SERVIÇOS SOBRE LINUX p.24
Os inovadores serviços da Veus Technology se baseiam em LAMP

CEZAR TAURION p.26
O ecossistema do Código Aberto e como entrar nele

IBM SEM MICROSOFT p.20
Big Blue vai oferecer desktops "Microsoft-Free"

LINUX NEW MEDIA
The Pulse of Open Source

46 Setembro 2008

LINUX MAGAZINE

A REVISTA DO PROFISSIONAL DE TI

RASTREANDO HACKERS

DEPOIS DE INVADIR UMA REDE, O HACKER PODE APAGAR ARQUIVOS, ALTERAR PROGRAMAS E ROUBAR DADOS. APRENDA A RASTREAR ESSAS AÇÕES E A SE RECUPERAR DOS DANOS p.28

- » Análise forense com o avançado Sleuth Kit p.30
- » Recupere arquivos apagados p.34
- » Ferramentas em Linux para restauração do Windows p.38
- » O framework forense OCFA é usado pela polícia p.44

REDES: NAGIOS p.58
Acompanhar arquivos de log é muito fácil com o plugin check_logfiles

SEGURANÇA: ACLS p.64
As permissões de arquivos legadas do Unix não são suficientes para as exigências modernas. Garanta a segurança com ACLs

VEJA TAMBÉM NESTA EDIÇÃO:

- » As novidades do kernel 2.6.26 p.17
- » LPI nível 2: Segurança do sistema p.46
- » Avaliamos um appliance VoIP que roda Linux e Asterisk p.54
- » Websites bonitos e funcionais com AJAX p.72

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

Expediente editorial

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

Editor-chefe
Tadeu Carmona
tcarmona@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

Assistente de Arte
Rafael Carvalho
rcarvalho@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

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

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

Novos modelos

Prezados leitores,

Como bem sabemos, não existe tal coisa como "o ano do Linux no desktop". Todo ano é declamada a previsão de que o sistema aberto finalmente conquistará o mercado dos computadores pessoais nos próximos 365 dias, mas os números não mudam muito de um ano para o outro. O Linux cresce firmemente e a um ritmo acelerado, mas a expectativa de que um eventual "ano do Linux no desktop" se inicie com 1% dos PCs rodando Linux e termine com uma fatia de 20% ou mais é simplesmente irreal.

No entanto, isso não significa que os incentivos já dados pelos fabricantes que vendem desktops com Linux pré-instalado sejam infrutíferos. É altamente positivo para o Linux como um todo que sua imagem seja associada às marcas mais tradicionais e conhecidas de PCs.

Ainda assim, há outras formas potencialmente mais interessantes para essas empresas, como a IBM parece ter finalmente descoberto. Usar o Linux como plataforma para alavancar seus próprios produtos, sejam eles proprietários ou livres, constitui uma modalidade ainda pouco explorada de emprego do sistema operacional aberto. Ao promover o uso de seu pacote proprietário Lotus sobre um sistema aberto, a IBM não combate projetos como OpenOffice.org e Mozilla Thunderbird – diferentemente do que muitos podem crer a princípio – ela apenas promove a diversidade, o que sabemos que é positivo quando são usados padrões abertos, como é o caso. Além disso, significa que a "Big Blue" ajudará também a promover ou desenvolver todos os programas de código aberto subjacentes.

Contudo, é duvidosa a eficácia de um ataque tão direto à Microsoft. Considerando que a ausência dos "aplicativos habituais" é um dos principais fatores de resistência e desistência quanto ao uso do Linux por parte dos novos usuários, anunciar desktops "Microsoft-Free" talvez tenha o efeito inverso ao desejado.

Ou então, pode ser que o uso direto do termo reforce para o público geral a idéia de que existem formas de usar um desktop sem os programas do Windows, como a Apple já vem fazendo com grande sucesso.

De qualquer forma, desejo muito sucesso ao Linux nessa nova empreitada.

Pablo Hess
Editor

CAPA

COLUNAS

NOTÍCIAS

CORPORATE

TUTORIAL

ANÁLISE

REDES

SEGURANÇA

PROGRAMAÇÃO

SERVIÇOS

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.*

Vídeo de notebook

Estou com um problema de reconhecimento da placa de vídeo do meu notebook, uma SiS Mirage 3+. Já testei algumas distribuições e não consegui fazê-las reconhecerem a placa. O site da SiS não disponibiliza o driver para Linux.

Elton Andrade Nascimento

Resposta

Caro Elton, a solução para seu problema parece estar nas mãos da comunidade do código aberto, pois o fabricante desse chip de vídeo não produz drivers abertos. Essa situação, felizmente, é cada vez mais rara, pois os maiores fabricantes de chips gráficos oferecem um crescente suporte ao Linux.

*No fórum do Ubuntu (*http://tinyurl.com/5nc428*), sugere-se o uso de um driver desenvolvido por um programador voluntário, mas que precisa ser pedido pessoalmente por email (*http://tinyurl.com/5vlm25*).*

O fórum do Ubuntu tem mais instruções de uso do driver. Espero que isso resolva o problema.

Receptor de TV digital

Comprei recentemente um receptor de TV Digital USB (Mic TV) e, como todo equipamento recém lançado, não tem suporte para Linux. Não consegui fazê-lo funcionar no Ubuntu 8.04, nem pelo Wine. Existe alguma maneira de resolver esse problema?

Márcio Monentenegro

Resposta

Prezado Márcio, de fato ainda é difícil fazer qualquer afirmação geral sobre receptores de TV Digital, pois cada dispositivo tem suas particularidades e exige um procedimento diferente. Esperamos que a solução para isso se torne mais fácil em breve.

*O local ideal para buscar dispositivos com suporte pelo kernel Linux é o wiki do projeto Linux TV (*http://tinyurl.com/34svla*).*

Google e Firefox 3

Fiquei chocado hoje: peguei emprestado de um colega de trabalho a Linux Magazine 44. Quando estava lendo a notícia sobre a nova API do Google para o Google Earth via navegador, me deparei com algo atordoante, que obriga os usuários do Firefox que utilizam a nova, estável e empolgante versão (3.0) a retornarem à versão anterior, pois o Google não elaborou o suporte ao Firefox 3.

Pergunto-lhes: Como uma empresa que presa pelos valores do SL/CA como o Google pode elaborar uma API para um IE7 e esquecer do Firefox 3? Aliás, seria questão de alterar algumas linhas no código-fonte. Será que seus programadores entraram em férias ou dormiram na planilha de cronogramas?

Elton Schivei Costa
São Paulo, SP


intel
Open Source @ Intel
Promovendo o poder da inovação
A Intel está comprometida com a inovação – todos os dias criamos soluções e softwares abertos, contribuindo com comunidades de código aberto através de uma grande variedade de projetos, desde virtualização e mobilidade a gráficos e desempenho eficiente. Compartilhe seus conhecimentos e aprenda mais. Vamos inovar juntos.
www.intel.com/opensource
© 2008 Intel Corporation. Intel, o logotipo Intel são marcas registradas da Intel Corporation ou de suas subsidiárias nos Estados Unidos e em outros países. Todos os direitos reservados.
© Intel do Brasil Ltda. 2008

COLUNA

Pergunte ao Klaus!

Klaus Knopper

É fácil perder seus dados importantes num disco, mas apagar efetivamente todos os dados do disco é mais complicado do que se imagina.

por Klaus Knopper

Apagamento de disco

Quero apagar todos os arquivos do meu disco, incluindo o *Lilo* ou *Grub*, pois vou me desfazer dele. Até o momento, consegui apagar somente alguns dados, mas o Grub continua lá.

Resposta

Se eu entendi corretamente a sua pergunta, seu objetivo é apagar todos os dados gravados no disco, ou seja, sobrescrever todos os arquivos e a estrutura de diretórios, e ainda apagar o registro de inicialização (MBR). Isso engloba vários níveis dos dados.

Em primeiro lugar, a parte dos dados estruturados, isso é, o sistema de arquivos. É nesse nível que ficam os arquivos e diretórios.

O segundo nível é a tabela de partições, que descreve a subdivisão do disco e engloba, em geral, os primeiros 512 bytes do disco. Esses 512 bytes têm anotações do início e fim das partições, além de informações sobre o propósito de cada partição (a chamada ID da partição) e uma marcação que diz se a partição é ou não inicializável. Essas informações são lidas pela BIOS durante a inicialização para descobrir onde está ou estão a(s) partição(ões) inicializável(is).

Quando você formata um sistema de arquivos com o comando `mkfs` correspondente, ele não apaga o conteúdo do sistema em questão, mas apenas cria nele uma raiz "vazia". Com isso, programas como o *Foremost* são capazes de reconhecer assinaturas de sistemas de arquivos conhecidos e recuperar corretamente vários arquivos que não tenham sido danificados.

Para apagar os dados de uma partição, é preciso não somente formatá-la, mas sobrescrever todo o seu conteúdo. Na maioria dos casos, como novas instalações de um sistema, simplesmente substituir todo o conteúdo por zeros é suficiente; porém, há métodos forenses capazes de recuperar o estado anterior a esses zeros. Portanto, é um pouco mais seguro usar, em vez de zeros, dados aleatórios, e repetir o procedimento algumas vezes para eliminar rastros de estados anteriores da mídia magnética. Mas note que, no caso de memórias *Flash*, isso não é necessário.

Vou dar alguns exemplos para a destruição completa dos dados, mas note que isso vai apagar todo o conteúdo do seu disco. Então, assegure-se de que esteja se referindo ao disco correto (`/dev/sda` é apenas um exemplo).

Para destruir a tabela de partições e todos os metadados do sistema de arquivos, além de sobrescrever todos os dados do disco, pode-se usar um dos dois comandos abaixo (o segundo é mais rápido):

```
cat /dev/urandom > /dev/sda
dd if=/dev/urandom of=/dev/sda bs=1024k
```

O programa *Wipe*, por sua vez, deve gerar um resultado ainda mais confiável, pois grava dados aleatórios várias vezes no disco – porém, ele também demora bem mais.

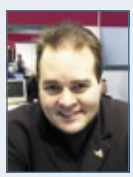
Considerando-se que também é importante apagar o disco inteiro, e não apenas uma partição, é importante se lembrar dos primeiros 512 bytes do disco, pois contêm a MBR. Apague-a, juntamente com a tabela de partições, com o comando:

```
dd if=/dev/zero of=/dev/sda bs=512 count=1
```

Além disso, como vamos usar números aleatórios, o processo de apagamento do disco inteiro pode ser intenso para a CPU, o que, somado ao grande tamanho dos dicos rígidos atuais, certamente tornará o procedimento demorado.

Se o seu objetivo for manter as partições e livrar-se da MBR, recomendo ferramentas como o *install-mbr* ou o *ms-sys*, capazes de produzir MBRs vazias ou que iniciam a partir de uma única partição. ■

Sobre o autor



Klaus Knopper é o criador do *Knoppix* e co-fundador do evento *Linux Tag*. Atualmente trabalha como professor, programador e consultor.

Participe de um dos maiores eventos de Software Livre do Centro-Oeste.

Faça a sua inscrição e/ou submissão de palestra.
Acesse o nosso site www.festivalsoftwarelivre.org
e faça parte dessa festa!

Realização

ATA Associação de Tecnologias Abertas · SERPRO · Universidade Católica de Brasília

Organização

Training Tecnologia
Treinamentos e Consultoria em TI

Local: Universidade Católica de Brasília
3 E 4 DE OUTUBRO
Para maiores informações, acesse o nosso site
www.festivalsoftwarelivre.org
ou ligue para (61) 3223-0995

COLUNA

Knockd

Charly Kühnast

Histórias de terror sempre têm personagens assustadores batendo em portas à noite. No Linux, isso é chamado de port knocking e pode ser muito útil.

por Charly Kühnast

Se você prefere não ter uma porta administrativa óbvia para seu firewall *Iptables* – mas precisa de uma secreta –, a técnica do *port knocking* é uma opção interessante que pode evitar ataques baseados em scripts. Para o administrador ambicioso e paranóico, a ferramenta mais adequada é o *Knockd*[1].

O pacote inclui dois componentes: o *knock* é o cliente que envia sinais de batida nas portas, enquanto o *daemon knockd* os recebe.

Batida

Para monitorar o processo, o cliente precisa somente dos números das portas nas quais deve bater e da opção `-v`:

```
knock -v 10.0.0.42 7000 8000 9000
```

A ferramenta responde com a saída na linha de comando mostrada na **figura 1**.

O arquivo de configuração `/etc/knockd.conf` permite que o administrador do sistema especifique a ação que o daemon deve realizar ao receber uma batida válida. O **exemplo 1** ilustra isso.

Código Morse

Se reconhecer o sinal, o knockd abre a porta 22 para o IP correto, que passa seu próprio IP.

Ao se bater nas portas na seqüência errada, o daemon pára o acesso SSH. Há ainda uma outra opção – o `knockd.conf` – que é mais ou menos assim:

```
start_command = /usr/sbin/iptables -A INPUT -s
➥%IP% -p tcp --syn --dport 22 -j ACCEPT
```

```
File  Edit  View  Terminal  Tabs  Help
charly@funghi:~$ knock -v 10.0.0.42 7000 8000 9000
hitting tcp 10.0.0.42:7000
hitting tcp 10.0.0.42:8000
hitting tcp 10.0.0.42:9000
charly@funghi:~$
```

Figura 1 Se reconhecer o sinal das batidas, a ferramenta responde.

Exemplo 1: /etc/knockd.conf

```
01 [options]
02   logfile = /var/log/knockd.log
03 [openSSH]
04   sequence = 7000,8000,9000
05   seq_timeout = 5
06   command = /sbin/iptables -A INPUT -s
➥%IP% -p tcp --dport 22 -j ACCEPT
07   tcpflags = syn
08 [closeSSH]
09   sequence = 9000,8000,7000
10   seq_timeout = 5
11   command = /sbin/iptables -D INPUT -s
➥%IP% -p tcp --dport 22 -j ACCEPT
12   tcpflags = syn
```

```
cmd_timeout = 10
stop_command = /usr/sbin/iptables -D INPUT -s
➥%IP% -p tcp --syn --dport 22 -j ACCEPT
```

Depois de bater, o daemon inicia o `start_command` e espera um número de minutos especificado em `cmd_timeout` antes de executar o `stop_command`.

Conclusão

Administradores de sistema realmente paranóicos vão adorar a opção de configurar um arquivo com a seqüência de portas. Cada seqüência expira após ser usada. ■

Mais informações

[1] Knockd: http://www.zeroflux.org/cgi-bin/cvstrac.cgi/knock/wiki

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.

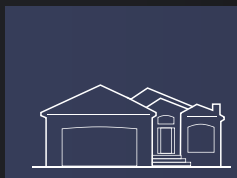
futurecom
SÃO PAULO • ANO 10

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

COLUNA

O desastre do SSL no Debian

Insegurança

Veja o que podemos aprender com o desastre do OpenSSL no Debian.

por Kurt Seifried

Depois de um acidente de avião, começam as investigações. Os investigadores revelam que a maioria dos acidentes com aviões se devem a erro humano ou a alguma nova confluência de circunstâncias jamais prevista. Conseqüentemente, o meio aéreo é um dos mais seguros por quilômetro por passageiro.

Software é diferente. Diferentemente de um acidente aéreo, quando ocorre um problema com um software, uma resposta típica é simplesmente resolver o problema imediato com um *patch* para o código-fonte, o que nada faz para solucionar o problema real. Portanto, estamos num estado permanente de tratamento de sintomas, mas nunca dos problemas de fato.

Como esses problemas jamais são corrigidos, continuamos a ver as mesmas falhas de software repetidamente (criação de arquivos temporários, estouros de *buffer* e pilha, entre outros).

Este mês, vamos investigar o desastre do OpenSSL no *Debian* como parte de um esforço para encontrar os problemas reais.

Resumidamente, o problema ocorreu quando um mantenedor do pacote do OpenSSL no Debian executou o *Valgrind*, uma ferramenta de análise de código, no OpenSSL e descobriu vários usos de memória não inicializada. O uso de memória não inicializada é potencialmente perigoso porque não há como saber o conteúdo dessa memória – todo `0`, todo `1` ou o código de um agressor, para listar algumas possibilidades.

O mantenedor então foi até a lista de emails *openssl-dev* e perguntou sobre o seguinte código:

```
MD_Update(&m,buf,j);
```

Várias respostas depois, o desenvolvedor decidiu que era seguro eliminar o código do pacote OpenSSL do Debian. Essas alterações foram repassadas para o Debian e a vida continuou como de costume.

O código em questão aparece duas vezes: uma em `ssleay_rand_bytes()` e outra em `ssleay_rand_add()`. Embora o código seja idêntico, ele possui duas funções bem diferentes. Em `ssleay_rand_bytes()`, o código simplesmente retorna dados aleatórios da memória para o buffer. Porém, na função `ssleay_rand_add()`, ele tenta ser inteligente e adicionar memória não inicializada ao *pool* de entropia. Na melhor das hipóteses, ele melhora a entropia, enquanto que, no pior deles, nada afeta.

Esse buffer é usado como fonte primária de entropia para qualquer aplicativo que use o OpenSSL (a menos que usem uma fonte personalizada de PRNG – geração de números pseudo-aleatórios). Ao comentá-la inteiramente, o desenvolvedor eliminou praticamente toda a aleatoriedade usada durante a criação da chave pela maioria dos aplicativos. O único dado "aleatório" restante usado durante a criação da chave era o ID do processo, reduzindo o espaço de chaves de 2^(um número grande, como 128 ou 1024) para 2^16 (ou menos, em certos casos). Ops.

O que deu errado?

Muitas coisas deram errado e, assim como na maioria dos desastres, tudo ficaria melhor se a série de eventos houvesse sido quebrada em algum ponto. Em vez disso, todo administrador de Debian precisou aplicar um patch em cada uma de suas máquinas e gerar novamente cada chave criptográfica que tivesse sido criada nos últimos dois anos.

Um dos problemas imediatos foi o código que fazia algo desnecessário de uma forma potencialmente perigosa: adicionar memória não inicializada à aleatoriedade verdadeira não é um ganho e faz com que o código pareça estar fazendo algo completamente diferente. Além disso, o código também não era bem documentado.

É por isso que equipamentos militares têm avisos gigantes como "aponte este lado para o inimigo".

Provavelmente, se esse código fosse mais fácil de entender, o desenvolvedor não teria precisado ir até a lista openssl-dev em busca de ajuda. Isso nos leva facilmente ao segundo problema: dificuldade de comunicação.

Ao lidar com código ou problemas relacionados a segurança, é crucial se comunicar com as pessoas certas ou com o fórum certo. Caso contrário, pode-se receber uma resposta aparentemente responsável, mas incorreta na realidade – talvez porque a questão tenha sida mal entendida, ou porque a pergunta estava simplesmente errada. No caso que estamos estudando, parece que tudo o que podia dar errado deu errado.

A equipe do OpenSSL alega que a pergunta foi feita na lista errada, enquanto outras pessoas dizem que a lista supostamente correta não é mostrada.

Além disso, a pergunta era um tanto imprecisa: o código dado como exemplo não mostrava o contexto completo e, em razão da natureza do código, isso pode ter levado a um mal-entendido a respeito do que estava acontecendo.

Incluir no email um prefácio como "Se você souber de alguém ou algum fórum mais apropriado para responder esta pergunta, por favor me diga" provavelmente é mais eficaz do que "É aqui o lugar mais adequado para perguntar?" e certamente melhor do que simplesmente disparar um email e esperar uma resposta responsável.

Além disso, quem relata uma falha pode verificar as mensagens do gerenciador de versão (CVS, SVN, Git...) em busca de quem faz muitas alterações no código afetado, para descobrir quem é o provável responsável pelo código em questão.

Como se pode ver, encontrar a pessoa certa pode ser um grande desafio e documentar claramente quem e como contactar pode ser de grande ajuda para evitar problemas com qualquer software.

Alteração de código

Como os empacotadores do Debian mantêm seus próprios repositórios de código, as alterações no código-fonte não foram notadas por pessoas fora do projeto, mas foram amplamente distribuídas. Se o patch do código-fonte tivesse sido enviado oficialmente pelo Debian para os desenvolvedores do OpenSSL, provavelmente eles levantariam bandeiras vermelhas e anulariam as mudanças. Isso leva a uma situação na qual, mesmo que o projeto original continue atualizando e mantendo o software, um simples patch mantido pelo Debian poderia introduzir uma falha sutil – ou, nesse caso, significativa. A solução para esse problema – enviar todos os patches *upstream* para inclusão – não é simples. Outra possibilidade é submeter oficialmente todos os patches upstream para revisão.

Falta de teste

Por último, o problema mais difícil. Na indústria de aviões, os materiais, projetos e componentes inteiros são testados até a destruição para verificar quanto abuso podem sofrer antes de falharem. Até onde sei, não existe uma plataforma de teste para o OpenSSL que gere um número de chaves estatisticamente significativo – por exemplo, várias centenas de milhares ou milhões – e depois as analise para verificar sua aleatoriedade.

Além disso, mesmo que existisse tal plataforma, ela precisaria ser aplicada regularmente a novas versões – ou seja, não apenas à versão upstream oficial, mas a todas que possuíssem modificações aplicadas por distribuidores.

É claro que esse tipo de plataforma de teste deveria ser aplicada a todos os produtos, como o teste de firewalls, por exemplo.

Conclusão

Infelizmente, é muito mais barato, a curto prazo, simplesmente tratar os sintomas mais perigosos da má engenharia de software do que resolver suas causas. Entretanto, a longo prazo, isso leva ao gasto de muito tempo dos usuários finais e desenvolvedores para resolver repetidamente os mesmos problemas.

A boa notícia é que muitas soluções para esses problemas não são tão caras, além de, a maioria requer pouca tecnologia para ser implementada.

Simplesmente comentar código, documentar canais de comunicação e fazer perguntas claras já ajuda muito. Além disso, é importante lembrar que Código Aberto não se trata apenas de acesso ao código-fonte, mas da própria cultura que escreve o código-fonte, o que significa que todos têm chance de ajudar a torná-lo muito melhor. ■

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.

COLUNA

Crônicas do kernel

Zack Brown

No verão no hemisfério norte, todos querem emagrecer o kernel, seja eliminando os firmwares binários ou pondo a série 2.4 em descanso.

por Zack Brown

Fora, firmware!

David Woodhouse quer eliminar todos os firmwares de terceiros do código-fonte do kernel. Ele é favorável a deixar o kernel carregar trechos arbitrários de código de firmware (contanto que o binário resultante possa ser legalmente distribuído sob sua licença), mas não acredita que eles pertençam ao kernel. David está tentando criar uma árvore *git* separada para todos eles.

Várias respostas reclamaram disso. A maioria das pessoas era a favor de isolar o firmware num único local, mas retirá-lo da árvore pareceu demais, pois criou questões de pacotes de firmware concorrentes e outros requisitos que algumas pessoas não querem resolver.

O maior oponente da idéia de David foi David S. Miller, que vem combatendo essa idéia há tempos, especificamente em relação ao driver *tg3*, que ele acha que precisa manter seu firmware na árvore principal. David Miller disse que se distribuições como o *Debian* quiserem evitar dados binários em seu kernel, cabe a elas escrever e manter os *patches* necessários fora da árvore.

David Woodhouse respondeu que já havia seguido os passos para garantir que mesmo sem eliminar o firmware, suas alterações ainda seriam úteis, mas ele achava que seria melhor retirá-las inteiramente. Nesse ponto, a discussão se tornou uma consideração técnica sobre como a idéia poderia ser implementada. No final, ficou claro que vários desenvolvedores têm opiniões diferentes, cada um com seus motivos legítimos. Em relação aos patches de Woodhouse, espero algum tipo de meio-termo. Eliminar o firmware é preferível para puristas do Software Livre, mas também pode ser preferível por motivos legais e práticos. Nesse caso, David Miller poderia perder a batalha, mas sem enfrentar muitos inconvenientes.

Direção para o 2.4

Depois do que pareceu um longo intervalo, Willi Tarreau vem liberando mais versões do kernel 2.4 ultimamente. Além disso, ele se esforçou para compilar algumas estatísticas a respeito de quem usa qual kernel e por que não atualizam para a série 2.6. Com apenas 22 respostas, os números de Willy têm uma margem de erro um tanto grande, mas ainda são interessantes. Aproximadamente metade dos pesquisados disseram que dependem do 2.4 para servidores de propósito geral, nos quais quedas e problemas de regressão de atualizações seriam inconvenientes para várias pessoas, e então não se interessam em atualizar para o 2.6, mas mantêm a última versão do 2.4. Cerca de 20% dos pesquisados usam o 2.4 para servidores de aplicações específicas, que tendem a ser de "missão crítica" nos quais qualquer queda – até mesmo a atualização para uma versão mais recente do 2.4 – seria um grande problema.

Aproximadamente 10% dos pesquisados usam o 2.4 para roteadores, firewalls e outras aplicações de rede. Nesses casos, atualizar para o 2.6 poderia ser relativamente fácil em razão dos poucos requisitos do sistema, mas como esses administradores não têm certeza de como fazer isso sem problemas, simplesmente ficam com o que conhecem.

Outros 10% dos pesquisados usam o 2.4 em seus produtos com sistemas embarcados. Nesse caso, a atualização do sistema dos dispositivos seria visível para os usuários e exigiria mudanças substanciais em todo o processo de montagem da empresa.

O resto das pessoas, aproximadamente outros 10%, rodam o 2.4 em seus sistemas pessoais, seja por não desejarem configurar um novo kernel para esse hardware, seja por estarem acostumadas a usar o sistema conforme ele se encontra.

O objetivo da pesquisa foi descobrir como incentivar essas pessoas a adotarem a série 2.6 para que a 2.4 possa partir dessa para uma melhor. ■

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.

Sempre aparece alguém prometendo a solução para todos os seus problemas. A diferença é que a Itautec cumpre a promessa.

Segurança e Infra-estrutura. É-exatamente-o-que-eu-preciso.

Saiba tudo o que acontece na sua empresa com a Tecnologia Itautec. Para você ter mais segurança e integração, a Itautec dá aos seus clientes controle, monitoramento e gerenciamento com recursos administrados pela Tecnologia Itautec. Para dar tranqüilidade na gestão de seus negócios, oferece a você um serviço de infra-estrutura que fornece ferramentas de cabeamento, instalação, atualização e suporte. Se você precisa de segurança, integração e comodidade, conte com a **Tecnologia Itautec, a melhor tradução de TI.**

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

COMPROMISSO COM A SUSTENTABILIDADE

Itautec

COLUNA

Profissionais móveis e o Linux nos netbooks

Augusto Campos

O netbook é uma fonte permanente de recursos de produtividade, conexão e até diversão.

por Augusto Campos

São 14h18 de uma segunda-feira de inverno e acabo de receber um email da redação da Linux Magazine me informando que já está na hora de preparar a coluna para a edição de setembro. Nada de especial nisso, exceto que estou na confortável sala de espera de uma imobiliária, com a perspectiva de aguardar mais meia hora por aqui enquanto eles agilizam o que vim solicitar.

Ocorre que já faz seis meses que passei pela maior revolução para esta minha profissão virtual desde a chegada da banda larga: a adoção de um netbook. A minha escolha foi o Eee PC e, a bem da verdade, eu já estou no meu segundo: passei rapidamente por um modelo 701, com sua tela de 7 polegadas e disco interno de 4GB, e agora sou um feliz usuário de um Eee PC 900, com tela de 9 polegadas e 20 GB de disco. Ambos rodando Linux, com a distribuição instalada por padrão pela fabricante.

