	✔			✔	✔	
Livraria Tempo Real	São Paulo	Al. Santos, 1202 – Cerqueira César – CEP: 01418-100	11 3266-2988	www.temporeal.com.br				✔	✔	✔
Locasite Internet Service	São Paulo	Av. Brigadeiro Luiz Antonio, 2482, 3º andar – Centro – CEP: 01402-000	11 2121-4555	www.locasite.com.br	✔				✔	✔
Microsiga	São Paulo	Av. Braz Leme, 1631 – CEP: 02511-000	11 3981-7200	www.microsiga.com.br			✔		✔	✔
Novatec Editora Ltda.	São Paulo	Rua Luis Antonio dos Santos, 110 – Santana – CEP: 02460-000	11 6979-0071	www.novateceditora.com.br				✔		
Novell América Latina	São Paulo	Rua Funchal, 418 – Vila Olímpia	11 3345-3900	www.novell.com/brasil			✔		✔	✔
Oracle do Brasil Sistemas Ltda.	São Paulo	Av. Alfredo Egídio de Souza Aranha, 100 – Bloco B – 5º andar – CEP: 04726-170	11 5189-3000	www.oracle.com.br					✔	✔
Proelbra Tecnologia Eletrônica Ltda.	São Paulo	Av. Rouxinol, 1.041, Cj. 204, 2º andar Moema – CEP: 04516-001	11 5052- 8044	www.proelbra.com.br	✔		✔			✔
Provider	São Paulo	Av. Cardoso de Melo, 1450, 6º andar – Vila Olímpia – CEP: 04548-005	11 2165-6500	www.e-provider.com.br			✔		✔	✔
Red Hat Brasil	São Paulo	Av. Brigadeiro Faria Lima, 3900, Cj 81 8º andar Itaim Bibi – CEP: 04538-132	11 3529-6000	www.redhat.com.br			✔		✔	✔
Samurai Projetos Especiais	São Paulo	Rua Barão do Triunfo, 550, 6º andar – CEP: 04602-002	11 5097-3014	www.samurai.com.br			✔		✔	✔
SAP Brasil	São Paulo	Av. das Nações Unidas, 11.541, 16º andar – CEP: 04578-000	11 5503-2400	www.sap.com.br			✔		✔	✔
Simples Consultoria	São Paulo	Rua Mourato Coelho, 299, Cj. 02 Pinheiros – CEP: 05417-010	11 3898-2121	www.simplesconsultoria.com.br			✔		✔	✔
Smart Solutions	São Paulo	Av. Jabaquara, 2940 cj 56 e 57	11 5052-5958	www.smart-tec.com.br		✔	✔		✔	✔
Snap IT	São Paulo	Rua João Gomes Junior, 131 – Jd. Bonfiglioli – CEP: 05299-000	11 3731-8008	www.snapit.com.br			✔		✔	✔
Stefanini IT Solutions	São Paulo	Av. Brig. Faria Lima, 1355, 19º – Pinheiros – CEP: 01452-919	11 3039-2000	www.stefanini.com.br			✔		✔	✔
Sun Microsystems	São Paulo	Rua Alexandre Dumas, 2016 – CEP: 04717-004	11 5187-2100	www.sun.com.br	✔		✔		✔	✔
Sybase Brasil	São Paulo	Av. Juscelino Kubitschek, 510, 9º andar Itaim Bibi – CEP: 04543-000	11 3046-7388	www.sybase.com.br					✔	✔
The Source	São Paulo	Rua Marquês de Abrantes, 203 – Chácara Tatuapé – CEP: 03060-020	11 6698-5090	www.thesource.com.br			✔		✔	✔
Unisys Brasil Ltda.	São Paulo	R. Alexandre Dumas 1658 – 6º, 7º e 8º andares – Chácara Santo Antônio – CEP: 04717-004	11 3305-7000	www.unisys.com.br	✔		✔		✔	✔
Utah	São Paulo	Av. Paulista, 925, 13º andar – Cerqueira César – CEP: 01311-916	11 3145-5888	www.utah.com.br			✔		✔	✔
Visuelles	São Paulo	Rua Eng. Domicio Diele Pacheco e Silva, 585 – Interlagos – CEP: 04455-310	11 5614-1010	www.visuelles.com.br			✔		✔	✔
Webnow	São Paulo	Av. Nações Unidas, 12.995, 10º andar, Ed. Plaza Centenário – Chácara Itaim – CEP: 04578-000	11 5503-6510	www.webnow.com.br	✔		✔		✔	
WRL Informática Ltda.	São Paulo	Rua Santa Ifigênia, 211/213, Box 02– Centro – CEP: 01207-001	11 3362-1334	www.wrl.com.br	✔		✔		✔	
Systech	Taquaritinga	Rua São José, 1126 – Centro - Caixa Postal 71 – CEP: 15.900-000	16 3252-7308	www.systech-ltd.com.br	✔	✔			✔	
2MI Tecnologia e Informação	Embu	Rua José Bonifácio, 55 – Jd. Independência – SP CEP: 06826-080	11 4203-3937	www.2mi.com.br		✔	✔		✔	✔