> ***Faz seis meses que passei pela maior revolução para esta minha profissão virtual desde a chegada da banda larga: a adoção de um netbook.***

O Eee pesa menos de um quilo, o que me permite levá-lo comigo o dia todo. Não tem grande poder de processamento ou armazenamento, mas o nome *Netbook* explica a magia que ele exerce: você não precisa de grande processamento local se puder contar com a variedade de serviços via Internet acessíveis quase em qualquer lugar das médias e grandes cidades, seja via celular (discretamente, com uma conexão Bluetooth entre o PC e o fone), ou usando as cada vez mais comuns redes Wi-Fi, como é o meu caso agora – bastou pedir a senha para a recepcionista.

O fator preço é uma das razões para a inclusão do Linux em várias das alternativas de netbooks, mas o fato de ele ser bem adaptado para atuar em redes usando protocolos abertos, como é o caso da Internet – e da crescente gama de serviços da chamada Web 2.0 –, o torna uma escolha ainda melhor. Grande parte do que eu faço no Eee ocorre dentro da janela do navegador, incluindo a composição desta coluna, que está ocorrendo no processador de textos do Google, e em seguida será enviada ao editor via Gmail. Muito raramente eu preciso aguardar para fazer alguma atividade só quando chegar no escritório porque o Eee não deu conta – exceto em situações de ausência de conectividade, mas isso vem se tornando cada vez mais raro.

Se a sua rotina envolve deslocamentos em que você gostaria de ter acesso a um PC com Web, informe-se sobre os netbooks. E não apenas o Eee – existem várias outras alternativas, disponíveis no (ou a caminho do) mercado menos ou mais formal, incluindo marcas como Acer, Dell, HP, Everex, Lenovo, Asus, Positivo e MSI.

Eu uso o meu todos os dias, assisto a vídeos nele, ouço músicas, envio e recebo arquivos nos mais variados formatos, converso online, consulto as mais variadas informações, espaireço com um eventual joguinho nas salas de espera da vida e... escrevo e envio a minha coluna da Linux Magazine em tempo recorde, tornando produtivo um período de 45 minutos que, de outra forma, eu teria que ocupar lendo um exemplar de uma revista de fofocas de pelo menos três meses atrás que está aqui na mesinha em minha frente. Valeu a pena! ■

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.

Pablo Hess

Após duas versões frenéticas, a evolução – não revolução – foi o tom da última versão do Linux.

por Pablo Hess

Com um ar mais de inovação do que de manutenção, foi lançado no dia 13 de julho o kernel Linux 2.6.26, após 87 dias de desenvolvimento e 10.487 *commits*. Entre as novidades, novos drivers para webcams e redes sem fio, além da expansão natural da solução de virtualização embutida no kernel.

Ritmo menos louco

Nas versões anteriores (2.6.24 e 2.6.25, principalmente), a quantidade de novidades no kernel foi simplesmente surpreendente, batendo todos os recordes de alterações de arquivos estabelecidos anteriormente. No 2.6.26, o ritmo foi mais lento, o que não significa, de forma alguma, que eles tenham "dormido no ponto" – afinal, ainda foram mais de 10 mil commits.

Infra-estrutura

Mesmo com Linus Torvalds deixando bem claro que não aprecia o depurador do kernel, KGDB, o recurso foi incluído na versão mais recente. Além disso, o recurso de *Page Attribute Table* presente em CPUs *x86* finalmente poderá ser utilizado, o que pode acelerar o sistema como um todo.

As montagens em *bind* (*bind mounts*) já existiam no Linux há diversos anos, auxiliando a implantação de ambientes *chroot*. Agora, com o suporte a bind mounts em regime "somente leitura", expandem-se as possibilidades de uso de contêineres (como fazem os projetos *OpenVZ* e *Linux VServer*) com maior segurança.

O testador de memória *memtest*, instalado por diversas distribuições por padrão, agora tem suas funcionalidades parcialmente cobertas pelo kernel. Basta iniciá-lo com o parâmetro `memtest`.

Como Zack Brown vem noticiando em sua coluna mensal, os semáforos vêm sendo atacados por alguns desenvolvedores, como Ingo Molnar, por exemplo. Com o uso de *mutexes*, muitos semáforos não são mais necessários para manter o bom desempenho do kernel, já começando a ser substituídos por outros mais genéricos. O escalonador do kernel, CFS, também continua a receber melhorias e ajustes. Agora já é possível, por exemplo, definir o valor de *nice* para grupos em processadores com múltiplos núcleos ou sistemas multiprocessados, mas infelizmente isso não será perceptível para usuários de desktop.

Novos drivers

A inclusão do driver Linux-UVC permite o uso de centenas de modelos de webcams já existentes, além de preparar o caminho para modelos futuros.

Além disso, com a chegada dos leitores Blu-ray, o kernel suporta agora o padrão UDF 2.5, usada com freqüência nessas mídias.

Já no campo das comunicações, as redes *mesh* (especificação 802.11s, ainda em estágio de rascunho) também já podem funcionar com o sistema do pingüim, graças à inclusão do código do projeto *open80211s* no kernel.

As redes 802.11n (próxima geração do padrão Wi-Fi) também continuam avançando, e a infra-estrutura *mac80211* já suporta seus novos recursos.

Virtualização

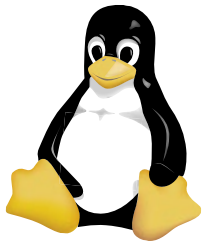
A *kernel-based virtual machine*, ou KVM, foi incluída no kernel 2.6.20 e finalmente chegou a arquiteturas além das tradicionais x86 e *x86-64*: no momento, ela já pode ser usada em *IA64*, *PPC* e *S390*.

Além disso, já começaram os trabalhos para implementação de paravirtualização no KVM, que suportava apenas a virtualização completa. Além disso, o número de processadores virtuais suportados para cada máquina virtual aumenta para 16.

Futuro

Com o uso da árvore *linux-next* já a pleno vapor, as próximas versões do kernel devem ser mais práticas para todos os desenvolvedores – em especial, Linus –, pois os erros são corrigidos bem antes da abertura da "janela de inclusão de código". Pelo que se pode ver nela, ao menos o KVM reserva várias novidades. ■

Linux domina os subnotebooks



O Linux de fato veio para ficar nos pequeníssimos computadores ultra-portáteis, ou subnotebooks. A taiwanesa MSI já comercializava seu Wind no exterior há poucos meses e anunciou uma parceria com a Novell para oferecer a opção de pré-instalação do *SUSE Linux Enterprise Desktop 10* no modelo – que antes só oferecia a opção do Windows XP.

Antes do anúncio, duas gigantes do mercado de laptops de "tamanho normal" também anunciaram sua entrada no fervilhante mercado das máquinas ultra-portáteis.

Dell

Após Asus, Acer, HP, Positivo, dentre diversos outros fabricantes de PCs, a Dell divulgou que estrearia em grande estilo, com cinco modelos diferentes e preços a partir de US$ 300.

São duas linhas de produtos, uma (linha *E*) voltada ao mercado hoje dominado pelo EeePC – os chamados *netbooks* – e a outra (linha *E Slim*) destinada a concorrer com notebooks tradicionais, porém pequenos e mais caros, como o MacBook Air da Apple. A linha E será equipada com tela de 8,9 polegadas (contra 12,1 da linha E Slim), processador Intel Atom de 1,6 GHz e 512 MB de RAM, com cerca de 5 horas de autonomia da bateria e aproximadamente um quilo.

A linha mais cara contará ainda com memória flash ou disco rígido de 1,8 polegadas à escolha do cliente, e seu peso deve exceder em pouco o limite de 1 kg.

Lenovo

A outra novidade do mercado coube à chinesa Lenovo, que adquiriu a divisão de computadores pessoais da IBM no final de 2004 por US$ 1,75 bilhões. Ela anunciou que lançará em outubro os modelos Ideapad S9 e S10, com custo em torno de US$ 400.

Os novos Ideapads, da Lenovo, tem espessura de 27,5 mm e pesam 1,1 kg. Com tela de 10 polegadas, estará disponível nos clássicos preto e branco, mas também em cores variadas, como azul, vermelho e rosa.

O modelo S9 é equipado com uma tela de 8,9 polegadas, enquanto o S10 é dotado de uma tela de 10 polegadas. Uma novidade no produto da Lenovo é a disponibilidade de um slot para cartões do tipo ExpressCard, de uso cada vez mais freqüente, por exemplo, para conexão de dispositivos de acesso móvel à Internet. O touchpad é sensível a múltiplos toques (*multitouch*) e reconhece, por conta disso, o contato de dois dedos, podendo interpretá-los como rolagem de página, rotação ou redimensionamento da imagem na tela. As características adicionais dos Ideapads são similares às dos netbooks já disponíveis no mercado: processador Intel Atom (modelo N270) com 1,6 GHz, webcam de 1,3 megapixels, WiFi, Ethernet, três entradas USB, leitores de cartões e peso pouco acima de 1 kg.

O modelo S9 típico deverá ser comercializado nos Estados Unidos por um preço em torno de US$ 400. Já o S10 deverá custar em torno de US$ 450 em sua configuração padrão. Entretanto, segundo o anúncio oficial da Lenovo, haverá versões mais baratas do S10, com preços já a partir de US$ 399. Ambos os netbooks podem ser equipados com até 1 GB de RAM e podem ser adquiridos opcionalmente com um disco-rígido de 160 GB ou com um cartão de memória flash de 4 GB.

Em comum

A decisão desses fabricantes de ingressar nesse mercado tem seus motivos: a Lenovo cita um recente estudo da IDC que afirma que, em 2012, mais de 9 milhões de ultra-portáteis deverão ser comercializados no mundo.

A oferta da opção pelo Linux é marcante nesse segmento, num forte contraste com o mercado de notebooks tradicionais, nos quais o sistema do pingüim jamais alcançou fatia significativa.

No caso específico da MSI, o emprego da versão Enterprise da distribuição da Novell talvez indique que a Novell está apostando numa maior penetração dessa categoria de equipamentos no mercado corporativo – provavelmente seguindo as indicações da IDC. ■

Firefox 3 em Qt?

Dedicados desenvolvedores do projeto Mozilla, em parceria com a Nokia, conseguiram portar a mais nova versão do Firefox para o conjunto de bibliotecas multiplataforma de desenvolvimento de interface gráfica Qt.

O código é baseado no tronco da versão 1.9.x. As primeiras atividades nesse sentido começaram no início do ano, já havendo uma primeira versão de testes do Firefox 3 para download, com a interface gráfica desenvolvida com o toolkit que serve de base para o projeto KDE. A Nokia deseja integrar o navegador ao *Qtopia*, sistema operacional para dispositivos móveis desenvolvido pela Trolltech, adquirida pela empresa finlandesa. ■

DragonFly BSD 2.0 lançado

Foi lançada no final de julho a versão 2.0 do sistema operacional Unix DragonFly BSD. Derivado do FreeBSD, essa é a oitava versão do projeto e contém como um de seus principais recursos um novo sistema de arquivos de alto desempenho, além de melhor suporte de hardware, novos programas e correções de falhas.

O sistema de arquivos Hammer tem funcionalidades semelhantes às do famoso ZFS da Sun Microsystems, com capacidade de gerar *snapshots*, desfazer operações e retornar o sistema a um estado anterior. Além disso, elimina a necessidade de verificação do sistema de arquivos com o comando `fsck` após quedas de energia ou outras falhas. ■

VirtualBox lançado desfalcado

A versão mais recente (1.6.4) do software de virtualização *VirtualBox*, de código aberto e recém adquirido pela Sun Microsystems, trouxe uma surpresa desagradável: faltam dois arquivos (Makefiles, mais especificamente), sem os quais não é possível compilar o programa.

Há formas de solucionar o problema, mas o mais surpreendente foi a declaração da Sun de que a falha será corrigida somente na próxima versão do software.

IBM anuncia desktops "Microsoft-Free"

A IBM anunciou ontem em São Francisco, EUA, uma parceria com alguns dos principais distribuidores Linux para oferecer desktops absolutamente livres de softwares da Microsoft. O uso direto do termo "Microsoft-Free" é o que mais marca no anúncio, pois deixa bem clara a intenção da Big Blue de manter seu antigo parceiro longe de seus consumidores.

No acordo com Canonical, Novell e Red Hat, a empresa afirma que "juntará forças globalmente com seus parceiros de hardware para fornecer escolhas de computação pessoal livres da Microsoft" e, logo em seguida, informa o motivo da parceria: os computadores virão "com Lotus Notes e Lotus Symphony no mercado mundial de um bilhão de unidades de desktops em 2009". Os dois softwares, em conjunto, oferecem todas as funcionalidades de colaboração, mensagens instantâneas, email, navegador, calendário e escritório, entre outras, e sua disseminação em desktops poderia representar importantes avanços na presença da IBM no mercado e também nas vendas do Lotus Domino, que integra o pacote Lotus no lado do servidor.

O anúncio cita que os parceiros percebem "mudanças nas forças do mercado e a demanda crescente por alternativas aos caros computadores baseados em Windows e Office" como "um conjunto ideal de circunstâncias que permitem a proliferação de desktops baseados em Linux no próximo ano".

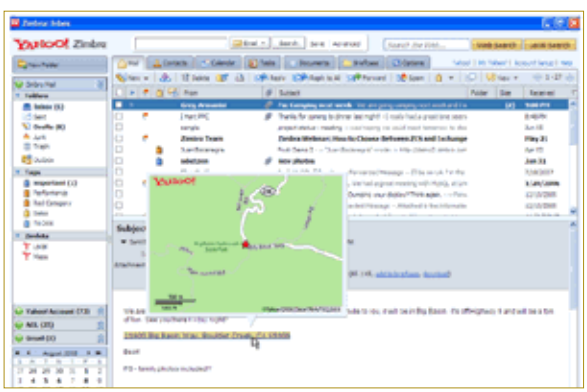
A demora na adoção do Vista em ambientes corporativos também foi citada por Kevin Cavanaugh, vice-presidente da IBM para softwares Lotus, como motivo de expectativas elevadas com relação à adoção do Linux pelas empresas no futuro próximo.

O modelo de comercialização e parcerias com ISVs, fornecedores de hardware e prestadores de serviço em cada mercado local deve variar de acordo com o setor de atuação do cliente. O anúncio cita o exemplo de clientes de governo, cujos ISVs parceiros forneceriam aplicativos de gerenciamento de documentos, de crise e serviços aos cidadãos, enquanto bancos contariam com suporte a thin clients e softwares Lotus sociais para a comunicação entre aplicações de seus diversos setores. Enquanto isso, em escolas, estudantes e professores disporiam de "uma plataforma aberta de baixo custo para capitalizar as forças dos softwares Lotus colaborativos e sociais", segundo o texto. ■

Zimbra Desktop em teste

Zimbra, a empresa especialista em soluções *groupware* de mesmo nome adquirida pelo Yahoo em setembro de 2007, liberou para download uma versão preliminar do seu cliente de email web escrito em AJAX, o *Zimbra Desktop*. A Yahoo Public License, licença de código aberto sob a qual o código do aplicativo está disponível, foi ajustada, de modo a torná-la compatível com o Fedora.

A nova versão do Zimbra tem como base o projeto *Prism*, da Mozilla Foundation. Por conta disso, o sistema dispõe de uma tela de abertura e — dependendo do sistema operacional utilizado — até mesmo integração com a barra de menus e a área de notificação, como se fosse um aplicativo comum, instalado localmente. Versões para Linux, Mac OS X e Windows podem ser baixadas diretamente do site do projeto. ■



UE e a qualidade do SL

A União Européia lançou a versão 0.8.1 de sua plataforma *Alitheia*, dedicada à avaliação automática de projetos de software. A 0.8.1 é descrita pelos desenvolvedores como a primeira versão alfa utilizável.

O software coleta metadados dos projetos a partir de listas de email, repositórios, código-fonte e bancos de dados de falhas. Os dados são processados com critérios específicos e depois — no caso mais simples — o Alitheia avalia todos os códigos acrescentados, as contribuições de desenvolvedores individuais ou o número de falhas.

O nome Alitheia significa "verdade", em grego. O objetivo do projeto é fornecer nada menos que provas científicas da qualidade dos softwares de código aberto. Ele auxiliará os desenvolvedores a monitorarem e melhorarem a qualidade para, com isso, aumentar a aceitação do código aberto em geral. ■

Google Code bane licença Mozilla

Há dois anos, quando o Google Code foi lançado, a MPL (*Mozilla Public License*) era uma das licenças suportadas pelo projeto, juntamente com diversas outras de código aberto, como Apache, BSD, GPL e LGPL. Agora, no entanto, houve uma mudança nessa política.

Ao que parece, o Google segue a proibição da MPL praticada, recentemente, pela FSF Affero GPL. Essa decisão, todavia, pode não ter sido muito ponderada, já que fará com que uma série de projetos abandonem o Google Code, trocando-o por serviços concorrentes. Tal como acontece com a Affero, as razões da decisão do Google não são totalmente claras. ■

Via libera documentação

Após anunciar sua iniciativa de fomento do desenvolvimento de drivers abertos para Linux, a fabricante de semicondutores Via liberou três guias de programação para alguns de seus dispositivos (num total de 800 páginas), o que abre as portas para a criação de drivers GPL para funcionamento com Linux.

A liberação da documentação vem após alguns meses com apenas um simples driver para o framebuffer no kernel Linux, o que já estava enfraquecendo a crença nas palavras anteriores do fabricante taiwanês. Quatro dias antes de compartilhar com o Código Aberto suas especificações, a Via contratou o alemão Harald Welte, fundador do site *GPL Violations.org* e programador envolvido com os projetos OpenMoko e Netfilter, para atuar como seu porta-voz e dar orientações para a empresa frente à comunidade do Código Aberto.

Os dispositivos cobertos pelos três guias são seu PadLock e os chipsets CX700 e VX800/820. ■

Alfresco com suporte a SharePoint

A partir de sua versão 3 beta, o sofware de gerenciamento de documentos Alfresco, um produto de código aberto, passa a ter suporte ao protocolo de comunicação do Microsoft SharePoint, software de gerenciamento e colaboração de documentos da empresa de Redmond.

O que está por trás da mais nova funcionalidade do Alfresco, no entanto, não é a benevolência de seu rival, muito menos os frutos de algum dentre os diversos acordos de interoperabilidade que a Microsoft tem levado adiante como bandeira de sua aproximação das empresas e do mercado do Código Aberto. A fonte dessa "abertura" está em recente decisão da União Européia, que obriga a Microsoft a liberar o código-fonte dos protocolos utilizados em todo o seu conjunto de aplicativos de escritório, o Microsoft Office, o que inclui as especificações de comunicação do SharePoint.

Com isso, o Alfresco passa a ser capaz de "emular" um servidor SharePoint, com a vantagem de não depender, exclusivamente, de um servidor MS SQL para prover serviços de bancos de dados à aplicação: bancos de dados Oracle, DB2 ou MySQL são inteiramente suportados. Além disso, não há mais a necessidade de se utilizar o Internet Explorer como navegador-cliente da aplicação: com o Alfresco, qualquer navegador passa a ser um cliente em potencial. ■

- **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

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

Entrevista com Marcelo Botelho, diretor da Veus Technology

Software Livre como serviço

Com um serviço inovador completamente implementado em Software Livre, a Veus Technology é um exemplo de sucesso no uso do código aberto.

por Pablo Hess

A empresa carioca Veus Technology foi criada com o objetivo de promover a comunicação entre fornecedores de serviços e os consumidores. Hoje, seu serviço de maior destaque é o fornecimento de resultados médicos para pacientes por meio de mensagens via telefone celular, o qual já conta com diversos clientes, incluindo alguns dos principais laboratórios de exames clínicos do país.

Marcelo Botelho, diretor da Veus Technology.

O diretor da empresa, Marcelo Botelho, contou à **Linux Magazine** como o Software Livre contribui para o sucesso da Veus Technology em todos os mercados em que atua.

Linux Magazine» ***Por favor, conte-nos um pouco da história da Veus e por que a empresa escolheu a área de software para laboratórios.***

Marcelo Botelho» A Veus Technology é uma empresa de tecnologia que nasceu voltada para a área da saúde, já que a atuação da sua equipe está dedicada a esse segmento há mais de dez anos. Observamos que existia uma oportunidade de serviços para os laboratórios de análises clínicas e hospitais em relação à integração e interação do cliente com os serviços das empresas de saúde.

O primeiro enfoque foi transformar o telefone celular num canal importante de comunicação entre o paciente e sua prestadora de serviços de saúde. Consultar seus laudos, receber alertas de laudos prontos, poder enviá-los por email e encaminhar para um fax foram os pontos iniciais para criarmos a Veus Technology.

A empresa foi criada há três anos, para fornecer informações aos seus clientes por meio de vários canais de comunicação, sempre aliando facilidade e comodidade. Atualmente, temos soluções para celulares (voz, SMS e acesso móvel), totens (terminais de auto-atendimento) e Web (envio por email ou fax).

LM» ***Que particularidades tem o mercado de softwares para laboratórios?***

MB» O mercado de tecnologia para laboratórios, clínicas e hospitais é bastante competitivo, com grandes players atuantes e, portanto, requer constante atualização e renovação tecnológica.

Por isso, nossa empresa busca trazer constantes inovações. Somos uma empresa focada em inovações e já temos clientes em vários Estados brasileiros.

Lidamos com multinacionais que investem muito em pesquisa e contamos apenas com o nosso talento. Ainda assim, mesmo sem incentivos externos, criamos programas inovadores e únicos no Brasil.

LM» ***Quais os diferenciais funcionais do seu produto nesse mercado?***
MB» Destaco o fato de termos sido a primeira empresa no Brasil a oferecer soluções completas, no modelo "turn-key", no mercado móvel para a área de saúde. Nosso produto, o Portal Veus Saúde, está presente nas maiores operadoras de celular do Brasil.

Consideramos como um destaque funcional absoluto da empresa o fato de termos uma solução que se integra automaticamente com vários canais de comunicação, de forma transparente para a empresa e o cliente.

Um laudo recebido pela Veus pode ser acessado pelo cliente por meio do celular, num totem ou pela Web, ou ainda pode disparar o envio de um SMS de "laudo pronto" para o cliente, além de um alerta médico para o médico solicitante (se for o caso). Todos de forma integrada e automática e sempre utilizando uma única identificação do cliente.

É importante destacar que não enviamos qualquer informação médica por SMS, mas somente alertas pontuais, como "Seu laudo está pronto" ou "Entre em contato com a unidade xyz". Além disso, acessando pelo celular, com senha, o paciente pode ver o resultado de seus exames, até mesmo de imagem, sem quebras.

LM» ***Por que a Veus opta pelo Código Aberto sempre que possível? Que vantagens isso traz?***
MB» Considero dois pontos em especial: qualidade e preço. Ambos, extremamente relevantes.

Os produtos que usamos para o desenvolvimento das nossas aplicações são leves e funcionais, sendo a base da nossa estrutura a pilha LAMP (Linux, Apache, MySQL e PHP).

Temos facilidade no uso das ferramentas, além de acesso rápido e dinâmico para as nossas dúvidas, nas diversas listas e fóruns pelo mundo.

E quando falamos em competitividade e preços finais, o fato de podermos contar com soluções leves e de código aberto ajudam a tornar os nossos serviços mais interessantes para o mercado.

LM» ***E quais são as desvantagens?***
MB» Ainda estamos tentando descobrir quais são as desvantagens, no nosso caso, em usar soluções de código aberto.

LM» ***De que forma os clientes (laboratórios) percebem a questão do uso de código aberto?***
MB» Os clientes não costumam se envolver quanto à plataforma que usamos. Desenvolvemos de modo que as soluções sejam acessíveis a qualquer sistema operacional e diferentes navegadores web.

A maioria dos nossos clientes deseja simplesmente que a aplicação seja operável via Web, sem qualquer restrição.

Para o cliente final, não é notado o fato de ser ou não de código aberto, pois o que ele exige é segurança, desempenho e estabilidade da aplicação.

LM» ***A Veus se envolve com as comunidades desenvolvedoras de projetos abertos?***
MB» Sim, temos participação constante em fóruns e eventos diretamente ligados à nossa plataforma. Posso acrescentar que existe, na nossa equipe, uma "adoração" pela plataforma com que trabalhamos.

LM» ***Os seus softwares oferecidos como serviços são hospedados num único datacenter?***
MB» Todos os nossos serviços são hospedados em datacenter, incluindo nosso servidor de desenvolvimento. Utilizamos três datacenters distintos para garantir segurança e estabilidade a nossas aplicações e não pretendemos ter um datacenter próprio.

Se o cliente desejar que os nossos serviços estejam obrigatoriamente "dentro" do seu ambiente, podemos instalar a aplicação na empresa. Porém, ainda não tivemos essa solicitação.

LM» ***A Veus também utiliza softwares de código aberto em outras aplicações?***
MB» Usamos soluções abertas no nosso ambiente de desenvolvimento, no dia-a-dia e na produção.

Temos uma aplicação específica de fax que opera em plataforma fechada por força da própria solução. Consideramos que, nesse caso específico, a solução fechada era melhor que a que tínhamos disponível em código aberto.

Os serviços de armazenamento e becape que utilizamos são fornecidos pelo próprio datacenter, sendo integrados ao ambiente Linux.

LM» ***A Veus é um exemplo de que é possível ganhar dinheiro com Software Livre, apesar do crédito de muitos. Como podemos mudar essa idéia equivocada dos gestores e profissionais de TI?***
MB» O mercado de serviços de tecnologia apresenta uma expressiva frente de oportunidades para ser atendida. Software Livre como serviço é uma oportunidade única, acessível a qualquer empresa ou pessoa que tenha criatividade, empenho, perseverança e consiga aliar essas características a uma oportunidade comercial.

Para mudar essa idéia enganada, é necessário informação e orientação. As oportunidades existem e podem ser amplamente trabalhadas. ■

Cezar Taurion

Algumas dicas para empresas que pretender usar o Open Source em seus próprios produtos.

por Cezar Taurion

O ecossistema em torno do Open Source já é maduro o suficiente para impactar a indústria e os usuários de software. Existe um crescente número de soluções de negócio baseadas em Open Source entregando valor real para as empresas. As organizações já olham e implementam softwares Open Source sem os receios de alguns anos atrás.

A IBM, que é um case de sucesso de adoção estratégica de Open Source, vem adotando-o de forma abrangente e pragmática, com forte integração com as comunidades. E nada mais natural do que quase sempre sermos consultados por parceiros ou empresas de software (ISVs) quanto à viabilidade de adotarem Open Source em sua estratégia de negócios.

Abaixo estão algumas dicas de "como ir para Open Source".

Primeiro, estude os conceitos do Open Source. Compreenda os prós e contras, entenda como as comunidades funcionam, estude as diversas alternativas de licenças e analise os projetos de sucesso e, também, os que não deram certo.

Analise a competição das alternativas Open Source no seu segmento de atuação. O resultado da análise é mostrar que, caso você adote Open Source, seu produto terá condições de conquistar uma comunidade de colaboradores com massa crítica o suficiente para que o novo modelo de desenvolvimento seja adotado.

Valide se seu modelo de negócio não conflita com Open Source e quais mudanças serão necessárias. Lembre-se de que Open Source não é um modelo de negócio por si só, mas uma inovação no processo de desenvolvimento, distribuição, marketing e comercialização. Verifique se a mudança em seu modelo de negócio irá gerar mais ou menos receita.

Caso sejam mantidas versões Open Source e proprietárias, defina claramente os limites de funcionalidades de cada uma. Não esqueça que Open Source deve ser atraente o suficiente para atrair usuários e comunidades de colaboradores. Especifique quais funcionalidades premium serão reservadas para a versão proprietária (se existir) e qual o valor percebido dessa funcionalidade para o usuário investir na aquisição de sua licença de uso.

Escolha um nome e um logo adequados, que diferenciem a versão Open da proprietária.

Escolha uma licença adequada ao seu modelo de negócios. Não invente novidades, mas adote uma já existente. E não se esqueça de verificar se o código do seu produto não contém componentes que invalidem a licença escolhida.

Tenha certeza de que o código-fonte a ser liberado está em boas condições. Um código-fonte que ninguém consegue entender vai desestimular a colaboração. Se seu código não estiver adequado, esqueça!

Crie uma infra-estrutura adequada para apoiar a comunidade, com um website, wiki, FAQ, fóruns, planejamento, regras de uso etc. Não confunda a comunidade integrando o site da versão Open Source com a proprietária. Publique o código-fonte em um diretório bem conhecido, como SourceForge.

Seja paciente. Não esqueça que seu software terá que se destacar na multidão, competindo muitas vezes pelos mesmos desenvolvedores que podem já estar envolvidos em outros projetos. Mantenha constante pressão para que a comunidade cresça e se torne atuante. Publicar o software e deixá-lo de lado é torná-lo órfão.

Open Source é um compromisso sério e permanente. Exige muito esforço e comprometimento. É uma decisão estratégica, e não meramente comercial.

E leiam o livro "Producing Open Source Software", de Karl Fogel, disponível gratuitamente em http://producingoss.com. Boa sorte! ■

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.

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.

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

POWERED BY: virtual target

Examinando a arte da computação forense

Pega ladrão!

Não é preciso ter caras ferramentas proprietárias para praticar a arte da computação forense.

por Nils Magnus, Achim Leitner e Joe Casad

Cena do crime: a sala dos servidores... O ladrão não precisa de um cartão para entrar nela, nem mesmo da proteção das sombras – o invasor pode usar a Internet para ir e vir. Porém, apesar da entrada secreta, o agressor ainda deixa vestígios que podem desmascará-lo. Encontrar e interpretar essas evidências é a maior prioridade dos investigadores criminais.

O tema de capa deste mês explora o mundo da computação forense. Mostraremos algumas ferramentas usadas pelos especialistas para encontrar pistas, recuperar arquivos apagados e desenterrar provas escondidas. Começaremos com um estudo das ferramentas forenses do *Sleuth Kit*. Em seguida, mostraremos o *Foremost* e o *Scalpel*, duas ferramentas para encontrar e recuperar arquivos deletados. Ensinaremos ainda a examinar discos de sistemas Windows com ferramentas do Linux, abordando, ao fim, a *Open Computer Forensics Architecture*, uma coleção gratuita de bibliotecas e ferramentas forenses desenvolvidas pela polícia holandesa.

Índice das matérias de capa

Porém, se você não vai enfrentar um julgamento e deseja apenas capturar o invasor do sistema, talvez não seja necessário realizar uma perícia completa. As seções a seguir descrevem algumas ferramentas para encontrar invasores no sistema usando utilitários padrão do Linux.

Bem debaixo do seu nariz?

Uma das primeiras perguntas que um investigador forense deve perguntar é se a investigação deve ser realizada abertamente – o que significa que também será visível para o agressor – ou se o invasor não deve saber que está sendo investigado.

Um computador sob investigação forense é bem semelhante a uma partícula na mecânica quântica: simplesmente olhar para ela já altera seu estado.

Um agressor poderia ver o comando `ps` e, rodando o `find` no disco rígido, sobrescrever os valores de *atime* dos objetos do sistema de arquivos, eliminando provas do último acesso de um usuário.

Apesar das possíveis complicações de se trabalhar abertamente, a necessidade de chegar à raiz de atividades ilícitas às vezes é mais importante do que usar técnicas elaboradas para evitar ser notado.

Além disso, lembre-se de que a maioria dos ataques são disparados por meio de scripts e programas automatizados e que, portanto, não é comum capturar um agressor em flagrante no console. As dicas a seguir são destinadas principalmente a casos em que não seja fundamental esconder suas atividades ou deixar uma trilha de papel.

Para evitar dar pouca atenção aos detalhes, uma abordagem sistemática é muito útil. A idéia de seguir um rastro ainda quente é sempre muito sedutora, mas se não levar o investigador a algum lugar, será decepcionante.

Por exemplo, se você investigar uma lista de processos com o comando:

```
ps gauxwww
```

você pode guardar a lista primeiro e consultá-la depois. O comando exibe todos os processos ativos e seus argumentos de linhad e comando, com todas as opções usadas.

Obviamente, se o sistema em questão tiver sido comprometido, o invasor poderá ter instalado versões alteradas (com cavalos de tróia, por exemplo) dos utilitários do sistema, como o próprio `ps`, para esconder seus atos.

Um pequeno script de shell pode fazer o mesmo lendo dados do diretório `/proc/`.

Extensões individuais podem ser facilmente adicionadas a um script como esse e podem ser particularmente úteis se o `ps` não for mais confiável.

Para um bom teste de sanidade, é necessário verificar os resultados usando

uma ferramenta semelhante ao popular `pstree`. Investigadores forenses também lembrarão que os programas podem mudar a lista de argumentos.

Kernel

Um macete simples, como procurar processos, é inútil contra um rootkit de kernel. Os rootkits modificam o kernel para impedir que ele forneça informações sobre certos processos ao sistema `/proc/` ou outros semelhantes.

Por outro lado, é muito surpreendente como alguns invasores não se dignam a cobrir seus rastros, então pode valer a pena tentar isso.

Conexões de rede

Além dos processos, conexões de rede também podem revelar pistas, tais como o vetor de ataque e o endereço usado pelo invasor para se conectar ao sistema.

O comando `netstat --ip -pan` exibe todos os soquetes IP locais, seus protocolos (TCP ou UDP) e possivelmente os parceiros de comunicação dos soquetes conectados – a menos que o comando ou o kernel tenham sido manipulados.

Usar a opção `-n` no `netstat` impede que o DNS resolva os endereços IP, o que é uma boa idéia para evitar tráfego desnecessário na rede, pois isto levantaria suspeitas no invasor. Se for realmente necessário, sempre se pode resolver os IPs em outro momento.

Os comandos `whois` e `traceroute` exibem mais informações sobre endereços IP, as quais dificilmente podem ser forjadas pelo agressor sem a colaboração de um provedor de acesso.

Origem da conexão

Um último fator que não deve escapar do investigador forense é que a origem das conexões TCP e UDP pode ser diferente da localização real do invasor. Alguns agressores utilizam sistemas "seqüestrados" como ponto de partida para seus ataques.

Se a conexão for originada num sistema muito próximo ao invadido, é importante tomar cuidado. Uma lista mais curta de pulos na saída do `tcpdump <destino>` será mais informativa. Se for possível eliminar as suspeitas de que se trata de um usuário comum, já haverá provas suficientemente convincentes de que um agressor remoto entrou pela rede. Um invasor próximo é muito mais perigoso do que um mais distante, pois capturar senhas na mesma sub-rede é bem mais fácil do que pela Internet).

Procurando pistas

Se o especialista forense descobrir um processo ou programa desconhecido em execução no sistema, a próxima pergunta será: "o que ele faz?".

Ele pode estar apenas usando a sua máquina como intermediário para mais ataques, o que significa que um processo desconhecido deve criar uma entrada na lista de soquetes abertos.

Ou então, ele pode estar capturando dados na rede. Se for esse o caso, certamente haverá uma interface de rede em modo promíscuo, que pode ser detectada com o comando `dmesg`, por exemplo.

Mas o que se deve fazer ao ver um processo ativo que faça algo desconhecido? Um bom primeiro passo seria fazer um becape do próprio executável responsável pelo processo, o que é fácil com o comando `ps gauxwww` ou consultando-se o arquivo `/proc/PID/cmdline`.

O que se pode fazer caso o agressor tenha iniciado uma ferramenta, imediatamente apagado-a e sobrescrito os setores do disco? Enquanto o programa estiver em execução, há esperança – o kernel mantém um link simbólico virtual para o executável em `/proc/PID/exe`, mesmo que o agressor o tenha apagado do sistema de arquivos. Se a equipe de recuperação salvar esse arquivo em algum local, provavelmente será possível analisá-lo em outro momento.

Lixo binário

Uma técnica simples, mas efetiva, é analisar o próprio binário. O comando `strings -a binário` procura caracteres imprimíveis no arquivo. Caso o programa malicioso se conecte a um servidor FTP ou Web que exija uma senha, pode ser possível encontrar a senha no código do programa. Porém, será necessária certa intuição para fazer a distinção entre as migalhas digitais e o lixo binário.

Conclusão

As estratégias simples descritas nesta introdução podem ajudar um administrador a flagrar o gatuno. Porém, se o intruso for um "profissional" experiente, ou se for necessário manter um processo formal e documentado para coleta de provas, então precisaremos de algo mais.

As matérias a seguir exploram técnicas mais avançadas para quem desejar se aprofundar na área. ■

Análise forense com BackTrack e Sleuth Kit

Rastreamento

Após determinar que um sistema foi atacado, use o Live CD do BackTrack e comece as investigações com o Sleuth kit.

por Kurt Seifried

O cibercrime é um problema sério – em grande parte porque praticamente todas as informações corporativas hoje são gerenciadas por computadores e não mais com ferramentas tradicionais como papel e pessoas. Computadores e redes constituem um suculento alvo para invasores e, dependendo do seu desejo, um ataque pode ir desde o irritante até o catastrófico. Como quase todas as informações da empresa estão em computadores, qualquer um que acesse essas informações com intenções criminosas provavelmente deixará pistas.

O que os ataques têm em comum é que, quando se percebe a ocorrência de um incidente, geralmente não se tem todas as informações necessárias para tratá-lo. Agrupar os fatos pode exigir uma investigação forense. O ataque pode ter sido desferido de dentro da rede da empresa ou pode ter explorado uma falha disponível externamente. O agressor pode ter acessado um único sistema ou talvez a rede inteira. Ele pode ter roubado dados, plantado um vírus ou até mesmo instalado um rootkit.

A distribuição em Live CD *BackTrack*[1] e o kit de ferramentas forenses *Sleuth Kit*[2] ajudam a coletar informações a respeito do ataque. Este artigo mostrará como usar o BackTrack e o Sleuth Kit, mas, primeiro, vamos começar com alguns passos iniciais antes de iniciar a análise forense.

A eletrônica forense é um tópico enorme que mesmo ao restringi-la a algumas ferramentas para sistemas Linux ainda nos deixaria com informações demais a cobrir. Este artigo, portanto, parte do princípio de que:

- você já sabe quais sistemas provavelmente foram comprometidos (as ferramentas de detecção de ataques, como *Snort* e *Tripwire* não serão cobertas);
- você não vai recorrer à lei – há questões demais envolvidas com a jurisprudência, coleta de provas e corrente de custódia nesse campo (**quadro 1**);

Quadro 1: No tribunal

Não sou advogado e este não é um conselho legal, porém sei que em algumas jurisprudência é permitido obter provas dentro de uma organização sem a necessidade de mandados de busca. Se o administrador decidir envolver a polícia, pode ser considerado como um agente da polícia e, portanto, precisar de um mandado de busca para qualquer outra descoberta e análise. Além disso, as regras de coleta de provas, corrente de custódia e ferramentas aceitáveis variam de acordo com a jurisdição. Se a polícia for contactada a qualquer momento, é importante consultar um advogado para auxiliar nos procedimentos necessários.

- é possível parar os sistemas afetados para gerar uma imagem de seus discos; e
- existem procedimentos de becape e recuperação em uso.

Apesar de o foco deste artigo ser o Linux, as ferramentas cobertas podem ser usadas para examinar outros sistemas, como Unix e Windows.

On/Off

Uma decisão importante a tomar é desligar ou não o sistema quando se souber ou suspeitar de que ele tenha sido comprometido. Caso se decida desligá-lo, como isso será feito? Da maneira comum ou simplesmente puxando o fio da tomada? O exame forense de um sistema ligado tem várias vantagens. É possível consultar a tabela de processos, listar conexões de rede e copiar o conteúdo da memória para posterior análise.

Por outro lado, há grandes desvantagens na análise de sistemas ligados, incluindo o fato de que o que se vê não é, necessariamente, o que está lá. Rootkits modernos podem facilmente ocultar processos e dados, por exemplo, inserindo ganchos (*hooks*) no kernel. Um sistema desligado é mais fácil de analisar e permite a certeza de que, após ser desligado, nada foi alterado ou apagado a partir do estado em que ele se encontrava.

Mas, como desligar o sistema? Um desligamento normal poderia acionar programas que fazem uma faxina no rastro do agressor e apagam as provas ou que, caso o invasor seja especialmente malicioso, sobrescrevem o firmware do disco rígido ou do sistema. Todavia, simplesmente puxar o plugue pode deixar o sistema num estado inconsistente ou impedir que dados sejam gravados no disco.

É fundamental examinar as questões com atenção – a melhor escolha da forma de desligamento do sistema depende da informação que se deseja coletar e do que se pretende fazer com elas.

Quadro 2: Bloqueador de escrita por hardware

Para fazer análises forenses sérias, é interessante investir num bloqueador de escrita por hardware (*hardware write blocker*). Segundo a Forensic Wiki [3], o bloqueador permite "a aquisição de informações de um volume sem criar a possibilidade de danificar seu conteúdo acidentalmente. Ele faz isso permitindo que comandos de leitura passem, mas bloqueando os comandos de escrita".

Geralmente, um bloqueador de escrita custa de 100 a 300 dólares, enquanto que um kit completo pode custar entre mil e dois mil. Entretanto, o custo de modificar ou apagar acidentalmente as provas deve ser considerado frente ao preço do dispositivo. Além disso, a falta de um bloqueador de escrita também pode ser suficiente para levantar dúvidas no tribunal.

Linux forense

O processo de coleta e análise de evidências em sistemas Linux segue um padrão geral:

1. Desligar o sistema afetado.
2. Gravar uma imagem do(s) disco(s) rígido(s).
3. Examinar a imagem com ferramentas como o Sleuth Kit.
4. Processar as provas e informações para chegar a uma conclusão.

As seções a seguir abordam esse processo em maior profundidade.

Desligamento

Recomenda-se o desligamento normal do sistema, se possível; porém, se houver alguma suspeita de que o agressor tenha deixado bombas lógicas ou scripts de faxina prontos para agir, pode ser mais adequado simplesmente puxar o fio. A vantagem de desligar o sistema é que se consegue reiniciá-lo a partir de alguma mídia confiável, como um CD de recuperação ou um de análise forense como o *BackTrack*, e então criar uma imagem do disco. Quando se faz uma imagem de um sistema ligado, os rootkits podem esconder informações importantes para a solução do problema.

Acesso ao disco

Em relação ao acesso aos discos afetados, novamente é preciso fazer uma escolha: pode-se deixá-los na máquina e iniciá-la por um CD ou pendrive, ou então remover os discos e conectá-los a outro sistema para criar sua imagem.

Se for preferível deixar os discos no lugar, é imperativo garantir que a BIOS esteja configurada para iniciar a partir das mídias alternativas e não do próprio disco.

Além disso, também é importante cuidar para não gravar nada no disco, por exemplo, por um engano na criação da imagem (veja os **quadros 2 e 3**).

BackTrack

Após desligar o sistema, uma alternativa popular é começar a investigação com um Live CD de Linux.

Assim como em várias outras áreas do Software Livre, o enorme número de escolhas de distribuições para

Quadro 3: Cuidado com o fstab

Ao conectar discos a um sistema para analisá-los, é importante saber como serão tratados pelo `/etc/fstab`. Durante a inicialização, o Linux varre todos os discos em busca de rótulos. Se ele encontrar um mesmo rótulo em partições diferentes, ele montará somente a última encontrada, o que pode causar a corrupção de provas.

esse fim é ao mesmo tempo bom e ruim. Quase todas as mídias de instalação (Red Hat, Debian etc.) possuem um modo de recuperação ou de emergência que pode ser usado para acessar o sistema. Distribuições *Live* para CDs ou pendrives também são uma possibilidade.

O BackTrack[1], uma das baseadas em Live CDs, tem algumas vantagens sobre as demais:

- suporte a múltiplos tipos de sistemas de arquivos, incluindo *Ext2*, *Ext3*, *VFAT*, *NTFS* e outros;
- está sob desenvolvimento ativo e é especificamente voltado a testes de penetração;
- inclui utilitários para análise forense, como o *dcfldd*[4], uma ferramenta avançada de cópia de discos.

Para baixar o BackTrack, é necessário um cliente BitTorrent, pois essa é a única forma de distribuição da imagem ISO da distribuição. Este artigo foi escrito com base na versão beta mais recente, que foi usada sem qualquer problema.

Outra possibilidade é baixar uma versão USB do sistema e gravá-la num pendrive.

Criação da imagem

Criar uma imagem completa de uma partição ou disco é relativamente fácil no Linux. O comando `dd`, incluído no BackTrack, é capaz de criar imagens de partições ou discos completos.

Para enviar o conteúdo das imagens a outros sistemas e reduzir a necessidade de desmontagem de máquinas para remoção de seus discos, é possível combinar o `dd` e ferramentas de rede como *netcat* ou *SSH*:

```
dd if=/dev/hda1 bs=2k | nc\
192.168.0.1 9000
```

Esse comando criará uma imagem da primeira partição do primeiro disco (`hda1`) e enviará os dados para a porta TCP 9000 na máquina 192.168.0.1. Esta, por sua vez, precisará estar escutando nessa porta para armazenar a imagem:

```
nc -l 9000 > imagem.dd
```

O `dcfldd` inclui várias funcionalidades úteis para a análise forense digital. Diferentemente do `dd`, ele pode criar *hashes* MD5 e SHA256 dos dados, facilitando a verificação dos dados e também a capacidade de dividir arquivos em múltiplos pedaços de tamanho definido.

O comando a seguir cria uma imagem de `hda1`, um hash MD5 e um SHA256 de cada bloco de 10 GB de dados (gravando-os num log), continua lendo caso encontre erros e preenche blocos de entrada caso seja necessário. Por último, o comando divide os dados em arquivos de 10 GB com nomes terminando em `aa`, `ab` e assim por diante:

```
dcfldd if=/dev/hda1 \
hash=md5,sha256 \
hashwindow=10G \
md5log=md5.txt \
sha256log=sha256.txt \
hashconv=after bs=512 \
conv=noerror,sync \
split=10G splitformat=aa \
of=driveimage.dd
```

É importante lembrar que, exceto se os dados são enviados por uma rede segura, é preciso provar que não tenham sido alterados no meio do caminho, o que pode ser feito com hashes criptográficos e com cópias seguras fora do sistema por meio de pendrives, por exemplo, ou criptografando os dados em trânsito com uma ferramenta como *OpenSSH* para criar um túnel.

Sleuth Kit e Autopsy

O *Sleuth Kit* é uma útil coleção de ferramentas forenses de código aberto, como o `mmstat`, que exibe informações sobre tabelas de partição, e o `jls`, que lista o conteúdo do *journal* do sistema de arquivos.

O procedimento padrão numa investigação com o Sleuth Kit é:

- com o `fls`, criar uma lista de arquivos e diretórios críticos presentes na imagem;
- com o `ils`, criar uma lista de informações dos inodes;
- com o `mactime`, criar uma linha do tempo (atividade do arquivo, acesso, apagamento etc.);
- com o `icat`, extrair arquivos interessantes (e apagados) dos inodes.

Um exemplo dos primeiros passos a serem seguidos é:

```
# fls -f ext -m / \
/provas/imagem.dd > dados_saida

# ils -f ext -m \
/provas/imagem.dd >> saida_dados

# mactime -b saida_dados\
01/01/2008-12/31/2008 >\
relatorio-de-atividade-2008
```

Se o agressor alterou as horas de acesso, será preciso especificar uma ampla faixa de datas para garantir a obtenção de todos os dados. A execução desse código deve resultar

Exemplo 1: Rastreamento de acesso

```
01 Mon Jun 02 2008 01:16:45 24 ..c -/-rw-r--r- kurt kurt 58498 /
➥home/kurt/.bash_logout
02 176 ..c -/-rw-r--r- kurt kurt 58499 /home/kurt/.bash_profile
03 124 ..c -/-rw-r--r-- kurt kurt 58500 /home/kurt/.bashrc
```

numa saída semelhante ao **exemplo 1**, em que se pode ver que um usuário chamado *Kurt* acessou uma conta por SSH.

Extração de arquivos com Icat

O *Icat* é um utilitário relativamente simples que encontra um inode num arquivo de imagem e copia os dados para um arquivo. Seu comando `icat` inclui várias opções úteis, como `-s`, que copia o *slack space* (espaço vazio entre o fim do arquivo e o fim do bloco físico), que pode conter informações ocultas ou interessantes e também `-r`, que recupera arquivos deletados. Por exemplo, o comando a seguir exibe o conteúdo de `/home/kurt/.bash_profile` (veja também o **exemplo 2**):

```
icat -s -f ext imagem.dd 58499
```

Exemplo 2: /home/kurt/.bash_profile

```
01 # .bash_profile
02
03 # Aliases e funcoes
04 if [ -f ~/.bashrc ]; then
05 . ~/.bashrc
06 fi
07
08 # Programas especificos do usuario
09
10 PATH=$PATH:$HOME/bin
11
12 export PATH
13
14 autopsy - a web interface to Sleuth Kit
```

Autopsy

Embora a curva de aprendizado do Sleuth Kit não seja muito íngreme, é fácil cometer um erro que custaria muito tempo e trabalho para consertar. O navegador forense *Autopsy*, disponível no site do Sleuth Kit[2], automatiza o processo e oferece uma interface web. Ele possui também outros recursos, como acompanhamento de casos, tratamento de notas e eventos e suporte a múltiplos usuários. Por padrão, o Autopsy permite somente conexões do localhost ao servidor web.

Para permitir a conexão de IPs remotos, é preciso usar a opção `-c`; porém, é importante lembrar que o Autopsy não oferece criptografia, então, se não for acessado localmente, é preciso conectar-se por uma rede confiável ou usar algo como o OpenSSH para criar um túnel seguro.

Tipos de arquivo

Há várias opções na tela de análise de imagem do Autopsy. A mais interessante em geral é a de *File Type*, mas é bom aguardar um pouco antes de mandá-lo ordenar a lista pelo tipo de arquivo (*Sort Files by Type*).

Esse recurso varre o arquivo de imagem inteiro, extrai os arquivos, ordena-os em várias categorias (imagens, documentos, executáveis, criptográficos etc.) e dá a opção de copiá-los para posterior análise. O **exemplo 3** ilustra a saída relacionada a arquivos criptográficos.

Exemplo 3: Saída de arquivo criptográfico

```
01 /home/secret/.pgp/secring.pgp
02 PGP key security ring
03 Image: /evidence/ddriveimage.dd Inode: 672945
04 Saved to: crypto/ddriveimage.dd-672945
05
06 /home/secret/.pgp/pubring.pgp
07 PGP key public ring
08 Image: /evidence/ddriveimage.dd Inode: 672959
09 Saved to: crypto/ddriveimage.dd-672959.pgp
```

Palavras-chave

Outro benefício do Autopsy é a tela de busca por palavras-chave. Ela permite o uso de expressões regulares e facilita muito o trabalho com várias buscas pré-configuradas, como números de cartão de crédito, de seguro social norte-americano, de IPs e datas. Os resultados das buscas são armazenados temporariamente, o que faz com que buscas já realizadas retornem resultados quase instantaneamente.

Conclusão

O Sleuth Kit oferece um conjunto incrivelmente poderoso – e gratuito – de utilitários para eletrônica forense e funciona em sistemas Linux, Windows e outras variantes de Unix. Com a ajuda da interface web Autopsy, a solução se torna extremamente fácil de usar, além de poder ajudar a apresentar novos usuários ao convidativo universo das análises forenses.

O Sleuth Kit definitivamente merece um lugar nas prateleiras de utilidades dos administradores e auditores de sistemas. ■

Mais informações

[1] BackTrack: http://www.remote-exploit.org/backtrack.html

[2] Sleuth Kit: http://www.sleuthkit.org/

[3] Bloqueadores de escrita: http://www.forensicswiki.org/wiki/Write_Blockers

[4] dcfldd: http://dcfldd.sourceforge.net/

[5] Linux LEO: http://www.linuxleo.com/

Ferramentas para restaurar arquivos apagados

Desapagado

Sistemas de arquivos modernos dificultam muito a recuperação forense de arquivos. O Foremost e o Scalpel são capazes de encontrá-los e recuperá-los.

por Ralf Spenneberg

Investigadores de TI têm muitos motivos para reconstituir arquivos deletados. Seja por um intruso ter apagado um log para ocultar um ataque ou por um usuário ter simplesmente destruído sua coleção de fotos digitais com um `rm -rf` acidental, é possível ter que enfrentar a necessidade de recuperar dados apagados. No passado, os especialistas em recuperação conseguiam restaurar arquivos perdidos com muita facilidade, pois os sistemas de arquivos da época simplesmente apagavam sua entrada no diretório. As metainformações dos dados do disco eram preservadas, o que possibilitava a várias ferramentas descobrirem as informações necessárias para recuperar o arquivo.

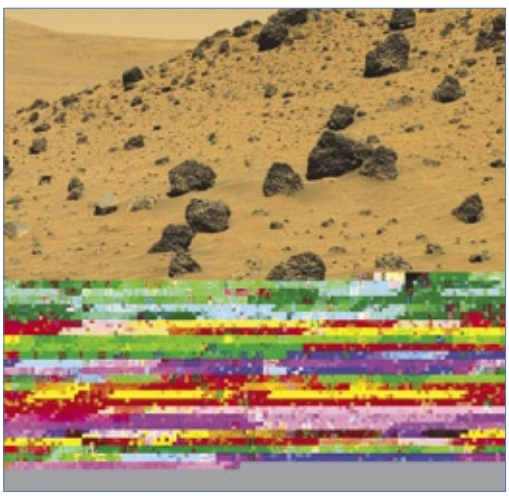


Figura 1 Recortadores de arquivos ignoram o sistema de arquivos e recuperam imagens diretamente de blocos de dados. Porém, fotos em arquivos fragmentados podem ficar imperfeitas.

Hoje em dia, vários sistemas de arquivos apagam todas as metainformações, mas deixam os blocos de dados. O ato de reunir essas duas informações é chamado de *file carving* (aproximadamente "recorte de arquivos") – os especialistas recortam os dados crus do disco e reconstroem os arquivos a partir deles. Quanto mais fragmentado estiver o sistema de arquivos, mais difícil fica a tarefa.

Várias ferramentas de código aberto automatizam o processo de recorte: a lista é liderada pelo *Foremost*[1] e seu descendente *Scalpel*[2], mas há outros, como *PhotoRec*[3] e *FTimes*[4]. O PhotoRec não suporta o carving de todos os arquivos e o FTimes é tão difícil de usar que não vale a pena para a maioria dos usuários.

O Foremost e o Scalpel não se preocupam com o sistema de arquivos subjacente. Eles simplesmente esperam que os blocos de dados dos arquivos residam seqüencialmente na imagem investigada. Esses programas encontram arquivos em imagens do `dd`, *dumps* da memória e arquivos de *swap*. O carving ajuda a identificar e reconstruir arquivos em sistemas de arquivos corrompidos, no *slack space* ou até mesmo após a instalação de um novo sistema operacional, contanto que os blocos de dados necessários ainda existam.

Obviamente nenhuma dessas ferramentas consegue fazer milagres, além de não terem sido projetadas para obter dados de discos rígidos danificados fisicamente. Além disso, o processo de carving não é capaz de acessar blocos de dados que tenham sido sobrescritos.

Como as ferramentas de carving não dependem do sistema de arquivos, elas precisam de outras fontes de informação para descobrir onde começa e termina cada arquivo. Felizmente, vários tipos de arquivos possuem estruturas conhecidas. Seu cabeçalho e seu final freqüentemente bastam para identificarmos o tipo do arquivo e sua localização. O comando `file`, por exemplo, também usa as informações de cabeçalho e final dos arquivos para identificar seus tipos.