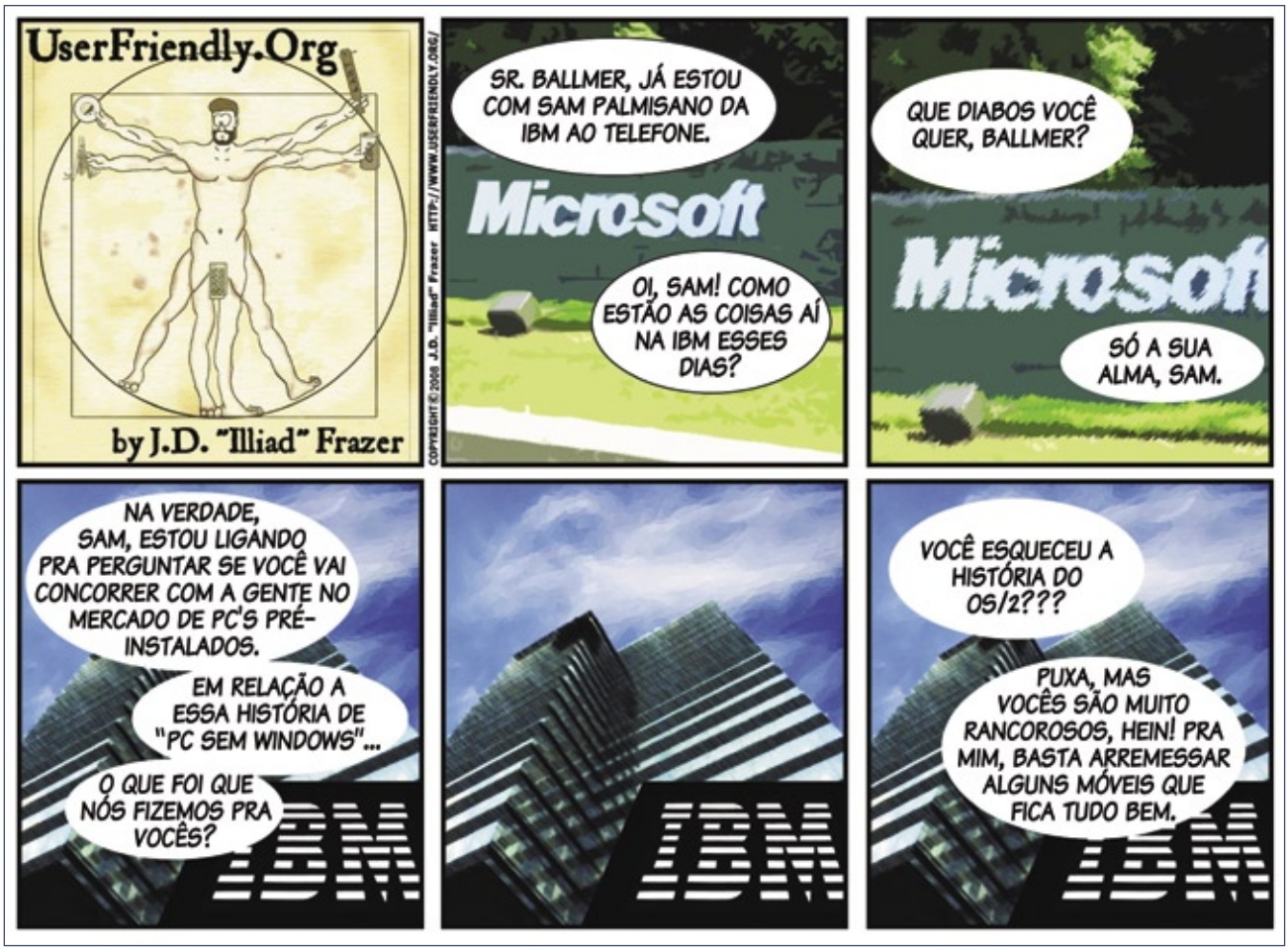
© Linux New Media do Brasil Editora Ltda.