As ferramentas de carving de arquivos analisam o disco rígido (ou imagem) inteiro para localizar cabeçalhos e finais de arquivos conhecidos. Depois, eles recortam os blocos entre o cabeçalho e o final e armazenam os dados como um novo arquivo.

Exemplo 1: Arquivo de configuração

```
01 gif y 155000000 \x47\x49\x46\x38\x37\x61  \x00\x3b
02 gif y 155000000 \x47\x49\x46\x38\x39\x61  \x00\x00\x3b
03 jpg y 20000000  \xff\xd8\xff\xe0\x00\x10  \xff\xd9
04 jpg y 20000000  \xff\xd8\xff\xe1\xff\xd9
05 jpg y 20000000  \xff\xd8               \xff\xd9
```

Alguns tipos de arquivos não possuem finais únicos. As ferramentas de carving tentarão ao menos adivinhar onde o arquivo termina com base no cabeçalho do próximo arquivo. No entanto, é claro que entre essas duas estruturas pode haver qualquer quantidade de dados não identificados.

Para evitar a coleta de dados desnecessários, os programas de carving permitem que os usuários especifiquem tamanhos máximos de arquivos. Infelizmente, cabeçalhos e finais costumam ser pequenos, o que leva a vários falsos positivos.

Os formatos de imagens são exceções. Por exemplo, todo arquivo *JPEG* inicia com uma seqüência de bytes de *oxFFd8*, geralmente seguida de *oxFFE0o010*. As ferramentas de carving, portanto, são muito competentes ao identificar esses arquivos. Contudo, se alguns blocos tiverem sido sobrescritos ou se o arquivo estiver fragmentado, as ferramentas recuperarão somente uma parte do arquivo, na melhor das hipóteses (**figura 1**).

As ferramentas

Jesse Kornblum e Kris Kendall, do escritório de investigações especiais da força aérea dos EUA, desenvolveram o Foremost em março de 2001 como uma ferramenta para analisar e recuperar arquivos deletados. O programa é inspirado em outro anterior, chamado *CarvThis*, criado em 1999 pelo Defense Computer Forensic Lab, mas jamais liberado para o público geral. Agora, o Foremost tem seu código aberto e Nick Mikus mantém o código-fonte após aumentar bastante o escopo do programa em seu mestrado.

Golden G. Richard III desenvolveu um programa separado, chamado Scalpel, com base no Foremost 0.69. Durante muito tempo o Scalpel foi considerado uma ferramenta avançada. Alguns até alegam que os próprios desenvolvedores do Foremost recomendam o Scalpel[2].

Os dois projetos estão em desenvolvimento ativo. Embora o Scalpel fosse muito superior a seu antecessor em 2005 – com a capacidade de analisar imagens dez vezes mais rápido –, o Foremost voltou a ganhar popularidade recentemente, graças a Nick Mikus, e hoje é melhor que seu "filho" para algumas tarefas.

Ambas as ferramentas usam arquivos de configuração para especificar quais arquivos devem procurar (**exemplo 1**). A primeira coluna define o tipo de arquivo e também especifica a extensão de arquivo a adicionar a todos os arquivos que o programa en-

Tabela 1: Buscas embutidas no Foremost

Formato	Descrição
Imagens	
JPG	Formatos JFIF, Exif e RAW
GIF	Formato GIF
PNG	Formato PNG
BMP	Bitmaps do Windows
Executáveis	
EXE	PE, DLL e EXE do Windows
Vídeo e Áudio	
AVI	Arquivos AVI
MPG	Detecta qualquer arquivo MPEG iniciado com `0x000001BA`
WMV	WMV e, em parte, WMA
MOV	Vídeos Quicktime
Documentos	
PDF	Arquivos PDF
OLE	Arquivos PowerPoint, Word, Excel, Access, Starwriter
DOC	Somente arquivos Word
HTM	Arquivos HTML
Compactados	
ZIP	Arquivos ZIP, JAR e XML zipado, como ODF e OOXML
RAR	Arquivos RAR
CPP	Código-fonte C; muitos falsos positivos

contrar. Os arquivos sensíveis à caixa no cabeçalho e no final possuem um `y` na segunda coluna; os outros têm um `n`. A coluna seguinte define o tamanho máximo de arquivo, seguido da seqüência de bytes do cabeçalho e também pela seqüência dos bytes do final, caso exista. O caractere `\w` introduz um byte na notação hexadecimal; as outras possibilidades são `\s` para um espaço e `?` como curinga. Há outras opções no fim.

Exemplo 2: Execução do Foremost

```
01 Foremost version 1.5.3 by Jesse Kornblum, Kris Kendall,
➥and Nick Mikus
02 Audit File
03
04 Foremost started at Sat Feb 9 18:36:29 2008
05 Invocation: ./foremost -v -T -i ../dfrws-2006-challenge.raw
06 Output directory: /linux-magazine/foremost/foremost-1.5.3/output_
➥Sat_Feb__9_18_36_29_2008
07 Configuration file: /linux-magazine/foremost/foremost-1.5.3/
➥foremost.conf
08 Processing: ../dfrws-2006-challenge.raw
09 |------------------------------------------------
10 File: ../dfrws-2006-challenge.raw
11 Start: Sat Feb 9 18:36:29 2008
12 Length: 47 MB (49999872 bytes)
13
14 Num    Name (bs=512)    Size    File Offset  Comment
15
16 0:   00003868.jpg   280 KB      1980416
17 1:   00008285.jpg   594 KB      4241920
18 2:   00011619.jpg   199 KB      5948928
19 3:   00012222.jpg     6 MB      6257664
20 [...]
21 20:  00045015.zip   274 KB     23047680
22 21:  00007982.png     6 KB      4086865  (1408 x 1800)
23 22:  00033012.png    69 KB     16902215   (1052 x 360)
24 23:  00035391.png    19 KB     18120696    (879 x 499)
25 24:  00035431.png    72 KB     18140936   (1140 x 540)
26 *|
27 Finish: Sat Feb 9 18:36:32 2008
28
29 25 FILES EXTRACTED
30
31 jpg:= 11
32 htm:= 5
33 ole:= 2
34 zip:= 3
35 png:= 4
36 ----------------------------------------------------
37
38 Foremost finished at Sat Feb 9 18:36:32 2008
```

Acha correndo

Em virtude de suas origens, o Scalpel usa o mesmo arquivo de configuração que o Foremost, embora as duas ferramentas funcionem de forma diferente internamente. Ambas encontram os mesmos arquivos, mas há diferenças na identificação de arquivos. Portanto, especialistas forenses devem usar sempre as duas ferramentas.

As versões 0.9.1 e posteriores do Foremost usam uma nova técnica de identificação de arquivos ZIP, JPEG, Office e outros formatos. Esses formatos são implementados diretamente no programa, o que significa que não há necessidade de definir informações de cabeçalho e final no arquivo de configuração. A opção `-t` ativa o uso dessa nova função de detecção, seguida dos formatos de arquivos em que ela deve ser aplicada:

```
foremost -T -t jpg,pdf -i arquivo
```

Os formatos suportados são listados na **tabela 1**. Para ativar a nova busca para todos, basta usar `-t all`. A opção `-T`, usada na linha de comando acima, faz o programa gravar em um diretório específico todos os arquivos encontrados. Isso facilita a organização da investigação forense, pois cada nova execução resulta num novo diretório.

Espaço

A possibilidade de falsos positivos significa que o "recortador" identifica um grande volume de dados, então, é preciso ter espaço livre suficiente no sistema de destino. O processo de recorte não precisa necessariamente de grande volume de cópias.

Sistemas de arquivos virtuais, como o *CarvFS*[5], são projetados para acessar os dados diretamente a partir da imagem original. O CarvFS se baseia no *FUSE* e precisa apenas que a ferramenta de carving forneça uma tabela

para descrever quais arquivos estão disponíveis em quais locais físicos.

O sistema de arquivos CarvFS foi desenvolvido pela polícia holandesa sob a *Open Computer Forensics Architecture* (veja o artigo "Da terra dos moinhos") para situações em que copiar todos os arquivos para um local separado resultaria em enormes volumes de dados. Porém, em outros casos, copiar os dados é mais eficiente do que acessá-los a partir da imagem original.

Uma execução típica do Foremost sem sua busca embutida é mostrada no **exemplo 2**. A imagem desse exemplo é uma cortesia do desafio do Workshop de Pesquisa Forense Digital (DFRWS[6]).

PhotoRec

Se o sistema de arquivos não estiver completamente destruído, ferramentas que o avaliam oferecem uma importante alternativa aos objetos deste artigo. A ferramenta PhotoRec[3] foi desenvolvida por Christophe Grenier para recuperar fotos em memórias *Flash* corrompidas. O programa também funciona se a tabela de partições estiver danificada.

Após identificar o sistema de arquivos, o PhotoRec extrai uma grande variedade de tipos de arquivos. Além de fotos, ele também recupera executáveis de sistemas Windows (*EXE*) e arquivos *ZIP*.

No total, a ferramenta suporta mais de 180 tipos de arquivos. O programa é controlado por meio de um prático menu de texto, que reduz o perigo de erros do usuário. Infelizmente, o PhotoRec não é capaz de analisar dumps de RAM e arquivos de swap.

Conclusão

Os softwares de carving de arquivos ajudam investigadores forenses a extrair arquivos deletados. O Foremost e o Scalpel ignoram o sistema de arquivos e demonstram ótima velocidade.

Se o sistema de arquivos ainda existir, no entanto, uma ferramenta como o PhotoRec pode ser muito útil para encontrar arquivos perdidos. ■

Mais informações

[1] Foremost: http://foremost.sourceforge.net

[2] Scalpel: http://www.digitalforensicssolutions.com/Scalpel/

[3] PhotoRec: http://www.cgsecurity.org/wiki/PhotoRec

[4] FTimes: http://ftimes.sourceforge.net/FTimes/

[5] CarvFS: http://ocfa.sourceforge.net/libcarvpath/

[6] DFRWS: http://www.dfrws.org/2006/challenge/

CAPA

Investigação de sistemas Windows com o Linux

Conserte as janelas

Um especialista forense explica como extrair detalhes interessantes de um disco Windows com ferramentas padrão do Linux.

por Hans-Peter Merkel e Markus Feilner

Criminosos, invasores e sabotadores corporativos deixam rastros de dados em qualquer computador que visitem. Muitos desses computadores são sistemas Windows, mas felizmente é possível usar sistemas Linux para extrair informações forenses valiosas de seus discos. Este artigo descreve algumas técnicas para obter dados forenses de um disco Windows usando o Linux.

Antes de iniciar qualquer análise forense, é importante criar uma cópia da mídia de armazenamento a ser investigada, seja como uma imagem exata ou com um grupo de imagens. Pode-se copiar a mídia como imagem crua com o `dd` ou usar um formato como o *Expert Witness Format* (EWF).

O EWF é um formato proprietário desenvolvido pela Guidance Software[1] que também é suportado pela ferramenta forense comercial *X-Ways*[2]. Como as imagens EWF são comprimidas, ficam muito menores que imagens cruas.

As ferramentas para Linux, como o *Linux Encase* (Linen) ou o *Ewfacquire*[3], podem ajudar a criar uma imagem EWF. O Linen está incluído no CD de análises forenses *Helix*[4] como contribuição gratuita da Guidance Software, mas a ferramenta `dd`, incluída em qualquer distribuição Linux, já é plenamente capaz de satisfazer as necessidades. Se ela for usada, é possível até mesmo rodar uma cópia do sistema Windows num ambiente virtual como o VMware; o EWF, por outro lado, só permite isso com um software adicional (e proprietário), justamente por causa de seu formato compactado.

Um comando como:

```
dd if=/dev/sda of=/win.dd
➥ bs=4096
➥ conv=noerror,sync
```

criará uma imagem crua do disco Windows. O parâmetro `conv` garante que a cópia não termine em caso de erro (setores defeituosos, por exemplo).

Em seguida, o comando `fdisk -lu win.dd` obtém as informações sobre a imagem do disco (**exemplo 1**).

No **exemplo 1**, a imagem contém uma partição, cujo sistema de arquivos é *NTFS*. O programa `disktype` dos repositórios padrão do Debian oferece mais informações (**exemplo 2**). A partição obviamente contém o carregador de inicialização do Windows.

Para acessar o sistema de arquivos, o administrador primeiro precisa montá-lo. A partição começa no setor 63, o que é comum em discos rígidos. A única exceção é o Windows Vista, cujo primeiro setor é o 2047. O comando `losetup`, então, especifica o deslocamento:

```
# losetup -o $((63*512)) /dev/
➥loop0 U win.dd
# mount -o ro,noatime,noexec /dev/
➥loop0 /mnt
```

Uma rápida análise do `/mnt/` revela os arquivos de inicialização e do sistema de arquivos do disco Windows. O arquivo `boot.ini` mostra que pertencem a um sistema Windows 2000 Server (**exemplo 3**).

Busca com find

Investigadores criminais freqüentemente usam o comando `find` do Linux com a opção `--exec` ou jogando

Exemplo 1: Informações com o fdisk

```
01 # fdisk -lu win.dd
02 Disk win.dd: 0 MB, 0 bytes
03 120 heads, 63 sectors/track, 0 cylinders, total
➥0 sectorsUnits = sectors of 1 * 512 = 512 bytes
04 Disk identifier: 0x840b840b
05 Device Boot Start End Blocks Id System
06 win.dd1 *     63 6327719163828+ 7  HPFS/NTFS
```

Exemplo 2: Utilitário Disktype

```
01 # disktype win.dd
02 --- win.dd
03 Regular file, size 3.021 GiB (3243663360 bytes)
04 DOS/MBR partition map
05 Partition 1: 3.017 GiB (3239760384 bytes, 6327657 sectors from 63,
➥ bootable)
06   Type 0x07 (HPFS/NTFS)
07   Windows NTLDR boot loader
08   NTFS file system
09     Volume size 3.017 GiB (3239759872 bytes, 6327656 sectors)
```

sua saída para outro programa com o `xargs` para buscar arquivos com conteúdo ilegal. Após criar e montar uma imagem, `find /mnt -type f` mostra uma lista dos arquivos contidos nela. Porém, como essa técnica não leva em conta arquivos com espaços ou caracteres acentuados no nome, o comando mais adequado é `find /mnt -type f -print0 |xargs -0 ls -al`.

Valores de *hash* também ajudam a encontrar arquivos idênticos e suspeitos no sistema. Para criar automaticamente um hash de cada arquivo listado, o comando `find /mnt -type f -print0 |xargs -0 md5sum` é o mais indicado; é possível até comparar os hashes dinamicamente com outras fontes de referência. Entretanto, faz mais sentido criar um arquivo contendo os hashes de todos os arquivos (**figura 1**).

Duplicatas com hash

Na maioria dos casos, os investigadores já possuem hashes dos arquivos que desejam encontrar. Há especialistas que compilam bancos de dados com hashes de arquivos conhecidos para ajudar na busca de materiais criminosos. Se o objetivo for somente filtrar DLLs da Microsoft num sistema sob investigação, essa técnica é muito útil.

Um simples comando `grep` encontrará qualquer correlação entre o alvo da investigação e a chave de busca. O comando a seguir salva em `grande.txt` uma lista de hashes dos arquivos existentes:

```
# find /mnt -type f -print0 |xargs
➥ -0 md5sum |awk '{print $1}'
➥ |sort -g |uniq >grande.txt
```

Se a lista de hashes de arquivos criminosos estiver em `pqno.txt`, um simples `grep -f pqno.txt grade.txt` encontrará os arquivos suspeitos no sistema.

Palavras-chave

Criar um arquivo texto com palavras-chave para buscar em sistemas comprometidos é interessante. Por exemplo, as palavras *password* e *secret* poderiam ser acrescentadas ao arquivo `palavras.txt`, e então o comando:

```
# cat win.dd |strings |egrep -i
➥ --color -f palavras.txt
```

atravessaria toda a imagem de disco do Windows em busca das palavras *password* e *secret*, realçando-as em vermelho na saída, como mostra a **figura 2**. Essa técnica é particularmente interessante para estender a busca além do sistema de arquivos, para outras áreas do disco rígido como:

- o arquivo de swap ou de hibernação;
- áreas não alocadas do disco;
- dados do *slack space*;
- arquivos apagados.

Exemplo 3: Sistema Windows

```
01 # ls -l /mnt
02 Total 787024
03 -r-------- 1 root root 150528 2003-06-19 13:05 arcldr.exe
04 -r-------- 1 root root 163840 2003-06-19 13:05 arcsetup.exe
05 -r-------- 1 root root      0 2007-12-02 11:59 AUTOEXEC.BAT
06 -r-------- 1 root root    186 2007-12-02 11:43 boot.ini
07 -r-------- 1 root root      0 2007-12-02 11:59 CONFIG.SYS
08 dr-x------ 1 root root   4096 2007-12-02 14:14 Documents and
➥ Settings
09 dr-x------ 1 root root  24576 2007-12-02 14:14 WINNT
10 (...)
11 # cat /mnt/boot.ini
12 [boot loader]
13 timeout=30
14 default=multi(0)disk(0)rdisk(0)partition(1)\ WINNT
15 [operating systems]
16 multi(0)disk(0)rdisk(0)partition(1)\WINNT="Microsoft Windows
➥ 2000 Server" /fastdetect
17 (...)
```

Para encontrar palavras-chave guardadas no texto Unicode de 16 bits usado por sistemas Windows NT, é preciso informar ao comando `strings` se ele deve realizar uma busca *little-endian* ou *big-endian*[5]. Os argumentos necessários são `-el` ou `-eb`, respectivamente. O **exemplo 4** usa o *Ntfsundelete* para ilustrar a recuperação de arquivos via alocações de inodes.

Sleuth Kit

O *Sleuth Kit*[6] (coberto no artigo "Trabalho de detetive") é uma das melhores ferramentas forenses. Ele traz três ferramentas para estender a funcionalidade do `ls`:

- `fls`, que lista os arquivos no nível do sistema de arquivos;
- `ils`, que lista os arquivos com base nos inodes;
- `dls`, que recupera arquivos anteriormente apagados.

Além desses arquivos, há alguns parentes próximos de outras ferramentas padrão do Unix, como `cat` (`icat`) e `find` (`ifind`), além de ferramentas estatísticas como o `istat`. O Sleuth Kit começa pela criação de uma lista com todos os arquivos, com informações de hora (**exemplo 5**). Para obter uma visão organizada com uma linha do tempo dos eventos, pode-se executar `mactime -b /tmp/body`. Para fazer a ferramenta buscar palavras-chave nos arquivos deletados numa partição NTFS:

```
# dls /dev/loop0 > naoalocados
# cat naoalocados |strings |egrep
➥ -i --color -f palavras.txt
```

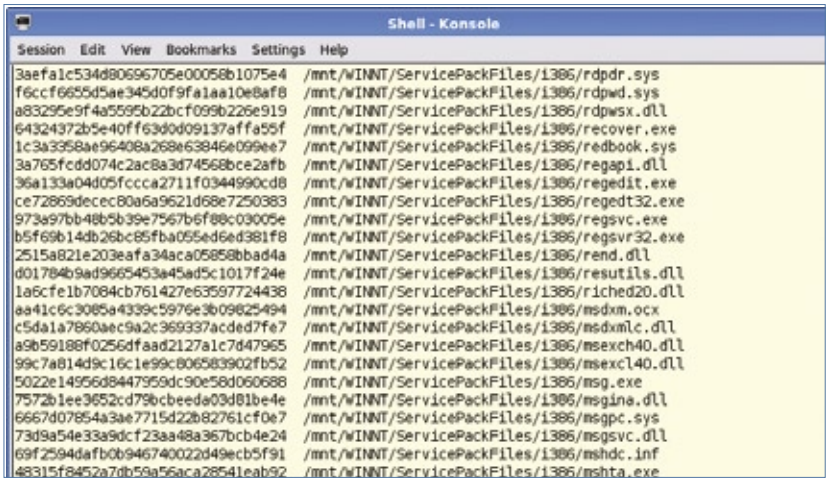


Figura 1 O *Md5sum* pode criar *hashes* únicos para os arquivos do sistema Windows. Com isso, pode-se comparar esses valores com os de arquivos conhecidos.

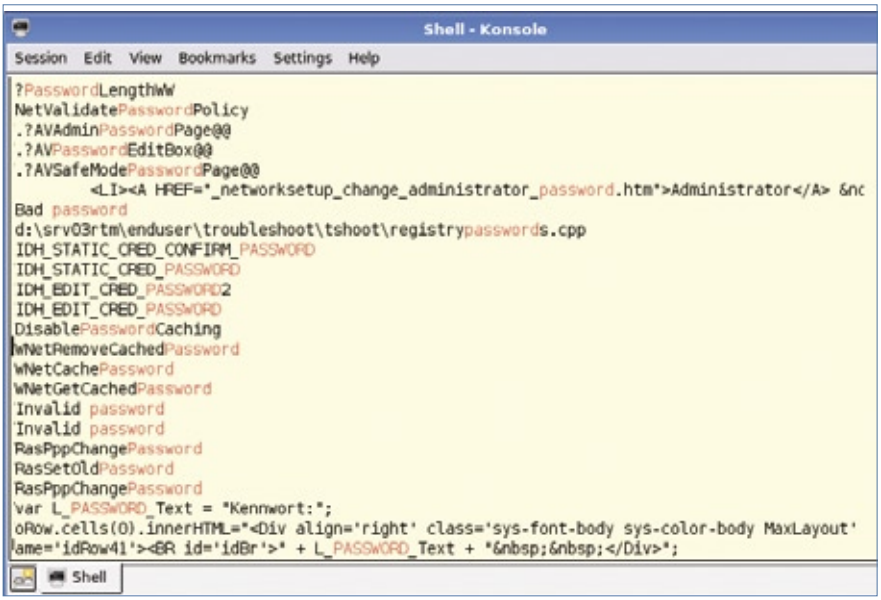


Figura 2 Ferramentas padrão do Linux em ação. Elas varrem a partição em busca de palavras-chave, realçando todas as coincidências.

O comando `dls` converte o espaço não alocado para um arquivo, que o `cat` depois repassa para o `strings` e para o `egrep`.

Slack space

O chamado slack space[7] é o espaço não utilizado dos blocos do sistema de arquivos. Ele existe, por exemplo, quando se salva um arquivo de 2 KB num bloco de 4 KB. Todos os sistemas Windows populares simplesmente preenchem esse espaço não utilizado com dados aleatórios da RAM para completar os blocos.

Ferramentas como o `dls` do Sleuth Kit ou o *bmap*[8] permitem que o investigador recupere dados que o usuário muitas vezes nem sabe que sequer armazenou no disco. Detetives já utilizaram essa técnica para reconstruir emails incriminadores.

O `dls`, em conjunto com a opção `-s`, é particularmente útil para esse propósito:

```
# dls -s /dev/loop0 > slack
# cat slack |strings |egrep -i U -
➥-color -f palavras.txt
```

Isso oferece aos especialistas a possibilidade de buscar palavras-chave no slack space. Já no Linux, os sistemas de arquivos modernos não são afetados

por esse problema, pois preenchem os bytes não utilizados com inofensivos zeros, diretamente do `/dev/zero`.

Restauração

O comando `nfsundelete`, do pacote *Ntfsprogs*, oferece a possibilidade de restaurar arquivos apagados em partições NTFS. Antes de executá-lo, no entanto, é preciso desmontar o dispositivo `/dev/loop0`. Sem especificar outras opções, o comando `ntfsundelete /dev/loop0` simplesmente exibe uma lista de todos os arquivos recuperáveis (**figura 3**). O exemplo da **figura 3** está recuperando o arquivo `msiinst.exe` no inode 11137.

Os arquivos do disco rígido podem fornecer muitas informações de usuários. Tanto o *Internet Explorer* quanto o *Firefox* armazenam seus históricos no sistema de arquivos. O investigador precisa apenas instalar dois programas para analisar essas informações:

- *Pasco*[9] para o Internet Explorer;
- *Mork.pl*[10] para o Firefox.

O **exemplo 6** mostra uma típica seqüência de análise: o Explorer guarda informações de cada perfil nos arquivos `index.dat`. Um `find` nesse arquivo mostra uma lista de páginas acessadas pelo navegador.

Dentro do Firefox

O Firefox armazena seus dados de histórico no arquivo `history.dat`. A primeira coluna contém informações de data e hora no formato do Unix. O terceiro comando do **exemplo 6** converte esse valor para um formato compatível com pessoas.

O *Dumphive*[11] oferece uma técnica para tor-

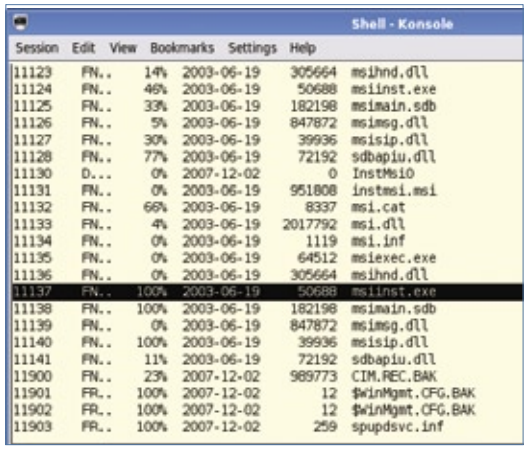


Figura 3 O *Ntfsundelete* exibe os arquivos apagados que podem ser recuperados. A primeira coluna contém os números de inodes, necessários para restaurar os arquivos no sistema de arquivos da imagem.

Exemplo 4: Recuperação de arquivos

```
01 # ntfsundelete -u -i11137 /dev/loop0
02 Inode  Flags %age Date       Size  Filename
03 ----------------------------------------------------
04 11137  FN..  0%   2003-06-19 50688 msiinst.exe
05 Undeleted 'msiinst.exe' successfully.
06 file msiinst.exe
07 msiinst.exe: MS-DOS executable PE for MS Windows (DLL) (GUI)
➥ Intel 80386 32-bit
```

Exemplo 5: Timestamp dos arquivos

```
01 01 # fls -o 63 -m "C:" -r win.dd > /tmp/body
02 02 # mactime -d -b /tmp/body
03 03 Thu Jun 19 2003 13:05:04,16656,m..,-/-rwxrwxrwx,0,0,315-128-3,C:/WINNT/system32/cdmodem.dll
04 05 Thu Jun 19 2003 13:05:04,11792,m..,-/-rwxrwxrwx,0,0,11267-128-3,C:/WINNT/ServicePackFiles/i386/
➥ partmgr.sys
05 07 Thu Jun 19 2003 13:05:04,7440,m..,-/-rwxrwxrwx,0,0,8093-128-3,C:/WINNT/ServicePackFiles/i386/
06 bhp.dll
07 09 Thu Jun 19 2003 13:05:04,1011764,m..,-/-rwxrwxrwx,0,0,7102-128-3,C:/WINNT/system32/mfc42u.dll
08 11 Thu Jun 19 2003 13:05:04,65593,m..,-/-rwxrwxrwx,0,0,6552-128-3,C:/Programme/Outlook Express/
09 csapi3t1.dll
10 13 Thu Jun 19 2003 13:05:04,122640,m..,-/-rwxrwxrwx,0,0,858-128-3,C:/WINNT/system32/idq.dll
11 14 Thu Jun 19 2003 13:05:04,166672,m..,-/-rwxrwxrwx,0,0,7178-128-3,C:/WINNT/system32/qcap.dll
12 15 Thu Jun 19 2003 13:05:04,65593,m..,-/-rwxrwxrwx,0,0,11555-128-3,C:/WINNT/Sersystem32/i386/
➥ csapi3t1.dll
```

nar o registro de sistemas Windows mais facilmente legível (**figura 4**). O comando `dumphive /mnt/WINNT/system32/config/system system.txt` guarda o registro num arquivo texto separado que pode ser analisado com ferramentas do Unix.

Senhas do Windows

O acesso ao sistema é interessante por si só, mas descobrir as senhas dos usuários costuma facilitar a investigação, pois muitos usuários mantêm uma única senha para acessar o sistema e seus diversos serviços online. Além disso, com a senha, o detetive consegue fazer login com as credenciais do usuário ao executar o sistema num ambiente virtual, por exemplo.

Além de empregar ataques de força bruta e ferramentas como *John the Ripper*, que usam dicionários, o administrador Linux pode recorrer a outras ferramentas, como *Bkhive*, *Samdump2* e *Ophcrack*[12].

Extrair senhas locais de sistemas Windows NT de um arquivo SAM não é difícil, principalmente se forem usadas múltiplas ferramentas. Por exemplo, o John the Ripper detecta automaticamente um arquivo SAM do Windows quando o recebe. Nesse ponto, o comportamento da Microsoft quanto a senhas é bem útil: embora as credenciais do Windows possam ter até 14 caracteres, o sistema as divide em duas cadeias de sete caracteres cada. Isso facilita muito a quebra de senhas sem grande força bruta.

No Vista, a Microsoft fechou essa brecha e substituiu os hashes *Lanmanager* por hashes NT. Administradores do XP podem configurar isso manualmente e investigadores com Linux precisam usar o `dumphive` para checar se o registro contém uma entrada com valor `1` para `HKEY_LOCAL_MACHINE/SYSTEM/CurrentControlSet/Control/Lsa`. Se for o caso, a única opção pode ser comprar um DVD de 8,5 GB com tabelas comerciais Lanmanager ou NT.

Assim como as outras ferramentas citadas neste artigo, o Ophcrack está disponível nos repositórios das principais distribuições Linux. Ele requer os hashes e tabelas *rainbow* da máquina Windows. Depois de terminar a instalação, o investigador já pode usar a conveniente interface gráfica e dar um duplo-clique para decifrar a senha de um usuário (**figura 5**).

Exemplo 6: Histórico do navegador

```
01 # mount -o ro,noatime,noexec /dev/loop0 /mnt
02 # find /mnt -iname "index.dat" -exec pasco '{}' ';'
03 TYPE  URL  MODIFIED  TIME  ACCESS  TIME   FILENAME   DIRECTORY
➥HTTP HEADERS
04 URL   http://www.google.com/favicon.ico  06/07/2006  21:35:34
➥12/02/2007
05 12:14:28  favicon[1].ico   NGORCTFI HTTP/1.1  200 OK
➥Content-Type: image/x-icon
06 Content-Length: 1406 ~U:administrator
07 REDR  http://msn.ivwbox.com/cgi-bin/ivw/CP/MSN01000000;?r=
08       12/02/2007 12:11:32 12/02/2007 12:11:32
09 URL  Visited: Administrator@http://www.google.com 12/02/2007 12:14:28
10 URL  Visited: Administrator@http://www.msn.com 12/02/2007 14:33:54
➥ 12/02/2007
11 14:33:54
12 # find /mnt -iname "history.dat" -exec mork.pl '{}' ';'
13 1202727704   1  http://www.linux4afrika.de/index.php?id=155&L=1
14 1202727670   1  http://www.linux4afrika.de/index.php?id=154&L=1
15 1202727641   1  http://www.linux4afrika.de/index.php?id=60&L=1
16 1202727641   2  http://www.linux4afrika.de/
17 1202727555   1  http://br-linux.org/916916.html
18 1202726960   1  http://br-linux.org/916917.html
19 1202726892   1  http://br-linux.org/916908.html
20 1202726827   3  http://br-linux.org/
21 1202726394   2  http://www.linux-magazine.com/
22 1202726204   2  http://www.google.com.br/
23 # find /mnt -iname "history.dat" -exec mork.pl '{}' ';' | awk
➥'{print strftime("%F,%R",$1),$2,$3}'
24 2008-02-11 11:40 1 http://www.linuxmagazine.com.br/noticia/
➥qual_o_destino_doreiser4
➥ausgaben/2008/03/zwerg_am_druecker
25 2008-02-11 11:39 2 http://www.linuxmagazine.com.br/
26 2008-02-11 11:36 2 http://www.google.com.br/
27 (...)
```

Controladores do domínio

Uma técnica diferente é necessária para sistemas Windows que fazem login em controladores de domínio; nesse caso, as credenciais não são guardadas localmente no cliente. Porém, na maioria dos casos, basta rodar um capturador de pacotes para obter a troca do login, identificar os pacotes de dados relevantes, salvá-los num arquivo e depois encaminhar

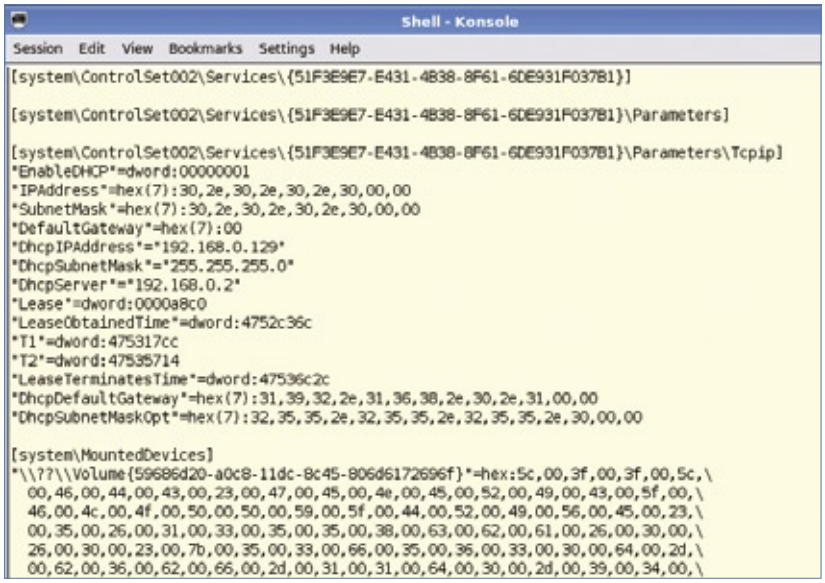


Figura 4 O *Dumphive* converte o registro do Windows para texto puro, permitindo o uso de ferramentas como o `grep` para reconstruir configurações críticas.

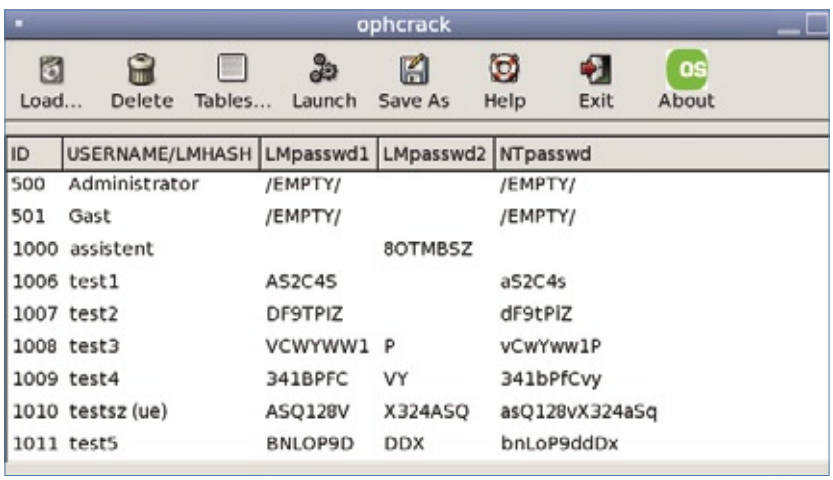
sua saída para o Ophcrack. Essa técnica é mais complexa e exige acesso à rede.

Comparativamente simples

Com a adição de alguns pacotes extras, o mundo do Windows fica totalmente aberto para um investigador munido do Linux. Quem se interessar pela área pode experimentar também as ferramentas da Foundstone[13]. Elas oferecem a possibilidade de recuperar *cookies*, itens da lixeira do Windows apagados há tempos e muito mais.

O campeão de usabilidade nessa área é o *Live CD* Ophcrack, que elimina até a necessidade de digitação de comandos de shell e exibe as senhas dos usuários locais da máquina logo após a inicialização.

Num sistema Windows XP SP2 usado para testes, o CD levou apenas 280 segundos para descobrir as credenciais das cinco contas de usuários (que incluíam até 14 caracteres; veja a **figura 5**). A versão do CD inclui somente as tabelas para senhas alfanuméricas sem caracteres acentuados. Para ter mais que isso, é necessário investir em tabelas rainbow comerciais. ■



ID	USERNAME/LMHASH	LMpasswd1	LMpasswd2	NTpasswd
500	Administrator	/EMPTY/		/EMPTY/
501	Gast	/EMPTY/		/EMPTY/
1000	assistent		8OTMBSZ	
1006	test1	AS2C4S		aS2C4s
1007	test2	DF9TPIZ		dF9tPIZ
1008	test3	VCWYWW1	P	vCwYww1P
1009	test4	341BPFC	VY	341bPfCvy
1010	testsz (ue)	ASQ128V	X324ASQ	asQ128vX324aSq
1011	test5	BNLOP9D	DDX	bnLoP9ddDx

Figura 5 Basta um duplo-clique para quebrar a senha de um usuário do Windows.

Mais informações

[1] Guidance Software: http://www.guidancesoftware.com

[2] X-Ways: http:/www.x-ways.net/corporate/index-m.html

[3] Ewfacquire: https://www.uitwisselplatform.nl/projects/libewf

[4] Helix: http://www.e-fense.com/helix

[5] Extremidade (Endianness) na Wikipédia: http://tinyurl.com/5ja5yu

[6] Sleuth Kit: http://sleuthkit.org

[7] Slack space na Wikipédia (em inglês): http://tinyurl.com/683j2x

[8] bmap: http://www.packetstormsecurity.org/linux/security/bmap1.0.17.tar.gz

[9] Pasco: downloads.sourceforge.net/odessa/pasco_20040505_1.tar.gz?modtime=1083715200&big_mirror=0

[11] Mork.pl: http://www.jwz.org/hacks/mork.pl

[12] Dumphive: http://v4.guadalinex.org/guadalinex,Äëtoro/pool/main/d/dumphive/dumphive_0.0.3,Äë1_i386.deb

[13] Ophcrack: http://ophcrack.sourceforge.net

[14] Foundstone: http://www.foundstone.com/us/sources,Äëfree,Äëtools.asp

Eric Gevaert – Fotolia

Explorando a Open Computer Forensics Architecture

Da terra dos moinhos

Automatize os processos de análise forense com a arquitetura desenvolvida pela polícia holandesa.

por Ralf Spenneberg

O crime digital costuma pôr a polícia sob pressão. Ela não tem o pessoal necessário para coletar e analisar os grandes volumes de provas digitais que geralmente acompanham investigações mais profundas. Ao mesmo tempo, provas digitais estão tornando-se progressivamente importantes – dados de telefones celulares e computadores de suspeitos podem fornecer provas circunstanciais e até fatos reais. A polícia holandesa desenvolveu a *Open Computer Forensics Architecture* (OCFA[1]) como uma ferramenta de código aberto para investigadores criminais profissionais, sendo que as autoridades holandesas utilizam essa plataforma modular em suas próprias investigações. A arquitetura da OCFA é uma combinação de várias ferramentas e bibliotecas forenses e divide o processo de análise em duas partes. Primeiro, os especialistas com conhecimento de análises forenses digitais extraem o conteúdo de discos rígidos e outros dispositivos. Depois, os investigadores criminais usam uma interface web para analisar os dados e buscar provas.

Há pacotes da OCFA 2.0.2 para várias distribuições, incluindo formatos RPM e DEB. Nos arquivos compactados disponíveis para download, há ainda pacotes extras e guias de instalação que descrevem os pacotes que requerem instalação manual.

A versão 2.1.0 é a mais atual e somente seu código-fonte está disponível. Os criadores da OCFA enxergam o processo de análise como um tipo de lavagem digital de dados (*Digiwash*) e, portanto, instalam as ferramentas no diretório `/usr/loca/digiwash/`.

Um dos maiores obstáculos da análise forense é o grande volume de evidências. Os investigadores enfrentam a tarefa de identificar materiais incriminatórios dentre centenas de gigabytes de dados irrelevantes. Mas pular arquivos e diretórios simplesmente porque seus nomes soam inofensivos não é uma solução plausível. Muitas ferramentas forenses auxiliam o investigador por meio de análises automáticas e caracterização de arquivos identificados. O Digiwash leva essa idéia um passo além, usando o utilitário `file` para identificar o tipo de arquivo. Depois, ele segue analisando automaticamente tipos específicos de arquivos, economizando um pouco do trabalho sujo para o investigador. A OCFA usa o *Lucene* para indexar arquivos do Microsoft Word e outros documentos do MS Office. O texto puro é extraído com o `antiword`. Arquivos PDF são convertidos com ajuda do `pdftotext`, e o `mailwash` extrai arquivos e metadados de caixas de email. Os desenvolvedores até criaram uma forma de capturar as informações de chaveiros PGP, mapeando as IDs das chaves de emails assinados e criptografados a nomes em texto limpo. A OCFA também agrupa fotos e gera miniaturas.

A plataforma disseca automaticamente arquivos zipados e analisa seu

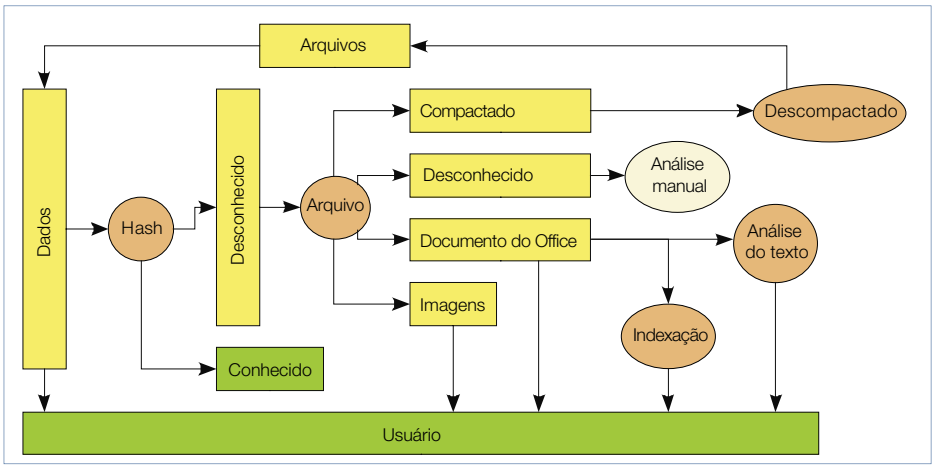


Figura 1 A *OCFA* analisa automaticamente a enchente de dados e a coloca à disposição dos investigadores.

conteúdo. Dessa forma, ela analisa recursivamente todos os dados, colocando-os à disposição dos investigadores (**figura 1**).

Impressões digitais

Para reduzir o volume de dados a serem analisados, os investigadores forenses podem integrar bancos de dados de *hashes* de arquivos conhecidos. Os bancos contêm *checksums* MD5 ou SHA1 de arquivos que devem perfeitamente ser ignorados. Por exemplo, todos os arquivos inalterados que pertençam ao sistema operacional são irrelevantes, apesar de os investigadores se interessarem por qualquer modificação feita por cavalos de tróia.

Bancos de dados de checksums de arquivos conhecidos estão disponíveis para download gratuito no National Institute for Standards and Technology (NIST)[2]. Outras ferramentas forenses, como o *Autopsy*[3], também se baseiam na National Software Reference Library (NSRL). A biblioteca está disponível em formato zipado como uma coleção de quatro imagens de CD. Um script em *Perl* presente no pacote da OCFA faz referência aos arquivos ISO para calcular seus próprios conjuntos de hashes, que o administrador deve copiar em seguida para o diretório `digiwash/`. O processo de conversão leva algum tempo.

Arquitetura

O roteador é uma parte central da arquitetura OCFA, pois é responsável pelo processamento recursivo de arquivos, que ele analisa utilizando softwares externos antes de retornar para o processo de análise os arquivos que o processo encontrar (**figura 2**). Um *relay anycast* lida com as comunicações entre os módulos individuais. O relay coordena o envio de mensagens e também lida com o balanceamento de carga. Essa técnica permite que os investigadores rodem múltiplas instâncias de um módulo em um ambiente distribuído em vários computadores; em outras palavras, a OCFA suporta a execução em clusters.

O framework OCFA pode usar outros pacotes externos de software caso necessário. Um *patch* permite que usuários integrem ainda o Sleuth Kit[4] e o *Scalpel*[5].

Interfaces misturadas

Dados recuperados pela OCFA ficam disponíveis para o investigador por meio da linha de comando. Ele precisa de privilégios de root para iniciar uma análise de caso, pois é necessário reiniciar o servidor web *Apache* para isso. Do ponto de vista do servidor web, cada novo caso é um *VirtualHost*. O investigador utiliza então o Apache para acessar os dados extraídos.

A interface também informa ao investigador se a extração dos dados está completa. A interface web exibe as filas atuais e seus respectivos estados. Em outras palavras, a OCFA de fato automatiza a comunicação entre o investigador e o especialista forense.

A polícia holandesa desenvolveu também um programa para Windows, para uso interno. O programa, chamado *Washbrush*, analisa caixas de email do Outlook e Outlook Express e passa os resultados para o Digiwash. No entanto, esse programa só está disponível para as autoridades holandesas.

Os policiais também estão criando outros módulos OCFA e uma interface mais moderna. O software não adotará a licença GPL, mas será disponibilizado sob um acordo de não divulgação (NDA).

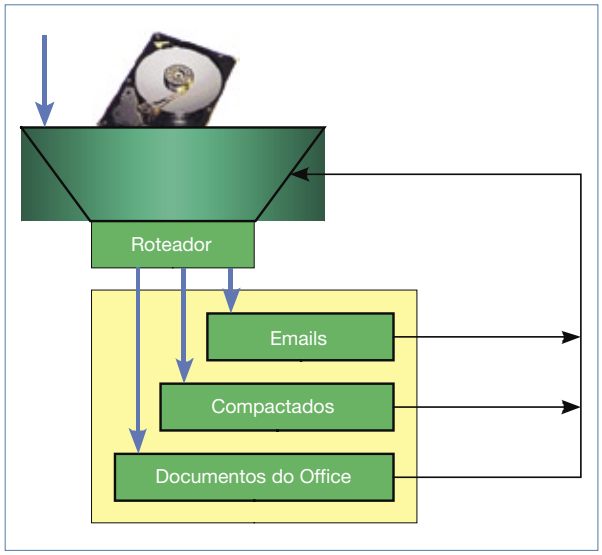


Figura 2 O roteador chama os módulos de acordo com o formato do arquivo, retornando os arquivos encontrados.

Processamento em massa

A complexa e demorada instalação da OCFA vale a pena em ambientes que suportem projetos de análise forense grandes e complexos – e também nos casos em que as tarefas forenses e de investigação sejam facilmente separáveis. Note, contudo, que as interfaces gráficas espartanas da OCFA não oferecem a conveniência de outras ferramentas dessa área. ■

Mais informações

[1] OCFA: http://ocfa.sourceforge.net

[2] NIST e NSRL: http://www.nsrl.nist.gov

[3] Autopsy: http://www.sleuthkit.org/autopsy/

[4] Sleuth Kit: http://www.sleuthkit.org/

[5] Scalpel: http://www.digitalforensicssolutions.com/Scalpel/

Décima quinta aula da preparação LPIC-2

LPI nível 2: aula 15

Roteadores, firewalls e NAT. Proteção de servidores FTP, utilização do OpenSSH, tcp_wrappers e outras ferramentas de segurança.

Tópico 212: Segurança do Sistema

2.212.2 Configurar um roteador

Classes de endereços

O primeiro passo para evitar problemas numa rede é assegurar-se da escolha correta dos endereços IP. O IANA (*Internet Assigned Numbers Authority*) define três categorias para endereços IP:

- **Categoria 1**: Hosts que não precisam ter acesso a outros hosts em redes externas ou na Internet. Os hosts nessa categoria podem utilizar números IP que existem em redes externas, mas que devem ser únicos na rede local;
- **Categoria 2**: Hosts que precisam ter acesso a alguns serviços externos (email, ftp, www) que podem ser mediados via um host gateway. Para a maioria dos hosts nessa categoria, o acesso direto com um IP único é desnecessário ou mesmo indesejado, visto que pode enfraquecer a segurança. Como na primeira categoria, os hosts podem utilizar números IP que existem em redes externas, mas que devem ser únicos na rede local;
- **Categoria 3**: Hosts que necessitam de conectividade direta com a Internet. O número IP para hosts nessa categoria deve ser único em toda Internet.

Os endereços de hosts na primeira e segunda categoria são chamados *privados*. Os da terceira categoria são chamados *públicos*.

Os número IPs reservados para hosts privados são delimitados dentro dos seguintes grupos:

```
10.0.0.0      -    10.255.255.255
➥(10/8)
172.16.0.0    -    172.31.255.255
➥(172.16/12)
➥192.168.0.0  -    192.168.255.255
(192.168/16)
```

Como os endereços privados são de competência exclusiva da rede local onde existem, as regras para criação de classes de rede são flexíveis. A máscara de rede pode ser manipulada para melhor satisfazer as necessidades da rede ou a preferência do administrador.

Rotas

É comum que redes privadas comuniquem-se com a Internet por meio de um roteador, que por sua vez comunica-se tanto com a rede privada interna quanto com a rede externa, por meio de um IP público. A tabela de rotas no roteador determina para onde devem ser encaminhados. Como já vimos no tópico 205, o principal comando para manejo de rotas é o `route`, verificável no **exemplo 1**.

Essa é uma tabela de rotas típica de um computador que age como roteador e gateway. Existem quatro interfaces de rede conectadas e configuradas. Três delas conectam-se a redes locais (endereços privados e

Exemplo 1: Comando route

```
# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
192.168.3.0     0.0.0.0         255.255.255.0   U     0      0        0 eth3
192.168.2.0     0.0.0.0         255.255.255.0   U     0      0        0 eth2
192.168.1.0     0.0.0.0         255.255.255.0   U     0      0        0 eth1
201.52.48.0     0.0.0.0         255.255.240.0   U     0      0        0 eth0
127.0.0.0       0.0.0.0         255.0.0.0       U     0      0        0 lo
0.0.0.0         201.52.48.1     0.0.0.0         UG    0      0        0 eth0
```

uma à Internet (endereço público). Também a interface `lo`, interface de comunicação interna que não corresponde a rede externa alguma, possui um endereço privado.

Todos os pacotes que chegarem serão direcionados aos respectivos destinos nas redes locais. Se o destino de um pacote não pertence a host algum em uma rede local, ele será direcionado ao gateway padrão, indicado na última linha da tabela de rotas com as *flags* `UG`.

Apesar de, via de regra, a rota para uma rede ser automaticamente criada quando a interface é configurada, pode ser necessário adicionar uma rota manualmente. Essa tarefa pode ser realizada com o comando `route`:

```
route add -net 192.168.1.0 netmask
➥255.255.255.0 dev eth1
```

Este comando adiciona a rota para a rede 192.168.1.0 por meio da interface eth1. Para criar uma rota padrão, outra forma é utilizada:

```
route add default gw 192.168.1.1
```

Essa forma é um atalho para a forma extensa:

```
route add -net 0.0.0.0 netmask
➥0.0.0.0 gw 192.168.1.1
```

De maneira praticamente idêntica, rotas podem ser removidas utilizando `del` no lugar de `add`.

NAT

Nessa configuração de exemplo, um pacote com origem na rede 192.168.1.0 e destino na rede 192.168.2.0 atravessará o roteador e a comunicação será estabelecida nas duas pontas através do roteador. Porém, um pacote com origem na rede 192.168.1.0 (ou qualquer outra rede de IPs privadas) com destino na rede de IPs públicos (como um site da Web) não conseguirá estabelecer comunicação, pois um IP privado é ambíguo na rede de IPs públicos.

Para resolver este problema, é utilizado um procedimento chamado NAT (*Network Address Translation*). Com o NAT é possível que um host na rede privada comunique-se com hosts na rede pública (Internet).

A ativação do NAT é feita no roteador, por meio do comando `ipchains` (em desuso) ou `iptables`. O comando `iptables` é responsável por definir regras para o trânsito de pacotes IP controlado pelo kernel. O trânsito dos pacotes é dividido em categorias pelo kernel, categorias essas chamadas de tabelas, justificando o nome do comando. Cada tabela possui linhas (também chamadas correntes, ou *chains*) que podem receber diversas regras, como veremos adiante.

A tabela de atuação é indicada pela opção `-t` do comando `iptables`. Se nenhuma tabela for especificada, a tabela assumida será a `filter`.

Tabelas do `iptables`:

- `filter`: É a tabela padrão. Contém as chains embutidas `INPUT` (para pacotes que chegam ao host local), `FORWARD` (para pacotes sendo roteados pelo host local) e `OUTPUT` (para pacotes gerados no host local e destino externo). Essa é a tabela utilizada para construção de firewalls;
- `nat`: Para pacotes que criam novas conexões (traduções e redirecionamentos). Contém as chains embutidas `PREROUTING`, `OUTPUT` e `POSTROUTING`;
- `mangle`: Para alterações especializadas de pacotes. Contém as chains `INPUT`, `OUTPUT`, `PREROUTING`, `FORWARD` e `POSTROUTING`.

As operações dentro de uma chain são determinadas por argumentos-comando:

- `-A`: Adicionar regra na chain
- `-I`: Inserir regra numa posição específica dentro da chain
- `-R`: Substituir regra na chain
- `-D`: Apagar chain
- `-N`: Criar chain personalizada
- `-X`: Apagar chain vazia
- `-P`: Definir política para uma chain embutida
- `-L`: Listar a(s) regra(s) presentes em uma chain
- `-F`: Apagar todas as regras em uma chain
- `-Z`: Zerar os contadores de pacotes em todas as regras de uma chain.

Especificações de regras (interceptam os pacotes que correspondam a elas):

- `-s endereço`: Ou `--source endereço`. Endereço de origem do pacote. Pode ser nome de rede, nome de host, IP de rede/máscara de rede ou simplesmente um endereço IP. Se endereço precedido de "!", intercepta os pacotes que não corresponderem à condição;
- `-d endereço`: Ou `--destination endereço`. Endereço de destino do pacote. Mesmo formato de `-s`. Se endereço precedido de "!", intercepta os pacotes que não corresponderem à condição;
- `-p protocolo`: Ou `--protocol protocolo`. Define o protocolo. Pode ser tcp, udp, icmp ou all. Se protocolo precedido de "!", intercepta os pacotes que não corresponderem à condição;
- `-i interface`: Ou `--in-interface interface`. Interface pela qual o pacote chegou. Se o nome interface for seguida do sinal "+" (`interface+`), aplicará a todas as interfaces cujos nomes comecem por "`interface`". Se `interface` for precedido de "!", intercepta os pacotes que não corresponderem à condição. Se `-i interface` não existir, todas interfaces serão assumidas;
- `-o interface`: Ou `--out-interface interface`. Interface pela qual o pacote será enviado. Se o nome

interface for seguida do sinal "+" (`interface+`), aplicará a todas interfaces cujos nomes comecem por "`interface`". Se `interface` precedido de "!", intercepta os pacotes que não corresponderem à condição. Se `interface` for omitido, toda interface será assumida;

- `-j ação`: Ou `--jump ação`. Targets (ações) para o(s) pacote(s) interceptados. Targets comuns para firewall são `ACCEPT` (Permite a passagem normal do pacote) e `DROP` (Descarta o pacote);
- `-m módulo`: Ou `--match módulo`. Usa módulo estendido "`módulo`". Há muitos tipos de módulos de controle adicionais e opções extras para cada um deles. Um muito usado para firewall é o módulo `state`, cuja opção `--state estado` permite determinar qual a relação de um pacote com as conexões existentes. Possíveis valores para estado são `INVALID` (o estado não pôde ser determinado), `ESTABLISHED` (o pacote pertence a uma conexão ativa), `NEW` (indicando que o pacote inicia nova conexão e `RELATED` (o pacote inicia outra conexão, porém relacionada a uma conexão existente).