Calendário de eventos

Evento	Data	Local	Website
Rails Summit Latin America	15 e 16 de outubro	São Paulo, SP	www.locaweb.com.br/rails
4º AtualTEC 2008/ Fórum de Software Livre do Cone Norte	15 a 17 de outubro	Boa Vista, RR	www.faculdadeatual.edu.br
Conisli	18 e 19 de outubro	São Paulo, SP	www.conisli.org
Futurecom 10	27 a 30 de outubro	São Paulo, SP	www.futurecom2008.com.br
Latinoware 2008	30 de outubro a 1 de novembro	Foz do Iguaçu, PR	www.latinoware.org
PHP Conference Brasil	27, 28 e 29 de novembro	Osasco, SP	www.phpconf.com.br

Índice de anunciantes

User Friendly – Os quadrinhos mensais da Linux Magazine



© Linux New Media do Brasil Editora Ltda.

Snow666

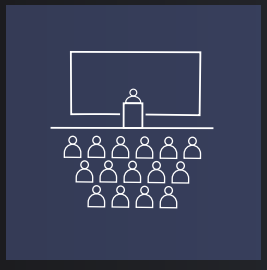
futurecom
SÃO PAULO • ANO 10

27 a 30 de outubro, 2008 • Transamerica Expo Center • **São Paulo**

Futurecom, o mais Qualificado evento de Telecom e TI da América Latina!

foto : Carlos Alkmin

Business Trade Show com aproximadamente **230 empresas,** demonstrando Serviços, Aplicações, Soluções, Sistemas e Tecnologia. A exposição abrangerá 3 pavilhões do Centro de Convenções Transamerica, com aproximadamente **20.000m² de área,** reunindo participantes de **mais de 35 países.**



mais de

10 Sessões Premium com presenças confirmadas de Presidentes de Operadoras.

85 Sessões de Marketing e Business visando o Desenvolvimento de Negócios e Relacionamento, no tradicional modelo de Palestras e Painéis Político-Estratégicos do Futurecom.

60 Sessões Técnicas abordando o mais moderno estado-da-arte em Tecnologia.

Faça já sua inscrição a preços promocionais!
www.futurecom.com.br

futurecom Organização e Promoção: Provisuale
(41) 3314-3200 • www.provisuale.com.br

© Linux New Media do Brasil Editora Ltda.

Snow666

PREVIEW

Na Linux Magazine #48

DESTAQUE

Autenticação e identidade

Apesar dos anos de constantes inovações de alta tecnologia, a senha continua sendo um recurso fundamental da maioria das redes. Há várias ferramentas para consolidação, criptografia e sincronização de senhas, porém, a menos que a sua empresa invista maciçamente em smartcards ou outras novas tecnologias, você precisará fazer login em algum lugar.