Para criar uma regra de NAT, o `iptables` pode ser utilizado da seguinte forma:

```
iptables -t nat -A POSTROUTING -s
➥192.168.2.0/24 -o eth0 -j SNAT
➥--to-source 201.52.50.11
```

Explicação das opções utilizadas:

- `-t nat`: Determina que a tabela `nat` deve ser utilizada;
- `-A POSTROUTING`: Inclui a regra na corrente `POSTROUTING`;
- `-s 192.168.2.0/24`: Determina que a regra aplica-se a pacotes originados na rede privada `192.168.2.0/24`;
- `-o eth0`: Determina que a regra aplica-se a pacotes cujo destino seja a interface `eth0`. No caso do exemplo, trata-se da interface com IP público;
- `-j SNAT --to-source 201.52.50.11`: A opção `-j` determina a ação a ser executada para o pacote que se enquadrar nas regras estabelecidas. `SNAT` (*source* NAT) determina que o IP de origem do pacote enviado será o informado com a opção `--to-source` (o IP do roteador). Dessa forma, o pacote será enviado para a rede de IPs públicos tendo como origem um IP público válido (o IP da interface etho do roteador). O roteador identifica pacotes pertencentes a uma conexão NAT e direciona para o host correspondente na rede privada local.

Caso o IP público do roteador seja dinâmico, deve ser utilizado o comando `-j MASQUERADE`.

Para permitir NAT, é necessário alterar o conteúdo do arquivo `/proc/sys/net/ipv4/ip_forward` para `1`. No caso do exemplo, este arquivo já fora alterado para permitir roteamento dos pacotes das redes privadas internas pelo servidor.

Redirecionamentos

A técnica de NAT possibilita que hosts com IPs provados acessem hosts na rede pública (Internet). No entanto, conexões iniciadas tanto em IPs privados quanto em IPs públicos continuam impossibilitadas de serem estabelecidas junto a um IP privado numa rede externa. Isso porque o único IP visível na rede pública é o IP do roteador.

Porém, é possível criar um filtro de redirecionamento no roteador para que determinados pacotes sejam enviados para um host na rede privada. O próprio `iptables` é utilizado para criar tais redirecionamentos.

Por exemplo, é possível fazer com que todas as conexões destinadas à porta 80 (http) sejam redirecionadas para um host na rede privada. Dessa forma, o servidor Web pode ser mantido num host dentro da rede privada ou numa máquina virtual dentro do roteador. O comando para criar esse redirecionamento pode ser escrito da seguinte forma:

```
iptables -t nat -A PREROUTING -p tcp
➥--dport 80 -j DNAT --to-destination
➥192.168.2.2:80
```

Dessa vez, foi utilizada a corrente `PREROUTING`, que manipula os pacotes na medida que entram na tabela. A opção `-p` especifica o protocolo para a próxima opção, `--dport`, que determina a porta que será redirecionada. Em seguida, `-j DNAT` indica que trata-se de uma tradução para outro IP de destino, indicado com a opção `--to-destination` ou simplesmente `--to`.

Outra possibilidade é definir uma porta no roteador que será utilizada para entrar no host privado. Por exemplo, pode-se redirecionar pedidos de conexão na porta 22000 do roteador para o login via ssh num host privado:

```
iptables -t nat -A PREROUTING -p
➥tcp --dport 22000 -j DNAT
➥--to-destination 192.168.2.2:22
```

Para apenas redirecionar uma porta do roteador para outra porta no próprio roteador, utiliza-se a ação `REDIRECT`:

```
iptables -t nat -A PREROUTING -p
➥tcp --dport 80 -j REDIRECT
➥--to-port 8080
```

Neste exemplo, todas as solicitações para a porta 80 serão redirecionadas para a porta 8080 na própria máquina.

Bloqueando ataques

Além de criar redirecionamentos, é possível utilizar o `iptables` para bloquear transmissões utilizando o

mesmo modelo de regras. Para criar regras de bloqueio, é utilizada a tabela `filter`, que é a tabela padrão do `iptables` e por isso não precisa ser indicada com a opção `-t`.

Uma estratégia simples de firewall é bloquear todas as tentativas de conexão para a máquina com IP público e apenas permitir a entrada de conexões já estabelecidas. A seguir, é mostrado um exemplo de aplicação dessa estratégia.

Apagar todas as regras da tabela `filter`:

```
iptables -t filter -F
```

Estabelecer política de descartar todos os pacotes na chain `INPUT` da tabela `filters`:

```
iptables -t filter -P INPUT DROP
```

Liberar todos os pacotes gerados localmente:

```
iptables -t filter -A INPUT -i lo
➥-j ACCEPT
```

Liberar para entrar pela interface etho somente os pacote pertencentes (`ESTABLISHED`) ou relacionados (`RELATED`) a uma conexão existente:

```
iptables -t filter -A INPUT -m
➥state --state ESTABLISHED,RELATED
➥-j ACCEPT
```

As novas regras podem ser verificadas com o comando `iptables -L`. Para liberar o acesso externo à porta 80 (http), cria-se outra regra:

```
iptables -A INPUT -p tcp --dport
➥80 -j ACCEPT
```

Também é recomendável manter a porta 22 (ssh) aberta, permitindo um acesso administrativo seguro ao servidor:

```
iptables -A INPUT -p tcp --dport
➥22 -j ACCEPT
```

Dessa forma, apenas a porta 80 e 22 estarão visíveis para a rede pública, o que garante um bom nível de segurança. Apesar disso, o servidor ainda pode estar vulnerável a um ataque como *Denial of Service* (DoS). Um ataque desse tipo consiste em fazer um número altíssimo de solicitações a um servidor, de forma que este não seja capaz de responder a todos e torne-se inacessível.

Para evitar que tal cenário ocorra, devem ser ativados os recursos de verificação de endereço de origem (para evitar IP *spoofing* – IPs forjados) e proteção TCP SYN Cookie:

```
# sysctl -w net.ipv4.conf.all.rp_
➥filter = 1
# sysctl -w net.ipv4.tcp_
➥syncookies = 1
```

Essas alterações devem ser incluídas em `/etc/sysctl.conf` ou nos srcripts de inicialização do sistema.

2.212.3 Segurança de servidores FTP

Existem vários servidores FTP. Dentre eles, o mais utilizado e considerado o mais seguro é o vsFTP (*Very Secure FTP*). O vsFTP foi desenvolvido com enfoque na segurança.

O daemon do vsFTP é o `vsftpd`. O arquivo de configuração `/etc/vsftpd.conf`. Uma das principais características do vsFTP é criar um ambiente *chroot* quando um usuário entra no sistema via FTP, sem necessidade de preparar uma árvore de diretórios específica para isso.

Em primeiro lugar, é necessário criar o diretório base para o FTP, geralmente `/home/ftp` ou `/var/ftp`. Este diretório deve pertencer ao usuário `root` e não ter permissão de escrita para o usuário `ftp`. Caso contrário, o `vsftpd` informará que existe uma brecha de segurança e não funcionará corretamente.

As configurações são simples e dificilmente é necessário alterar o padrão. Se o `vsftpd` não for utilizado com o `inetd` ou com o `xinetd`, a opção `listen=YES` deverá estar presente no arquivo `/etc/vsftpd.conf`.

Para que o servidor aceite conexões anônimas, as seguintes opções devem ser utilizadas em `/etc/vsftpd.conf`:

```
anonymous_enable=YES
write_enable=YES
anon_upload_enable=YES
chroot_local_user=YES
```

A opção mais importante é `chroot_local_user=YES`, que fará com que o diretório raiz mostrado ao cliente FTP seja o próprio diretório do usuário ou `/home/ftp` para conexões anônimas. Para que o usuário anônimo seja capaz de copiar arquivos para o servidor, deve ser criado um diretório em `/home/ftp` – normalmente chamado `incoming` – que permita a escrita para o usuário ftp:

```
# mkdir /home/ftp/incoming
# chown ftp:ftp /home/ftp/incoming
# ls -ld /home/ftp/incoming
➥drwxr-xr-x 2 ftp ftp 6 2007-06-12
17:16 /home/ftp/incoming/
```

Para que usuários cadastrados no sistema possam utilizar o FTP com seus nomes de usuário e senha, a opção `local_enable=YES` deve estar presente no arquivo `vsftpd.conf`. Caso `chroot_local_users=YES` esteja presente, o diretório raiz enviado ao cliente FTP será o diretório do usuário local. Neste caso, os nomes de usuário presentes no arquivo indicado pela opção `chroot_list_file` não terão um chroot para seus diretórios pessoais. Caso a opção `chroot_local_user` não esteja presente ou seja igual a `no`, este arquivo determinará quais usuários utilizarão o FTP com chroot em seus diretórios pessoais.

2.212.4 Shell seguro (SSH)

Apesar de ser um item fundamental na manutenção e operação segura de computadores remotos, o próprio OpenSSH não é totalmente livre de brechas de segurança. Para limitar ao máximo brechas de segurança, algumas opções devem ser observadas no arquivo `/etc/ssh/sshd_conf`:

```
PermitRootLogin no
```

Ao bloquear o acesso direto ao usuário root, é acrescentada uma segunda camada de segurança, pois somente após um invasor ou mesmo um usuário legítimo conseguir entrar como um usuário comum é que poderá fazer o login como root.

```
Protocol 2
```

O OpenSSH pode trabalhar com o protocolo do tipo 1, menos seguro, e o protocolo do tipo 2, mais seguro. Por isso, é muito recomendável manter apenas o protocolo 2.

```
IgnoreRhosts yes
```

Ignora uma modalidade de acesso legada, na qual bastava que os hosts presentes nos arquivos `~/.rhosts` e `~/.shosts` pudessem entrar sem fornecer senha.

```
X11Forwarding yes
```

Essa opção permite que janelas de programas sejam abertas por meio da conexão SSH. É necessário passar a opção `-X` para o comando `ssh`.

Para evitar que a senha seja necessária em todo login, pode-se criar um arquivo chamado `authorized_keys` para que o `ssh` realize a autenticação de usuário via chave no lugar de senha. O arquivo `authorized_keys` deve existir no host de destino e pode conter uma ou mais chaves que foram criadas no host de origem por meio do comando `ssh-keygen`. Para gerar uma chave *dsa* de 1024 bits:

```
$ ssh-keygen -t dsa -b 1024
```

Esse comando gerará as chaves `id_dsa` e `id_dsa.pub` em `~/.ssh/` no host de origem. O conteúdo de `id_dsa.pub` poderá então ser incluído em `~/.ssh/authorized_keys` para o usuário específico no host de destino. Supondo que 192.168.1.1 é o IP do host de destino e que o usuário possui conta com o mesmo nome ali, o arquivo pode ser copiado com uso do comando:

```
$ cat ~/.ssh/id_dsa.pub | ssh
➥192.168.1.1 'cat >> ~/.ssh/
➥authorized_keys'
```

Por questão de segurança, é importante que todos os arquivos contendo chaves em `/etc/ssh/` e `~/.ssh/` tenham permissão 600 (escrita e leitura apenas para o dono do arquivo).

2.212.5 TCP_wrappers

Daemons de serviços de rede compilados com suporte à biblioteca libwrap podem utilizar-se do mecanismo chamado TCP wrappers para controlar o acesso por hosts na rede. Esse controle é estabelecido por meio de regras criadas nos arquivos `/etc/hosts.allow` e `/etc/hosts.deny`.

O arquivo `/etc/hosts.allow` contém as regras para os hosts que poderão acessar a máquina local. Se um host corresponder a uma regra em `/etc/hosts.allow`, ele será liberado e o arquivo `/etc/hosts.deny` não será consultado.

O arquivo `/etc/hosts.deny` contém as regras para os hosts que não poderão acessar a máquina local. Se um host não constar em `/etc/hosts.allow` nem em `/etc/hosts.deny`, ele será liberado.

Cada regra é escrita em uma linha e o formato é o mesmo tanto para `/etc/hosts.allow` quanto para `/etc/hosts.deny`, sendo serviço um ou mais nomes de daemon de serviço, ou instruções especiais; host é um ou mais endereços ou instruções especiais; e comando é a linha de comando a executar no caso de cumprimento da regra e é opcional.

Hosts podem vir na forma de domínios, IPs de rede ou IPs incompletos. Caracteres curinga "?" e "*" podem ser utilizados.

Instruções especiais são `ALL`, `LOCAL`, `KNOW`, `UNKNOW` e `PARANOID`. O operador `EXCEPT` exclui um host ou grupo de hosts de uma determinada regra.

Em `/etc/hosts.allow`, liberar todos os serviços a todos os hosts da rede `192.168.1.0` com exceção do `192.168.1.20`:

```
ALL: 192.168.1.* EXCEPT
➥192.168.1.20
```

Bloquear todos os serviços a todo host que não constar em regra de `/etc/hosts.allow`, em `/etc/hosts.deny`:

```
ALL: ALL
```

A documentação completa para a criação de regras pode ser encontrada na página manual hosts_access.

Para servidores disparados por meio do daemon super-servidor `inetd`, é necessário especificar a utilização de TCP_wrappers. O arquivo `/etc/inetd.conf` configura o daemon `inetd`. Cada linha corresponde à configuração para um serviço válido que deve constar em `/etc/services`. A linha de configuração contém os seguintes campos:

- *service name*: Nome de um serviço válido em `/etc/services`;
- *socket type*: `stream` se TCP e `dgram` se UDP. Outros valores possíveis são `raw`, `rdm` e `seqpacket`;
- *protocol*: Protocolo válido em `/etc/protocols`, como `tcp` ou `udp`;
- *wait/nowait*: Especifica se o `inetd` deve ou não esperar o programa servidor retornar para

continuar aceitando conexões para o mesmo;
- *user.group*: Rodar o programa servidor como usuário e grupo especificados. Dessa forma, é possível rodar o programa servidor com permissões diferentes de root. O grupo é opcional;
- *server program*: Caminho do programa para executar quando um pedido existir no respectivo socket. Para controle dos pedidos por meio do TCP_wrappers, deve ser `/usr/sbin/tcpd`;
- *server program arguments*: Quando `tcpd` é usado para controlar os pedidos, neste campo deverá constar o caminho para o programa que de fato é o servidor do serviço.

Exemplo de entrada em `/etc/inetd.conf` para o servidor de email pop3:

```
pop3  stream  tcp  nowait  root  /
➥usr/sbin/tcpd  /usr/sbin/popa3d
```

Para desativar o uso de um servidor, basta comentá-lo com o caracter #:

```
# pop3  stream  tcp  nowait  root
➥/usr/sbin/tcpd  /usr/sbin/popa3d
```

Após alterar o arquivo `/etc/inetd.conf`, é necessário fazer com que o daemon `inetd` releia as configurações, o que pode ser feito reiniciando o daemon ou enviando o sinal SIGHUP. O PID para o daemon inetd pode ser consultado por meio do arquivo `/var/run/inetd.pid`.

Versão aprimorada do servidor `inetd`, o super-servidor `xinetd` dispensa a utilização do daemon `tcpd`, pois ele próprio se encarrega de controlar os pedidos. A configuração é feita por meio do arquivo `/etc/xinetd.conf` ou de arquivos correspondentes a cada serviço em `/etc/xinetd.d/`.

Os valores de configuração para cada serviço são como os do `/etc/inetd.conf`, porém o formato do arquivo difere.

Se iniciado com a opção `-inetd_compat`, o `xinetd` adicionalmente usará as configurações em `/etc/inetd.conf` (se existirem).

A estrutura de configuração de um serviço no arquivo `xinetd.conf` é mostrada no **exemplo 2**.

As opções em destaque, `no_access` e `only_from`, podem ser utilizadas adicionalmente ao TCP_wrappers. Os valores podem ser números IP, nomes (resolvidos inversamente a partir do IP de origem), nomes de rede que constem em `/etc/networks` (somente para IPv4) e abreviações endereço/máscara. Outras abreviações possíveis podem ser consultadas na página de manual `xinetd.conf(5)`.

Ao contrário do TCP_wrappers, se um host de origem se enquadrar para ambos, o acesso ao respectivo serviço será bloqueado.

2.212.6 Procedimentos de segurança

Todo administrador deve estar atento a possíveis ataques e invasões na rede. Para isso, operações de monitoramento devem fazer parte da rotina de administração da rede. Além disso, é importante atualizar-se junto a boletins de segurança mantidos por entidades especializadas.

Entidades tradicionais que emitem boletins de segurança freqüentes e relevantes são a lista Bugtraq (`www.securityfocus.com/archive/1`), atualmente mantida pelo Security Focus, CERT (`www.cert.org`) e CIAC (`www.ciac.org`). Nestes sites, é possível tomar conhecimento de falhas em programas antes que estas possam causar dano.

Kerberos

O método tradicional de controle de acesso é a utilização de senhas por parte do usuário. Apesar de ser o mais utilizado, existem algumas inconveniências, como perda e roubo das senhas. Parte da segurança fica sob responsabilidade do usuário, que nem sempre é zeloso e pode comprometer a segurança de toda a rede.

Uma alternativa possível em alguns casos é a utilização do protocolo *Kerberos*. O Kerberos é um serviço de autenticação que consiste numa terceira entidade confiável (KDC – *Key Distribution Center*) que comprova a identidade de ambas as pontas (cliente e servidor) antes de estabelecerem a comunicação. Vários servidores são compatíveis com Kerberos. Alguns deles são o Apache, OpenSSH, NFS (NFSv3), Samba (versão 3 ou superior) etc.

A autenticação via Kerberos funciona de maneira semelhante a um certificado SSL de um servidor Web. O AS (*Authentication Server*) age como uma autoridade certificadora,

Exemplo 2: Estrutura de serviço no xinetd.conf

```
nome do serviço {
    disable = yes/no
    socket_type = stream,dgram,raw,rdm ou seqpacket
    protocol = Protocolo válido em /etc/protocols
    wait = yes/no
    user = Usuário de início do servidor
    group = Grupo de início do servidor
    server = Caminho para o programa servidor do serviço solicitado
    no_access = 192.168.1.0
    only_from = 0.0.0.0
```

que informa tanto ao servidor quanto ao cliente se estes são quem eles informam ser. Toda a autenticação é mediada pelo servidor de autenticação, que emite a permissão para as partes na forma de tickets. Esses tickets podem ser comparados aos certificados SSL e possuem restrições ao tipo de serviços que podem ser acessados, além de prazo de validade.

Outro recurso muito popular é utilizar um IDS (*Intrusion Detection System*). Em linhas gerais, um IDS monitora atividades suspeitas na rede e toma procedimentos caso as identifique.

Tripwire, nessus, snort e PortSentry

Ferramentas de detecção de intrusos, conhecidas como IDS, podem ser tão úteis quanto difíceis de utilizar. Para fins de preparação para a prova 202, basta que conheçamos conceitualmente algumas delas.

O Tripwire é uma ferramenta cuja função básica é monitorar mudanças em arquivos do sistema. Semelhante à verificação que pode ser feita em arquivos instalados com o RPM, o Tripware gera um relatório sobre os arquivos importantes do sistema e é capaz de avisar caso alguma mudança ocorra com eles. Porém, erros mais profundos, como no setor de boot, não podem ser detectados.

O Nessus é um daemon que verifica diversas brechas de segurança no sistema, como serviços mal configurados, software desatualizado, senhas fracas e portas abertas. Já o Snort e o PortSentry são IDS baseados no monitoramento de portas. Eles podem identificar comportamentos suspeitos e informar o administrador. Ainda, a atividade suspeita pode ser totalmente bloqueada.

Considerações sobre o tópico

É importante sobretudo conhecer muito bem a utilização do Open SSH e seu arquivo de configuração, mas sem negligenciar os outros temas. ■

Sobre o autor

Luciano Antonio Siqueira é editor e desenvolvedor da Linux New Media do Brasil. Escreveu os livros Certificação LPI-1, Certificação LPI-2 e outros títulos. Trabalha com Linux há mais de dez anos e é formado em psicologia pela Universidade Estadual Paulista.

TUDO SOBRE COMUNICAÇÃO IP

www.ipcomm2008.com.br

Agende!

IP COMMUNICATIONS BRASIL 2008

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 | Asterisk Brasil.org

REALIZAÇÃO | IP COMMUNICATIONS BRASIL

ORGANIZAÇÃO | Comunicação e Marketing

Análise do appliance VoIP Grandstream GXE5024

Poderosa caixinha

Instalar uma distribuição Linux e o Asterisk num servidor é uma solução comum, mas talvez valha a pena substituí-la por um appliance dedicado poderoso e flexível.

por Pablo Hess

Os benefícios do VoIP são claros para todos: custos reduzidos de chamadas e do equipamento, além de uma flexibilidade muito superior na administração da telefonia da empresa ou da casa. Com um PBX digital como o famoso e quase onipresente *Asterisk*, basta um PC comum para desfrutar de recursos como conferência, secretária eletrônica com senha, agendamento de ligações, unidade de resposta automática e muito mais, a um custo significativamente inferior ao de soluções tradicionais de PBX.

No entanto, supondo um cenário de uma pequena empresa com 20 funcionários no escritório, o número de telefones IP, ATAs ou placas FXS/FXO pode aumentar esse custo de forma indesejável. Além disso, o consumo de energia de um PC comum, somado ao de todos esses dispositivos acessórios, também deixa a solução inteira menos convidativa.

Uma saída que vem ganhando muitos adeptos é o uso de aparelhos dedicados a essa funcionalidade, os famosos *appliances*. No campo do VoIP, o uso desses aparelhos pode representar importantes economias sem que se precise abrir mão da flexibilidade e do poder de um PBX digital como o Asterisk.

No mercado brasileiro, em que o fator custo freqüentemente tem peso maior que a flexibilidade ou as funcionalidades, o Grandstream GXE5024[1] é uma nova opção de PBX IP integrado com grande potencial. Este artigo analisa o aparelho da empresa americana, que funcionou com perfeição no ambiente Linux usado para os testes da **Linux Magazine**.

Híbrido

Um dos primeiros aspectos que chamam atenção no GXE5024 são suas quatro portas FXO e duas FXS (**figura 1**), o que significa que é possível conectar até quatro linhas analógicas ao aparelho. Isso gera inúmeras possibilidades de uso híbrido do dispositivo. Por exemplo, é possível utilizar as linhas analógicas em regime de *fail-over* para manter as comunicações caso seu provedor de acesso à Internet entre em colapso (ou alguém desavisadamente chute a tomada do modem, evidentemente). Alternativamente, pode-se atribuir números de telefone específicos que devem sempre ser chamados pelas linhas telefônicas legadas.

Figura 1 No painel traseiro do aparelho, as portas *Ethernet*, FXO, FXS e USB.

Energia

Quanto à energia, o dispositivo oferece suporte a PoE (*Power over Ethernet*), o que significa que ele pode funcionar também como fonte de energia para dispositivos compatíveis com essa tecnologia. Em sua operação normal, trata-se também de um dispositivo econômico, especialmente quando comparado a um servidor dedicado – como se espera de qualquer appliance.

Instalação

Com 17,5 x 27,0 x 4,0 centímetros, o GXE5024 é um pouco menor que um switch Ethernet comum de 16 portas. Como se percebe, o local mais apropriado para instalação do dispositivo não é no rack do datacenter, mas sobre uma mesa – bem de acordo com o público-alvo especificado (pequenas empresas e usuários domésticos).

Na rede, sua localização ideal é entre o switch e a WAN, pois o aparelho possui funcionalidade de roteador, incluindo *port forwarding* e um servidor DHCP embutido.

Operação e configuração

A interface de administração do dispositivo é extremamente funcional. Não possui recursos AJAX avançados, mas oferece a tranqüilidade de

Figura 2 Menu simples.

sabermos que se trata de um sistema Linux embarcado.

Seguindo a estrutura bastante clara do menu (**figura 2**), pode-se começar a configuração do aparelho pelo item *System Configuration | Network Setting* (**figura 3**). Nele é feita a configuração da LAN, do servidor DHCP e da WAN. Nesse ponto, um destaque: o dispositivo tem suporte a PPPoE, uma ótima notícia para quem desejar ligá-lo diretamente ao modem ADSL. Recursos como VPN (PPTP, IPsec e L2TP), *Telnet*, DNS dinâmico e port forwarding completam o conjunto.

Em seguida, no item *System Configuration | Route*, é possível definir rotas para máquinas ou redes específicas, enquanto *System Setting* permite a configuração de diversos parâmetros específicos de VoIP, como porta UDP SIP e servidores STUN, SMTP – o dispositivo pode enviar emails informando ao administrador aspectos importantes do sistema – e NTP, entre outros. Também é nesse ponto que são definidas a fonte da música do sistema (entrada auxiliar ou um diretório com arquivos de música) e as cotas de uso do espaço de armazenamento por categoria de usuário (**figura 4**).

O restante das configurações sob o menu *System Configuration* seguem com a mesma facilidade, incluindo o uso de um servidor *Syslog* remoto.

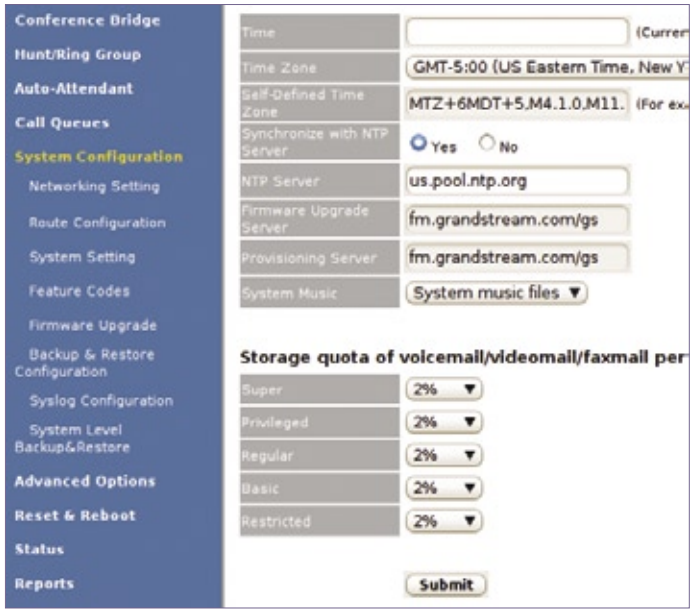


Figura 4 O dispositivo permite a definição de cotas de armazenamento por categoria de usuário.

Extensões

A configuração de extensões é um dos pontos altos da interface web. É possível adicionar cada extensão individualmente ou fazê-lo em lote (**figura 5**) na opção *Batch Add*, no menu *Phone Extensions | Extensions Directory*. Cada extensão possui um perfil de privilégios, o que facilita o gerenciamento simultâneo de grande número de extensões.

Já os comandos de operação, como o número do correio de voz e o resgate de chamadas, são definidos no menu *System Configuration | Feature Codes*.

Terminais

É bastante impressionante a profusão de opções disponíveis para configuração dos terminais FXO, como impedância do terminal elétrico, por exemplo. Felizmente, os valores padrão são suficientemente bons para a maioria das instalações. Tom, volume e duração dos tons DTMF e dos sinais de discagem e de ocupado também podem ser alterados, mas é importante ter cuidado para não fugir demais dos padrões aos quais todos os funcionários já estão habituados.

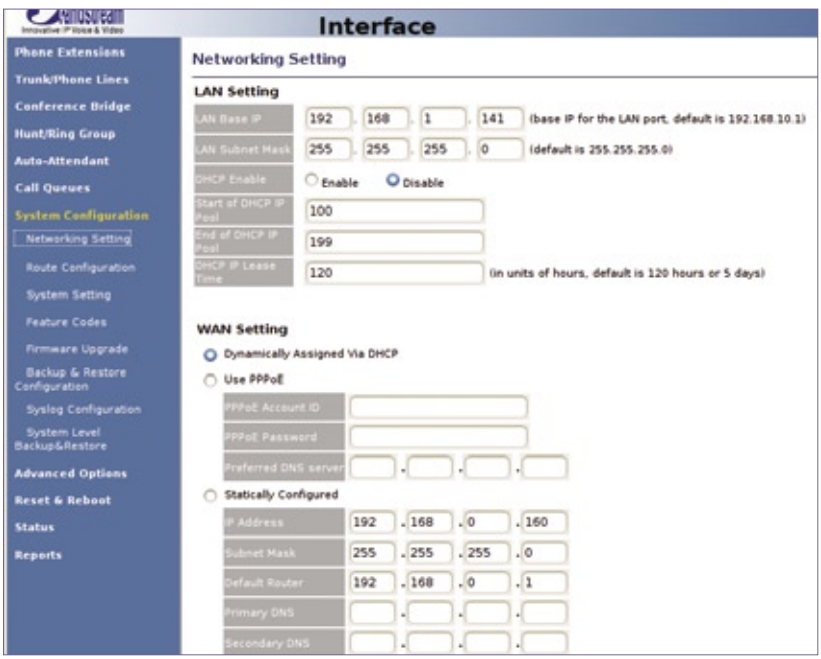


Figura 3 É muito simples especificar as configurações de rede do GXE5024.

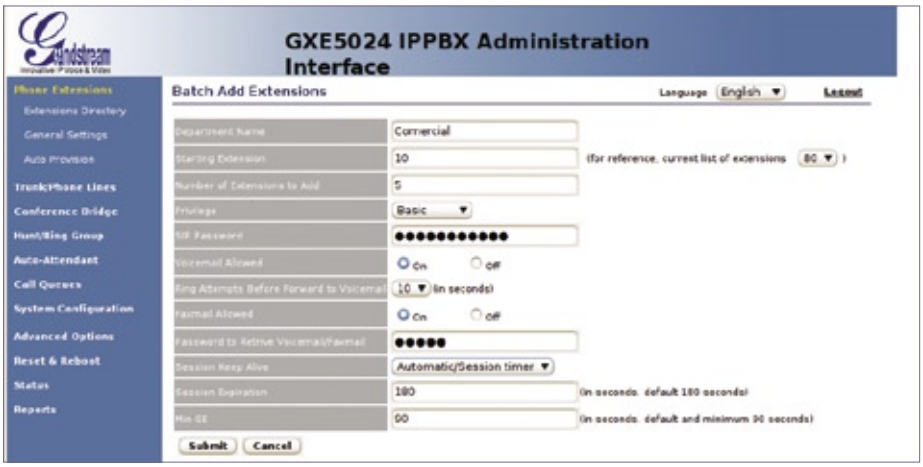


Figura 5 Criação de diversas extensões com a funcionalidade *Batch Add*.

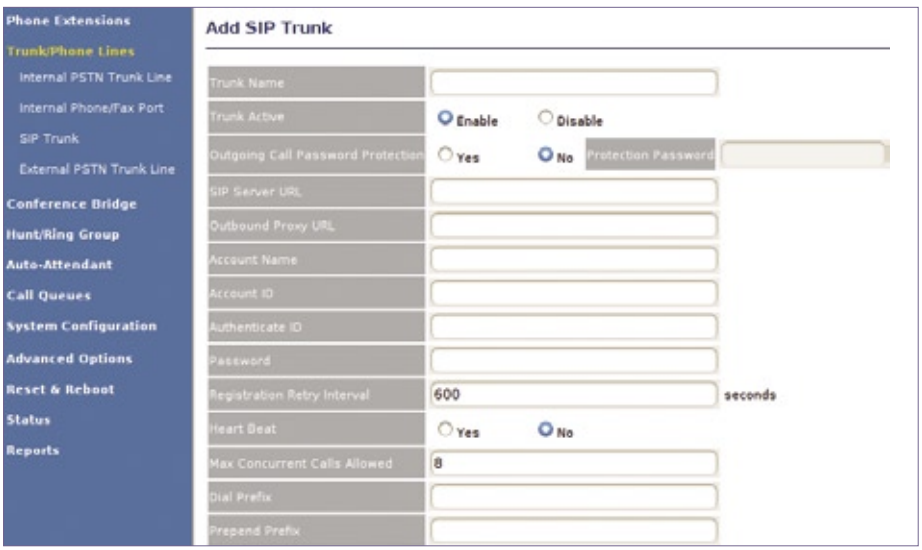


Figura 6 Configuração do tronco SIP.

Troncos

Uma das configurações mais importantes do servidor VoIP é a do tronco SIP – um dos motivos de sua adoção, em última análise –, no menu *Trunk/Phone Lines | SIP Trunk* (**figura 6**). Como de costume, estão presentes as opções mais diversas, com padrões sensatos sempre que possível.

No item *Trunk/Phone Lines | External PSTN Trunk Line*, configura-se também o tronco PSTN legado.

Recursos

O GXE5024 disponibiliza duas salas de conferência (**figura 7**), cada uma com sua senha para melhor controle dos participantes.

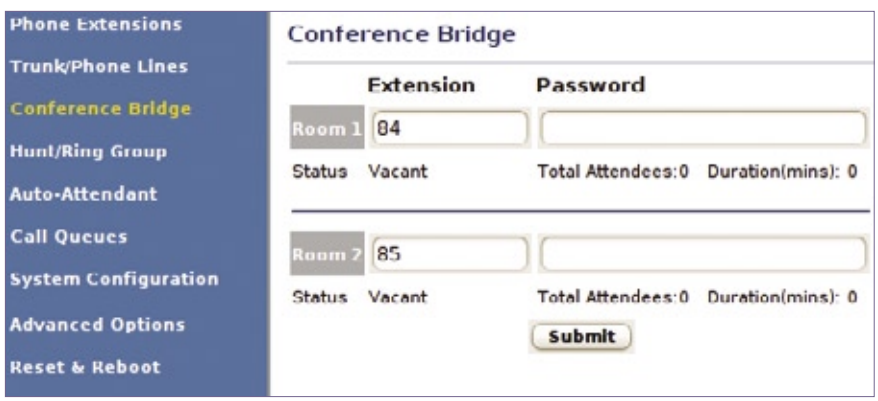


Figura 7 Duas salas de conferência são disponibilizadas pelo GXE5024.

Também impressionantes são os recursos de fax do aparelho, que permitem a conversão de documentos recebidos para PDF e seu envio para um ou vários endereços de email.

Conclusões

Por R$ 2.388, o GXE5024 oferece uma grande diversidade de recursos úteis a qualquer empresa. Os fatores que limitam seu uso em cenários e empresas maiores são principalmente o número de conexões FXO (o modelo GXE5028 possui oito) e, em maior grau, o hardware embarcado. Embora seja usado com responsabilidade por um sistema Linux, todo hardware tem limites.

O suporte a todos os principais padrões de áudio e vídeo são características já esperadas em qualquer appliance VoIP da atualidade e, portanto, não chamam especial atenção. Em contrapartida, a capacidade de armazenar toda a configuração do dispositivo num arquivo que pode ser recuperado a qualquer momento e armazenado numa mídia USB ou no computador do administrador é uma vantagem significativa, assim como os belos gráficos de relatórios e estatísticas de chamadas e do sistema. ■

Mais informações

[1] Linha GXE502x: http://www.grandstream.com/gxe502x.html

LATINOWARE 2008

V Conferência Latino-Americana de Software Livre

30 | outubro a 01 | novembro | 2008

atrações

- Mesas redondas
- Palestras
- 4ª Olimpíada de Robótica Livre
- Chamada de Trabalhos
- Minicursos
- Maratona de Tradução
- Exposição
- Latinoware Kids
- Prêmio Latinoware de Software Livre

PARQUE TECNOLÓGICO ITAIPU
FOZ DO IGUAÇU PR

O objetivo da Latinoware é abrir espaço para discussões e reflexões sobre a utilização de programas de código aberto em todas as áreas do conhecimento.

Aberto à comunidade, usuários, desenvolvedores, estudantes, profissionais da área pública e privada, e a todos que queiram contribuir com a expansão do conhecimento para o desenvolvimento econômico e social auto-sustentado do continente.

CELEPAR INFORMÁTICA do PARANÁ

SERPRO Soluções para um Brasil de Todos

PTI Parque Tecnológico Itaipu

ITAIPU BINACIONAL

A liberdade da informação passa por aqui

www.latinoware.org

Análise de logs com o plugin check_logfiles do Nagios

Viajante dos logs

O plugin do Nagios check_logfiles ajuda a monitorar arquivos de log – mesmo quando há rodízio e mudanças de nome.

por Gerhard Lausser

A ferramenta de monitoramento *Nagios* é, na realidade, uma plataforma bastante genérica para observação. Ele permite o monitoramento de computadores, processos, dispositivos e serviços de rede. Outra coisa que ele pode acompanhar são arquivos de log. Os plugins *check_log* e *check_log2*, por exemplo, são populares com muitos administradores; entretanto, esses plugins às vezes têm problemas em situações nas quais um aplicativo ou script esteja fazendo o rodízio dos logs. As ferramentas tendem a escorregar ocasionalmente e perder algumas linhas, o que não se pode permitir quando se precisa de 100% de cobertura.

Para cobrir essas falhas, o plugin *check_logfiles*[1] foi desenvolvido para verificar cada uma das entradas – mesmo que um log seja movido, mude de nome ou desapareça num arquivo compactado durante o período de monitoramento.

Quadro 1: Opções de configuração

Na execução do `configure`, antes da instalação, há algumas opções importantes para regular o funcionamento do *check_logfiles:*

- `--with-perl` deve ser usado caso já exista um interpretador *Perl* instalado no sistema.
- `--prefix` especifica o diretório *home* da instalação do *Nagios*. O plugin é instalado no subdiretório `libexec/`.
- `--with-seekfiles-dir` especifica o diretório onde serão armazenadas as informações de estado entre as execuções do programa.

Mas isso não é tudo: vários outros recursos sofisticados fazem esse plugin se destacar de seus antecessores. Por exemplo, o check_logfiles consegue lidar com múltiplas chaves de busca, com exceções que identificam um subconjunto especial de uma chave de busca como inofensiva, aplicar limites que disparam alertas após um número mínimo de ocorrências e integrar programas externos.

Este artigo mostrará como começar a monitorar arquivos de log com o plugin do Nagios check_logfiles. Para começar, vamos supor que o leitor tenha um conhecimento básico do Nagios. Quem estiver em busca desses conhecimentos pode consultar a edição 31 da **Linux Magazine**[2], que teve como tema de capa justamente o monitoramento de redes com Nagios.

Instalação

O plugin check_logfiles está disponível como um tarball[1]. Após ser baixado e descompactado, basta entrar no diretório `check_logfiles-2.3.1.1/` e seguir os passos padrão de instalação: `configure; make; make install`. Há várias opções disponíveis na etapa do `configure` (veja o **quadro 1**).

Primeiro caso

Com o plugin instalado e configurado, já é possível usá-lo para monitorar logs.

Para um primeiro contato com o plugin em ação, considere o exemplo a seguir, que realiza uma busca simples pelo texto `BIGERROR` num arquivo chamado `rhubarbomat.log`. A chamada ao plugin é semelhante a:

```
check_logfiles \
-criticalpattern='BIGERROR' \
-logfile=rhubarbomat.log
```

Se a string *BIGERROR* ocorrer em uma linha que tenha sido adicionada após a última execução do check_logfiles, o plugin retorna um status *CRITICAL*; caso contrário, ele retorna *OK*. O texto é, na verdade, uma expressão regular.

Em vez de `-criticalpattern`, seria possível usar `-warningpatter`, como em `--warningpattern='SMALLERROR'`. O código de saída para uma busca com êxito é *1* para *WARNING*. Obviamente, nada impede o uso das duas opções ao mesmo tempo.

Esse primeiro exemplo não leva em consideração o rodízio de logs. Embora o plugin identifique a chave de busca, ele não procuraria em arquivos de log que tivessem sido rotacionados, atuando apenas no arquivo mais recente.

Para permitir que a busca cubra logs rotacionados entre duas chamadas ao check_logfiles e, portanto, para evitar descontinuidades, o plugin precisa de uma dica de onde encontrar os arquivos mais antigos.

O parâmetro que lida com isso é o `-rotation`, que passa o nome do novo arquivo ou contém uma expressão regular que coincide com os nomes de arquivos rotacionados (**figura 1**). Primeiro, suponha que o

arquivo `rhubarbomat.log` seja automaticamente renomeado para `rhubarbomat.log.0` diariamente e que seja criado um arquivo vazio `rhubarbomat.log`. Depois disso, o antigo arquivo `rhubarbomat.log.0` é renomeado para `rhubarbomat.log.1`, o `rhubarbomat.log.1` vira `rhubarbomat.log.2` e assim por diante. Nesse caso, o parâmetro `-logfile=/var/log/rhubarbomat.log -rotation='rhubarbomat\.log\.\d+'` faria o plugin encontrar e efetuar buscas nas versões anteriores do log. Como alternativa, pode-se especificar explicitamente o nome de arquivo `rhubarbomat.log.0`.

O plugin check_logfiles investiga somente as linhas do arquivo que tenham sido alteradas desde sua última execução, o que significa que reexecuções imediatas produzirão resultados diferentes. Após retornar um resultado *CRITICAL*, a próxima chamada produz *OK*, como mostra o **exemplo 1**. Uma definição de serviço para o Nagios é fornecida no **exemplo 2**.

Arquivo de configuração

O check_logfiles realmente se destaca quando é usado um arquivo de configuração em vez de parâmetros de linha de comando. O exemplo anterior precisaria do seguinte arquivo de configuração:

```
$ cat rhubarb.cfg
@searches = ({
  tag => '0815',
  logfiles => '/var/log/
➥rhubarbomat.log',
  criticalpatterns =>
➥ '.*0815.*',
  rotation => 'loglog0log1',
  options => 'noprotocol'
});
```

O plugin é executado por uma linha de comando como `check_logfiles -f nome_do_arquivo`. É fácil ver que o arquivo de configuração é composto por código Perl. Os elementos no vetor `@searches` (referido apenas como "a busca" daqui em diante) são referências a *hashes* que combinam o log e a chave de busca. A tag é um identificador único para essa combinação. O plugin precisa disso para evitar ambigüidades nos arquivos que armazenam as informações de status para a próxima execução do check_logfiles. Um vetor possibilita a busca em múltiplos arquivos de log com uma única chamada ao plugin.

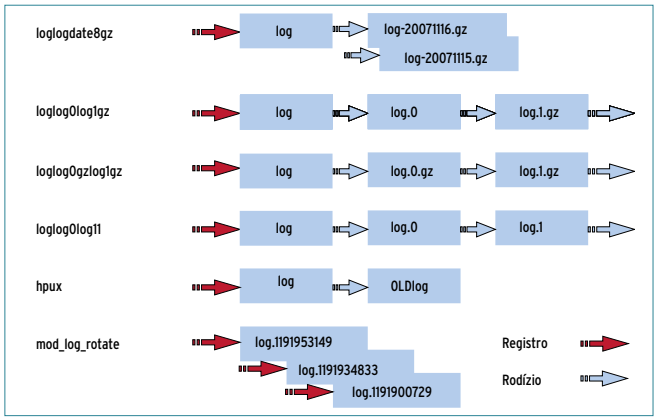


Figura 1 O rodízio de arquivos de log dificulta o trabalho de alguns plugins. As entradas podem acabar sendo perdidas em virtude das mudanças de nomes de arquivos.

O código em Perl para essa configuração é mostrado no **exemplo 3**.

Por outro lado, se for desejável que o plugin soe um alarme caso uma chave de busca esteja ausente do log, o padrão de busca precisa iniciar com um ponto de exclamação (!). A seguinte sintaxe informa se o becape da noite passada foi completado sem erro:

```
criticalpatterns => '!backup
➥ successful']
```

Além disso, é possível definir exceções que se parecem com a mensagem buscada, mas representam um caso especial:

Exemplo 1: Execuções recorrentes

```
01 $ logger "test1 this is 0815"
02 $ logger "this isn't because its 0916"
03 $
04 $ check_logfiles -logfile=/var/log/ rhubarbomat.log --tag=0815
➥-criticalpattern= '.*0815.*' --rotation='loglog0log1'
05
06 CRITICAL - (1 errors in check_logfiles. protocol-2007-10-10-15-10-02) -
Oct 10 15:09:56 localhost lausser: test1 this is 0815 |0815_ lines=2
07 0815_warnings=0 0815_criticals=1 0815_unknowns=0
08 $
09 $ echo $?
10 2
11 $
12 $ logger "rhubarb"
13 $ check_logfiles -logfile=/var/log/ rhubarbomat.log --tag=0815
➥-criticalpattern= '.*0815.*' --rotation='loglog0log1'
14
15 OK - no errors or warnings |0815_lines=1 0815_warnings=0 0815_criticals=0
16 0815_unknowns=0
17 $ echo $?
18 0
```

```
criticalpatterns => [‘SCSI Error’].
criticalexceptions => [‘SCSI
➥Error. *disk0 .*’],
```

Essas entradas soariam o alarme do Nagios para a linha:

```
SCSI Error /dev/disk5 I/O Timeout
```

mas fariam o plugin ignorar a linha:

```
SCSI Error /dev/disk0 I/O Timeout
```

Exemplo 2: Definição de serviço

```
01 define service {
02  service_description check_
➥0815msgs
03  host_name logserver
04  max_check_attempts 1
05  is_volatile 1
06  check_command
07  check_logfiles_critical!0815!/
➥var/log/ rhubarbomat.
➥log!loglog0log1!.*0815.*
08 }
09 define command {
10  command_name check_logfiles_
➥critical
11  command_line $USER1$/check_
➥logfiles $$
12  --logfile=”$ARG2$”
13  --criticalpattern=”$ARG4$”
➥--tag=”$ARG1$” $$
14  --rotation=”$ARG3$”
15 }
```

Exemplo 3: Exemplo de configuração

```
01 @searches = (
02  {
03   tag => ‘lamp-apache’
04   logfile => ‘/var/log/apache/
➥error.log’,
05   criticalpatterns => [‘.*error.*,
➥‘.*fatal.’],
06   rotation => ‘solaris’
07  },
08  {
09   tag => ‘lamp-mysql’,
10   logfile => ‘/var/log/mysql.log’,
11   criticalpatterns =>
➥[‘corruption’,’you hit a bug’]
12  }
```

Rodízio

No final de cada rodada, o plugin guarda a última posição lida no log, juntamente com a data de alteração e o número do inode do arquivo. Essa informação é armazenada no que se conhece como arquivo de busca, ou *seek file*. O check_logfiles gera o nome do arquivo de busca a partir do nome do arquivo de log e do dia.

Na próxima vez que o plugin for executado, ele comparará esses dados com as propriedades do arquivo de log atual e verificará se o arquivo foi expandido, deletado, rotacionado ou criado como um arquivo novo.

Dependendo do intervalo desde a última execução do plugin, vários rodízios podem ter ocorrido. O plugin usa a hora e o parâmetro `rotation` para encontrar arquivos correscpondentes.

Na maioria dos casos, o arquivo de log tem apenas algumas linhas novas. O check_logfiles continua na posição que marcou no fim da última execução e lê as linhas seguintes até chegar ao fim do arquivo. Esse princípio oferece ao plugin uma enorme vantagem de velocidade quando comparado a outras técnicas que se baseiam no *diff* para precisar as diferenças entre o arquivo de log atual e uma cópia gravada, principalmente no caso de crescimento rápido do arquivo.

Digitando `./configure with-seek-file-dir`, pode-se especificar um diretório para os arquivos de busca, ou pode-se alterar o caminho dinamicamente com a variável `$seekfilesdir` no arquivo de configuração.

O plugin usa `/tmp/` por padrão; porém, é bom alterar isso para `/var/tmp/`, pois alguns sistemas operacionais não mantêm o conteúdo de `/tmp/` após a reinicialização.

Tipos

Além do parâmetro `rotation`, é possível procurar no parâmetro `type`, que especifica o tipo de arquivo de log. Se existir o parâmetro de rodízio, `type` supõe um valor para `rotation`. Isso significa que os arquivos compactados são relevantes para a busca. Se o parâmetro de rotação não existir, `type` supõe o valor `simple`. Essa configuração faz sentido se um aplicativo gerar continuamente novos arquivos de log e apagar os arquivos existentes, ou se o administrador estiver preparado para aceitar o fato de que as últimas poucas linhas de um arquivo de log rotatório não serão levadas em consideração.

Exemplo 4: Pesquisa em tipos de log virtual

```
01 @searches = (
02 {
03  tag => ‘host0’,
04  logfile => ‘/sys/class/scsi_host/host0/state’,
05  type => “virtual”,
06  criticalpatterns => [
07   ‘Link [^Up]+’ # Soar o alarme se “Link Up” ausente
08  ],
09  options => ‘noprotocol’,
10 },
11 );
```

O log `virtual` é outro tipo em que o check_logfiles buscará. Esse tipo é usado, por exemplo, para o sistema de arquivos `/proc/` em máquinas Linux (e oferece a opção de configurar facilmente o monitoramento de hardware).

Os arquivos desse sistema de arquivos não crescem; em vez disso, é preciso tratá-los como se tivessem sido criados imediatamente antes de serem lidos. O plugin sempre investiga esses arquivos a partir da primeira linha (veja o **exemplo 4**).

Além disso, o tipo `errpt` procura o relatório de erros do AIX. Isso informa ao plugin que ele deve procurar padrões na saída do comando `errpt`, exatamente como se fosse um arquivo de log normal. O tipo `psloglist` ainda é experimental; ele permite que o plugin faça buscas no log de eventos de máquinas Windows.

Parâmetros de busca

Vários parâmetros estão disponíveis para padrões para buscas nos logs. Os parâmetros mais importantes estão listados no **quadro 2**, enquanto que os parâmetros globais são descritos no **quadro 3**. Uma lista completa de todos os parâmetros possíveis pode ser encontrada online[1].

Os parâmetros são usados no arquivo de configuração conforme mostrado no trecho do **exemplo 5**.

Exemplo 5: Parâmetros do arquivo de configuração

```
01 $ cat rhubarb.cfg
02 @searches = ({
03  tag => '0815',
04  logfile => '/var/log/rhubarbomat.log',
05  archivedir => '/var/log/archives',
06  rotation => 'loglog0gzlog1gz',
07  criticalpatterns => '.*0815.*',
08  criticalexceptions => '.*0815 macht aber nix.*',
09  warningpatterns => ['.*failure.*','!successful'],
10  warningthreshold => 10,
11  okpatterns => '.*cleared.*',
12  options => 'case,noprotocol,script'
13  script => 'restart_rhubarbomat'
14 });
```

Saída e desempenho

A saída do check_logfiles contém referências às descobertas, como por exemplo:

```
CRITICAL - (3 errors in check_
➥logfiles.protocol-2007-10-10-16-
➥21-09) InnoDB: Database pae
➥corruption on disk or a failed
➥...|mysql_lines=12 mysql_
➥warnings=0 mysql_criticals=3
➥mysql_unknowns=0
```

Além do código típico de saída do Nagios, pode-se ver três pontos (`...`), o que indica que há mais linhas coincidentes.

Para cada busca ou dia, o plugin também retorna um conjunto de quatro estatísticas de desempenho:

- `_lines`: o número de linhas buscadas no log.

Quadro 2: Parâmetros de busca

Uma busca eficiente nos arquivos de log depende do bom uso dos parâmetros disponíveis. Conheça todos eles:

- `tag` Um curto descritor único para a busca.
- `logfile` Nome do arquivo de log a ser varrido.
- `archivedir` Diretório com os logs rotacionados.
- `rotation` Expressão regular usada para localizar arquivos de log rotacionados. Há valores pré-definidos para os padrões mais comuns.
- `criticalpatterns` Um único padrão que o plugin busca no log. Se for necessário presquisar múltiplos padrões em uma categoria, é preciso especificá-los como elementos de um vetor (veja o **exemplo 6**).
- `criticalexceptions` Suporta uma especificação de padrões mais granular: o parâmetro ignora exceções que não sejam contabilizadas como erros.
- `warningthreshold` Limites (*thresholds*) são usados sempre que se deseja contar um certo número de ocorrências antes de emitir um alerta:
- `warningthreshold=>n` Significa que somente uma ocorrência será contabilizada a cada *n*.
- `okpatterns` Reinicia o contador e deleta todas as ocorrências dos tipos *warning* e *critical* encontradas anteriormente.
- `nologfilenocry` Ignora arquivos de log inexistentes; caso contrário, se o arquivo não estiver presente, o plugin retorna um status *UNKNOWN* (desconhecido).
- `syslogserver` Se o log contiver mensagens de múltiplos servidores, o plugin usa essa opção para procurar somente as mensagens vindas da máquina local.
- `syslogclient=nome` O mesmo que a opção acima, mas procura apenas em mensagens de um cliente específico. Essa opção é interessante para servidores *Syslog* centralizados.
- `nocase` Ignora a caixa nas expressões regulares.
- `options` Múltiplas opções separadas por vírgulas oferecem um controle mais fino sobre as ações do plugin. Um prefixo `no` inverte o significado da opção.
- `nocase` Torna os padrões insensíveis à caixa.
- `noprotocol` Impede que o plugin crie um arquivo de protocolo. Normalmente, qualquer linha do log que contenha o padrão de busca é escrita no arquivo de protocolo. Essa opção economiza processamento trabalhoso em caso de alertas.

Exemplo 6: Busca de múltiplos padrões

```
01 @searches = ({
02   tag => 'minor_errors',
03   type => 'errpt',
04   criticalpatterns =>
➥['ADAPTER ERROR',
05    'The largest dump device
➥is too small.',
06    'The copy directory is too
➥small.',
07    'Kernel heap use exceeds
➥allocation count',
08    'Kernel heap use exceeds
➥percentage thres',
09    'LINK ERROR',
10    'SCSI BUS OR DEVICE ERROR',
11    'SCSI DEVICE OR MEDIA
➥ERROR',
12    'Possible malfunction on
➥local adapter',
13    'ETHERNET DOWN',
14    'UNABLE TO ALLOCATE SPACE
➥IN KERNEL HEAP'
15   ],});
```

- `_warnings`: o número de linhas que contêm padrões de alerta.
- `_criticals`: o número de linhas que contêm padrões críticos.

Esses parâmetros oferecem uma rápida indicação da densidade do problema.

Ações

A opção `script` pode executar um programa no caso de uma coincidência específica:

```
script => 'nome_do_programa'
```

ou na versão mais recente:

```
script => sub { código_em_perl }
```

As ações incluem reiniciar um aplicativo ou enviar *traps SNMP* e mensagens NSCA. Isso significa que o check_logfiles pode rodar como aplicativo *standalone* sem depender do tratamento de eventos do Nagios.

Os parâmetros `scriptparams` e `scriptstdin` permitem que os usuários rodem scripts externos com parâmetros de linha de comando – e até passem a entrada a partir de STDIN. O **exemplo 7** ilustra essa possibilidade.

No **exemplo 7**, sempre que aparecer uma mensagem de erro numa linha do arquivo `messages`, a linha é enviada para o servidor Nagios com o comando `send_nsca` como resultado passivo de um serviço.