A Linux Magazine 48 de aniversário vai apresentar algumas técnicas para tornar a autenticação de usuários no Linux mais simples e segura. Use *one-time passwords* com as ferramentas OPIE e OTPW para não repetir uma mesma senha inúmeras vezes, autentique usuários no Apache de forma segura tanto na Internet quanto em intranets, aprenda de uma vez por todas a integrar clientes Linux em redes *Active Directory* com uso do recurso Windbind do *Samba* e conheça o *OpenID*, uma solução para autenticação única em várias contas. ■

REDES

Nagios

O Nagios tem grande destaque entre os administradores de redes. Além de ser muito competente no que faz, sua flexibilidade é praticamente ilimitada.

Um dos motivos disso é o *Nagios Remote Plugin Executor*, um módulo capaz de realizar verificações de forma ativa na máquina remota, sem depender da interpretação dos dados por parte do Nagios. Para quem não deseja o monitoramento ativo, há também o NSCA, um módulo que faz a verificação passivamente.

A Linux Magazine 48 de aniversário vai apresentar as diferenças entre os módulos NRPE e NSCA, além de explicar como instalar e configurar o primeiro. ■

Na EasyLinux #14

Monitores gigantes

Os monitores LCD de 19 polegadas já têm preços bem melhores que há um ano. Será que já chegou o momento de você comprar aquele monitor cinematográfico? Quais são as vantagens e desvantagens do LCD em relação aos antigos monitores de tubo? Até onde 21 polegadas valem mais que 19? Na Easy Linux 14, vamos comparar marcas, modelos e tecnologias de monitores à venda no Brasil para orientar suas compras. ■

A melhor parte de todos os sistemas

Empresas como Apple e Microsoft investem pesado em design para deixarem seus sistemas mais atraentes. Como resultado, tanto o Mac OS quanto o Windows Vista têm forte apelo visual. Porém, engana-se quem pensa que é impossível alcançar um grau de beleza semelhante no Linux. Na Easy Linux 14, vamos mostrar o caminho das pedras para deixar seu Linux com a cara do Mac OS X, do Vista e do Windows XP, seja por pura diversão ou para facilitar o uso do Linux por quem já está habituado a esses sistemas e tem dificuldade de adaptação ao pingüim. ■

© Linux New Media do Brasil Editora Ltda.

Snow666

NovaForge™

Nós conectamos nossos Clientes a nossos Centros de Competências de Software Livre

NovaForge, no centro da abordagem Industrial para Desenvolvimento de Sistemas da Bull.

O NovaForge é um poderoso conjunto de ferramentas e serviços amplamente testados e projetados para reduzir o esforço, otimizar custos de gestão e cronogramas, garantindo a qualidade dos produtos finais em Projetos de Desenvolvimento de Sistemas. O NovaForge foi concebido para ser utilizado em Projetos de Desenvolvimento e Atualização de Aplicações em ambientes J2EE, PHP e .net , na manutenção de aplicações desenvolvidas por terceiros e para o teste profissional e integrado dos sistemas.

BULL

Architect of an Open World™

www.bull.com

© Linux New Media do Brasil Editora Ltda.

Snow666

IBM, o logo IBM, ibm.com e Smart SOA são marcas registradas ou de titularidade da International Business Machines Corporation nos Estados Unidos da América, em outros países ou em ambos. Caso estes e outros termos protegidos, na primeira vez em que aparecem nesta informação, estejam marcados com os símbolos ® ou ™, isto significa que os mesmos constituem marcas registradas ou marcas comerciais de titularidade da IBM nos Estados Unidos da América na ocasião em que esta informação foi publicada. Tais marcas podem também constituir marcas registradas ou marcas comerciais em outros países. Uma relação atualizada das marcas de titularidade da IBM está disponível no Web site "Informações sobre direitos autorais e marcas" ibm.com/legal/copytrade.shtml

© Dutra New Media do Brasil Editora Ltda.