No caso mais simples, o código de saída retornado pelo script externo será irrelevante e não influenciará o código de saída do check_logfiles. Por exemplo, mesmo que a aplicação *Rhubarbomat* do **exemplo 5** reinicie com sucesso, o check_logfiles ainda retornará um estado *CRITICAL* para o Nagios.

A opção `smartscript` passa o código de saída do script externo para o check_logfile. O plugin age como se tivesse descoberto outra linha após aquela que o disparou, o que permite o disparo de um erro,

Exemplo 7: Execução de scripts

```
01 $scriptpath = '/usr/bin/nagios/libexec: /usr/local/nagios/contrib';
02 $MACROS = {
03   CL_NSCA_HOST_ADDRESS => "lpmon1.muc",
04   CL_NSCA_PORT => 5778
05 };
06
07 @searches =(
08   {
09     tag => 'rhubarb',
10     logfile => '/var/log/rhubarbomat.log',
11     criticalpatterns => ['ERROR', 'crashed'],
12     script => 'reinicia_rhubarbomat',
13     scriptparams => '--rhubarbprefix=bla',
14     options => 'script'
15   },
16   {
17     tag => 'san',
18     logfile => '/var/adm/messages',
19     criticalpatterns => [
20       'Link Down Event received',
21       'Loop OFFLINE',
22       'fctl:.*disappeared from fabric',
23       '.*Lun.*disappeared.*'
24     ],
25     options => 'script',
26     script => 'send_nsca',
27     scriptparams => '-H $CL_NSCA_HOST_ADDRESS$ -p $CL_NSCA_PORT$
➥-to $CL_NSCA_TO_SEC$ -c $CL_NSCA_CONFIG_FILE$',
28     scriptstdin => '$CL_HOSTNAME$\t$CL_SERVICEDESC$\t$CL_
➥SERVICESTATEID$\t$CL_SERVICEOUTPUT$\n',
29 });
```

Quadro 3: Parâmetros globais

Além dos parâmetros relacionados a uma única entrada de busca, outras variáveis globais são lidas por todas as buscas e definem o comportamento do plugin, independentemente de uma busca individual.

- `$seekfilesdir` Especifica o diretório onde os arquivos com informações de status serão salvos.
- `$scriptpath` Uma lista de caminhos que o plugin varre em busca de plugins externos, que ele ativa com o parâmetro `script`.
- `$prescript` Especifica um programa externo para ser executado no início.
- `$postscript` Especifica um programa externo para ser executado antes do término.
- `$protocolsdir` Diretório onde o *check_logfiles* guarda os arquivos de protocolo com as linhas retornadas nas buscas.

Exemplo 8: Script supersmart

```
01 @searches =(
02 {
03   tag => 'rhubarb',
04   logfile => '/var/log/rhubarbomat.log',
05   criticalpatterns => ['ERROR', 'crashed'],
06   script => sub {
07     if (`reinicia_rhubarbomat` =~ /successful/) {
08         if ($ENV{CHECK_LOGFILES_SERVICEOUTPUT} =~ / ERROR/) {
09           printf "OK - rhubarbomat reiniciado\n";
10           return 0;
11         } else {
12         printf "WARNING - rhubarbomat travado reiniciado\n";
13         return 1;
14         }
15     } else {
16     printf "CRITICAL - impossivel reiniciar rhubarbomat\n";
17     return 2;
18     }
19   },
20   options => 'supersmartscript'
21 },
```

mas não a reversão da mensagem original do log ou a reavaliação da mensagem.

A terceira opção é `supersmart script`. Scripts desse tipo sobrescrevem a entrada coincidente no arquivo de log com seus códigos e textos de saída em vez de adicionar uma entrada. Diversas variáveis de ambiente estão disponíveis para esses scripts:

- `CHECK_LOGFILES_SERVICEOUTPUT` – conteúdo da linha de ativação
- `CHECK_LOGFILES_SERVICESTATE` – `WARNING`, `CRITICAL` ou `UNKNOWN`
- `CHECK_LOGFILES_SERVICESTATEID` – `1`, `2` ou `3`

Com o uso dessa informação e outros dados – assim como a hora do dia ou os resultados do reinício do aplicativo – , a mensagem de erro depois pode ser reavaliada. Isso permite que o verificador de arquivo de log troque um status *CRITICAL* para *WARNING* ou retorne um código de saída de *o* e, portanto, cancele o alerta. O **exemplo 8** dá um exemplo.

Scripts

As ações também podem ser ativadas antes de se começar a procurar em um arquivo de log, ou após terminar todas as buscas.

O parâmetro `$prescript`, que aponta para um script externo ou subrotina em Perl, ajuda a chamar uma ação. Os *prescripts supersmart* cancelam a execução do check_logfiles caso o código de saída seja maior que zero. Isso possibilita a verificação do estado de execução de um processo.

Se o processo não estiver rodando, por que se preocupar em verificar seus logs respectivos? Os prescripts também podem forçar aplicativos a escreverem (fazer um *flush*) em seus logs, certificando-se de que os dados estejam atualizados.

Os *postscripts supersmart* podem substituir completamente os resultados do check_logfiles, não importa quantas mensagens de erro eles contivessem originalmente.

Ou, se o formato de saída padrão do check_logfiles não for agradável, pode-se executar um postscript supersmart para modificá-lo, a fim de obter uma melhor ênfase. ■

Mais informações

[1] Check_logfiles: http://www.consol.com/opensource/nagios/check-logfiles/

[2] Julian Hein, "Nagios: O verdadeiro grande irmão": http://www.linuxmagazine.com.br/article/1011

Como e por que usar ACLs no sistema de arquivos

Auxílio à lista

O antiqüíssimo sistema de permissões do Linux costuma ser insuficiente para ambientes de produção complexos. As ACLs oferecem uma alternativa muito flexível.

por Tim Schürmann

Alice gosta da conveniência de uma agenda eletrônica em seu computador. Porém, para manter sua privacidade, ela estabeleceu permissões bem estritas para o arquivo da agenda: ela pode criar novos compromissos, mas os outros usuários do seu grupo somente têm permissão para ler o arquivo, enquanto os demais usuários não podem nem ler sua agenda.

Essa configuração pode parecer interessante a princípio, mas certo dia, Bob, de outro departamento, começa a colaborar com Alice. Para permitir que isso ocorra, ela precisa fornecer a ele acesso aos dados de sua agenda.

Nesse cenário, fica claro que o sistema legado de permissões do Linux já ultrapassou seu limite de vida útil. Para permitir que Bob leia o arquivo com a agenda de Alice, ela precisará pedir ao administrador para mover seu novo colega para seu grupo, mas isso permitiria que Bob abrisse todos os outros documentos produzidos pela equipe de Alice.

Outra abordagem seria criar temporariamente um grupo de usuários completamente novo, contendo as contas de Alice e Bob como membros. Nesse cenário simples, um grupo temporário talvez até seja uma solução aceitável, mas, em ambientes corporativos reais, o gerenciamento de grupos é significativamente mais complicado, além de o hábito de criar grupos temporários poder levar a um excesso de grupos sem forma alguma de acompanhar seu ciclo de vida.

As listas de controle de acesso, ou ACLs, prometem uma solução. Elas acrescentam um controle de acessos muito flexível ao sistema legado de permissões do Unix, permitindo que os usuários acrescentem permissões para qualquer grupo ou usuário. Alice nem precisa avisar ao administrador; basta ela inserir Bob na lista de usuários autorizados, podendo até especificar permissões padrão para todos os novos arquivos.

As ACLs já existem há tempos – sendo, inclusive, um padrão *POSIX* (**quadro 1**) – e gradativamente estão se tornando parte da vida cotidiana em vários ambientes de produção; entretanto, a estrutura de segurança das ACLs ainda não é familiar para muitos usuários Linux. Este artigo mostrará como usar ACLs no Linux.

Discos que giram

Para usar ACLs, é necessário que o sistema de arquivos usado ofereça suporte a atributos estendidos (*extended attributes*, ou *xattrs*). Da safra atual de sistemas, *Ext2*, *Ext3*, *Ext4*, *ReiserFS*, *JFS* e *XFS* suprem essa necessidade, sendo que esses recursos estão presentes por padrão no JFS e no XFS; nos outros, é preciso passar a opção `acl` ao montar a partição:

```
mount -o acl,defaults /mnt/bla
```

A maioria das distribuições atuais utilizam por padrão os parâmetros `acl,user_xattr` no arquivo `/etc/fstab`.

No caso de discos internos, não é necessário fazer alterações, e as ACLs também funcionarão em volumes NFS, contanto que o servidor tenha um sistema de arquivos e um kernel que suportem ACLs. Já no caso de sistemas Windows, o **quadro 2** informa o panorama atual do suporte.

Sobre o kernel

Além do sistema de arquivos, o kernel também precisa suportar ACLs – afinal, é ele que permite ou proíbe o acesso aos arquivos. Todos os kernels atuais da série 2.6 possuem suporte a ACLs, além de haver *patches* para a série 2.4. As principais distribuições costumam ativar os atributos estendidos para todos os sistemas de arquivos citados aqui, o que permite que os usuários criem suas próprias permissões "do zero". Para conferir a existência desse suporte, basta o comando:

```
zgrep 'XATTR|POSIX_ACL' /proc/
➥config.gz
```

Se as linhas retornadas não mostrarem o seu sistema de arquivos seguido de `=y`, então o suporte não está ativo para ele.

Quadro 1: POSIX e ACLs

O termo *POSIX ACLs* é muito freqüente em documentação e na Internet. Embora vários rascunhos da especificação tenham surgido no final do último século (POSIX 1003.1e, comumente chamado de *POSIX.1e*, e 1003.2c), os rascunhos jamais foram aprovados por vários motivos. A maioria das implementações de ACL ainda são orientadas por esses rascunhos. Para reforçar essa orientação, muitos autores usam o termo "POSIX ACLs"[1].

Bando de dois

Além do sistema de arquivos e do suporte por parte do kernel, também é necessário um pacote com aplicativos que exibam as ACLs de cada arquivo e as modifiquem conforme necessário. A maioria das distribuições inclui o pacote *acl* justamente para esse fim. Dois programas incluídos nele são particularmente úteis:

- `getfacl` mostra a ACL de um arquivo;
- `setfacl` define ou altera as permissões do arquivo.

Ambas as ferramentas dependem das bibliotecas *Libattr* e *Libacl*, que várias distribuições também já instalam por padrão.

Minimalismo

Uma ACL pode ser imaginada como um pedaço de papel no qual é anotada uma lista de todas as outras permissões de acesso e direitos associados a elas. O Linux grampeia esses resultados no arquivo e efetiva as permissões da lista.

Na prática, Alice primeiro verifica quais permissões já estão atribuídas a sua agenda e, para isso, ela usa o comando `getfacl`, que exibe a ACL do arquivo. Como ela ainda não adicionou Bob, a lista dele deve estar vazia:

```
# getfacl agenda.cal
# file: agenda.cal
# group: equipe
user::rw-
group::r--
other::r--
```

Essa lista apenas replica as permissões definidas pelo sistema legado, chamadas de */ACL mínima*. Assim que for acrescentada alguma entrada (ACE, ou *Access Control Entry*) de de verdade à lista, teremos o que se chama de *ACL estendida*.

Fazendo ACEs

Para permitir o acesso de Bob à agenda, Alice precisa criar uma nova entrada para a ACL do arquivo. A ferramenta `setfacl` faz isso com facilidade:

```
setfacl -m user:bob:rw- agenda.cal
```

Os parâmetros podem parecer crípticos à primeira vista: o comando acima cria uma nova entrada (`-m`) na ACL para o arquivo `agenda.cal` (outras opções no **quadro 3**). O acesso será permitido a um único usuário chamado *bob*. Bob pode ler e escrever o arquivo, mas não consegue executá-lo (`rw-`). O comando `getfacl` gera a seguinte lista de saída:

```
$ getfacl agenda.cal
# file: agenda.cal
# owner: alice
# group: equipe
user::rw-
user:bob:rw-
group::r--
mask::rw-
other::r--
```

Toda entrada da ACL segue o mesmo padrão, começando com o tipo, que especifica a quem as permissões a seguir se aplicam. Ele pode ser um único usuário (`user`) ou um grupo inteiro (`group`). Depois dos dois pontos (`:`) vem um rótulo, que determina a quem a entrada pertence. Quando esse rótulo não for necessário, pode-se simplesmente omiti-lo. A linha termina com a notação de permissões já familiar.

Quadro 2: Windows e ACLs

As versões do Windows da família NT (XP, 2000 e mais recentes) oferecem suporte a ACLs, mas somente com o sistema de arquivos NTFS. Por outro lado, o Linux não tem suporte a ACLs em NTFS.

Pelo menos o *Samba* suporta ACLs, supondo que o sistema subjacente tenha suporte a permissões estendidas. Os arquivos gravados no servidor Samba mantêm suas permissões como se estivessem num volume NTFS comum. Porém, ainda há um problema: como as ACLs do Windows e do Linux são diferentes, o Samba oferece a usuários Windows somente uma parte das funcionalidades a que estes estão habituados.

Com que direito?

Ao longo do projeto, Alice precisa conceder aos outros membros do grupo de Bob acesso temporário a sua agenda. Para isso, ela simplesmente acrescenta uma entrada para *equipebob*:

```
$ setfacl -m group:equipebob:r--
 ➥ agenda.cal
$ getfacl agenda.cal
# ...
user::rw-
user:bob:rw-
group::r--
group:equipebob:r--
mask::rw--
other::r--
```

Quando for pedido acesso de leitora a um arquivo, o Linux simplesmente vai verificar as permissões, uma por uma.

Primeiro, o kernel verifica se o usuário possui uma entrada e, em caso positivo, aplica as permissões definidas pela entrada. No exemplo deste artigo, Bob recebe acesso de leitura e escrita para a agenda. Porém, como não há uma entrada pessoal para outros usuários, o Linux tenta as permissões de grupo. Como o usuário *Carl* pertence ao grupo *equipebob*, ele recebe direito de leitura. Se o kernel não encontrar uma entrada de grupo na qual Carl se ecaixe, ele usará as permissões legadas do Unix.

Legado

Infelizmente, `ls -l` não exibe as permissões estendidas. Em vez disso, um sinal `+` indica sua presença:

```
$ ls -l agenda.cal
-rw-r--r--+ 1 alice equipe 5410 7
➥ Jan 11:21 agenda.cal
```

Como as ACLs mapeiam permissões Unix padrão, o `setfacl` também substitui o bom e velho comando `chmod`. Basta alterar as entradas. Por exemplo, o comando:

```
setfacl -m other::rw- agenda.cal
```

dá acesso de leitura e escrita no arquivo para todos os outros usuários:

```
-rw-r--rw-+ 1 alice equipe 5410 7
➥ Jan 11:21 agenda.cal
```

Toda ACL estendida contém uma entrada de `mask` bastante clara. A máscara descreve as permissões máximas que o usuário pode receber.

Se a máscara definir permissões mais restritivas do que as dadas a um usuário numa ACE, a máscara sempre terá prioridade. Por exemplo, Alice poderia ter dado aos outros membros do grupo de Bob acesso a seu calendário.

Se, agora, ela quiser revogar temporariamente essas permissões, ela precisará modificá-las para cada membro do grupo; ou, então, pode modificar apenas a máscara:

```
setfacl -m mask::r-- agenda.cal
```

Independente das permissões que os usuários tivessem antes, a partir de agora só poderão ler a agenda. Obviamente isso também se aplica a Bob. Embora ele ainda tenha permissão de escrita na agenda, a máscara tem prioridade, deixando-o com somente três permissões.

Sob o capô, o Linux realiza uma operação lógica *E* para calcular os direitos efetivos. Para conseguir ler um arquivo, o usuário ou grupo precisa, portanto, ter permissão de leitura, que precisa existir na máscara.

Para evitar que os administradores se percam nesse excesso de informações, o `getfacl` exibe as permissões efetivas para cada usuário:

```
$ getfacl agenda.cal
# ...
user::rw-
user:bob:rw-      #effective:r--
group::r--
mask::r--
other::r--
```

Embora Bob possua permissão de escrita, ele só consegue obter efetivamente permissão para ler a agenda.

As máscaras introduzem outra armadilha: o `setfacl` altera de forma autônoma a máscara sempre que forem modificadas as permissões

Quadro 3: Usos do setfacl

O comando `setfacl` possui alguns parâmetros muito úteis. O primeiro é `-m`, que modifica ou cria uma nova entrada:

```
setfacl -m user:bob:r-- agenda.cal
```

O parâmetro `-x` apaga-a:

```
setfacl -x user:bob agenda.cal
```

Mas isso apaga apenas a entrada especificada, sem afetar o grupo a que Bob pertence. `--set` apaga todas as entradas anteriores, definido somente as novas:

```
setfacl --set user:bob:r-- agenda.cal
```

Por último, a opção `-b` esvazia a lista completa. Além dela, usar o parâmetro `-R` ativa o comportamento recursivo. Ao mesmo tempo, é possível até definir múltiplas permissões separadas por vírgula:

```
setfacl -m user:bob:r--, group:equipe:rw- agenda.cal
```

Em vez de especificar os nomes de dono e grupo, pode-se usar seus UIDs e GIDs; o `setfacl` também aceita permissões em formato numérico.

Também podem ser usadas algumas abreviações com o comando. Em vez de `user`, pode-se usar `u`, da mesma forma que `g` para `group`, `m` para `mask`, `o` para `other` e `d` para `default`. Também é possível substituir múltiplos traços seguidos por um único, contanto que o comando não fique ambíguo:

```
setfacl -m u:bob:r- agenda.cal
```

de usuário ou grupo. Em caso de mau uso, como Alice fez em nosso exemplo, não há alternativa além de verificar sua validade.

Padrão

Enquanto trabalha num projeto, Alice cria vários arquivos que Bob também precisará ler. Para cada novo documento, Alice poderia modificar as permissões manualmente. Uma opção mais fácil seria criar diretórios com uma ACL padrão. Os subdiretórios e arquivos sob ele herdam automaticamente suas permissões padrão. Os diretórios têm tanto uma ACL própria quanto uma nova ACL padrão para o diretório-pai. É possível criar uma nova ACL padrão com o `setfacl` (**exemplo 1**).

O `getfacl` sempre lista permissões padrão ao final. O formato reflete as entradas legadas, mas começando com `default`. Para exibir somente as entradas padrão, o `getfacl -d` deve ser usado, enquanto `getfacl -a` omite-as todas.

Como o acesso ao arquivo é tratado pelo próprio kernel, programas legados não têm dificuldade para trabalhar com permissões estendidas, diferentemente de aplicativos que manipulem permissões de arquivos, como *Konqueror* e *Nautilus* – exceto as versões mais recentes.

Comandos Unix padrão, como `cp` e `mv`, já têm as alterações necessárias para lidar com ACLs. Para copiar arquivos mantendo absolutamente todas as suas características, portanto, é importante que o sistema de destino também suporte ACLs. Senão, serão mantidas as permissões legadas.

Programas sem as modificações para suporte a ACLs simplesmente modificam as permissões padrão. Um exemplo disso é o `tar`, que não enxerga ACLs, e que, por isso, deve ser substituído por seu concorrente `star`, também incluído em diversas distribuições.

Para evitar formatos exóticos no intercâmbio de dados, um macete interessante é fazer o `setfacl` obter seus parâmetros a partir de um arquivo de texto. O formato precisa obedecer precisamente a saída do `getfacl`, portanto, deve-se usar o próprio `getfacl` para armazenar todas as ACLs num arquivo de texto e depois executar o `setfacl` para restaurá-las. O **quadro 4** informa ainda como usar isso para copiar ACLs de um arquivo para outro.

O comando:

```
getfacl -R --skip-base \
diretório/ > /becape.acl
```

faz o `getfacl` ir para o diretório `diretório/` e gravar as ACLs de todos os objetos dele no arquivo `becape.acl`. O parâmetro `-R` torna esse processo recursivo. Depois, para restaurar as ACLs, basta o comando:

```
setfacl --restore=backup.acl
```

Conclusões

As listas de controle de acesso permitem um gerenciamento de permissões muito flexível. Ao mesmo tempo, aliviam um pouco do trabalho do administrador, pois transferem a responsabilidade pelas permissões de acesso para o dono do arquivo. Graças a ACLs e máscaras padrão, o administrador mantém o controle dos objetos.

O intercâmbio de dados permanece um problema importante. Como quase todo sistema operacional usa uma variante diferente de ACL, as várias versões geralmente são incompatíveis ou parcialmente compatíveis, o que significa que as permissões costumam ser perdidas em processos de conversão. Os administradores de ambientes heterogêneos, portanto, devem ser cautelosos com esse recurso. ■

Exemplo 1: Criação de uma ACL padrão

```
01 $ setfacl -m user:bob:rw-
➥dirprojeto
02 $ setfacl -d --set user:
➥bob:rw dirprojeto
03 $ getfacl dirprojeto
04 # file: projectfolder
05 # owner: alice
06 # group: ateam
07 user::rwx
08 user:bob:rw-
09 group::r-x
10 mask::rwx
11 other::r-x
12 default:user::rwx
13 default:user:bob:rw-
14 default:mask::rwx
15 default:other::r-x
```

Quadro 4: Transferência de permissões

Para criar ACLs complexas, é preciso rodar o `setfacl` múltiplas vezes, o que é trabalhoso e toma tempo. Felizmente, os administradores podem armazenar as entradas num arquivo texto no mesmo formato de saída do `getfacl` e depois executar o `setfacl` com o parâmetro `--set-file="arquivo.txt"` para restaurar as permissões. O uso de `-` em vez de um nome de arquivo faz com que a ferramenta leia os parâmetros da entrada padrão. Isso significa que é possível mover uma ACL de um arquivo para outro:

```
getfacl arq1 | setfacl --set-
➥file=- arq2
```

Mais informações

[1] Rascunhos de ACLs POSIX: http://wt.xpilot.org/publications/posix.1e

De volta ao shell – mas com janelas

Papo de botequim 2.0 Parte I

Na volta da série sobre programação shell, modernize seus scripts com o uso do Zenity.

por Julio Cezar Neves

Há alguns dias, eu estava andando no calçadão e eis que encontro o Chico, meu garçom preferido. Conversamos um pouco, quando ele me contou que estava trabalhando, vejam só, no "Pingüim Carioca". Como já sou mesmo chegado a um boteco e muito mais a um pingüim, no primeiro sábado que passaria em casa convidei um amigo para tomar um chope.

Lá chegando, logo após a primeira rodada de canecas, o amigo me fala na maior cara de pau:

– Pô, mermão (com sotaque carioca), você tem de se modernizar. Por que insistir em desenvolver programas com interfaces arcaicas orientadas a caractere? Por que você não parte para algo mais atual?

Fiquei indignado com aquilo e respondi:

Tabela 1: Opções de comando para diálogos

Opção	Efeito
`--calendar`	Mostra um calendário
`--entry`	Mostra uma caixa para inserir textos
`--error`	Mostra uma janela de erros
`--file-selection`	Mostra uma janela para selecionar arquivos
`--info`	Mostra uma janela com informações
`--list`	Mostra uma caixa com lista
`--notification`	Mostra uma janela de notificação
`--progress`	Mostra uma barra de progresso
`--question`	Mostra uma janela para fazer uma pergunta
`--text-info`	Mostra uma janela de texto
`--warning`	Mostra uma janela de advertência
`--scale`	Mostra um escala para opções

– Cara, é você que tem de se modernizar e largar essa porcaria que você chama de sistema operacional e que te deixa na mão a toda hora, sem falar nos custos, vírus e nas telas azuis que você ganha pelo menos uma vez ao dia.

A partir de hoje e nas próximas vezes que viermos aqui, vou lhe mostrar as interfaces gráfica que foram desenvolvidas para serem usadas diretamente do Shell, e você verá que são muito mais concisas e fáceis de usar do que as que você conhece.

É necessário, no entanto, ter um bom conhecimento de Shell, o que você consegue nos *Papos de Botequim* que foram publicados na Linux Magazine desde o fascículo número 1 até o número 11[1].

– Chico, traz mais dois chopes! Você não esqueceu que o meu é sem colarinho, né?

Agora, e nos nossos próximos encontros, vou lhe dar um curso sobre os programas que fazem interface gráfica para o Shell, começando pelo *Zenity*, que é completo e extenso o suficiente para que esse nosso papo se estique por mais uns três encontros.

Tabela 2: Parâmetros genéricos

Opção	Efeito
`--title`=TITULO	Define um título para as caixas
`--window-icon`=ICONE	Define um ícone para as caixas
`--width`=LARGURA	Define a largura das caixas
`--height`=ALTURA	Define a altura das caixas

Tabela 3: Parâmetros de calendário

Opção	Efeito
`--day`=INT	Define o dia padrão
`--month`=INT	Define o mês inicial
`--year`=INT	Define o ano inicial
`--date-format`=PATTERN	Define formato da data (mesmo formato do `date`)

Como tudo começou

Um dia, o Tujal me pediu para dar um acabamento legal em uns scripts que ele desenvolveu para o *CVS* (Concurrent Version System – ou seja, um gerenciador de código-fonte) e fui procurar na Internet como fazê-lo usando o Zenity, do qual já tinha ouvido falar muito bem.

Considero esse software muito importante, pois, dominando Shell e Zenity, você poderá desenvolver seus scripts em Shell – que, como você já viu ou verá nos 11 primeiros números da Linux Magazine, é uma linguagem simples e concisa – e dar-lhes um excelente acabamento gráfico com Zenity.

O Zenity é a cara do Shell: facílimo de usar e super conciso. Essas duas ferramentas se complementam de forma a facilitar sua vida em programas curtíssimos e poderosos.

Afinal de contas, quem é esse cara?

Zenity é um programa que se utiliza de ferramentas da GTK+ para produzir interfaces gráficas muito bem acabadas que atuarão entre scripts em Shell (e outras linguagem orientadas a caractere) e os usuários, provendo entre ambos uma correlação amigável e bonita.

O Zenity é um executável que recebe todos os parâmetros via linha de comando e retorna – no código de retorno (`$?`) ou na saída primária (*stdout*) – a escolha do usuário. Isso permite apresentar, pedir e trocar informações com o operador.

Por exemplo, o comando `zenity --question` apresentará uma janela com uma pergunta e dois botões (um de *OK* e outro de *CANCEL*). O código de retorno (`$?`) será `0` ou `1`, dependendo se você clicou no *OK* ou *CANCEL*. O comando `zenity --entry` pedirá uma entrada de dados e mostrará os mesmos dois botões. Você obterá, da mesma forma, um código de retorno (`$?`) igual a `0` ou `1` e a saída primária receberá o que foi digitado. Experimente!

Opções de diálogo

A **tabela 1** mostra uma síntese das opções da linha de comando para exibição do diálogo. A partir de agora, esmiuçaremos todos os parâmetros aceitos em cada uma destas opções descritas. No entanto, alguns argumentos podem ser utilizados por qualquer uma das opções e é por eles que começaremos nossa aprendizagem.

Parâmetros genéricos

A **tabela 2** contém os parâmetros genéricos do comando `zenity`.

O parâmetro `--title` especifica o título que fica na borda superior da caixa. `--window-icon` permite alterar os ícones padrão das caixas de diálogo, enquanto `--width` e `--height` especificam, respectivamente, a largura e a altura das caixas de diálogo, em pixels.

Calendário

O diálogo `calendar` serve para você escolher uma data. As opções `--day`, `--month` e `--year` permitem a especificação, respectivamente, do dia

Figura 1 Diálogo de calendário.

Exemplo 1: Comando para exibição de calendário

```
01 until data=$(zenity --calendar    \
02     --title="Datas dos vôos"      \
03     --text="Escolha uma data para o
➥ vôo de ida" \
04     --day=$(date +%-d)            \
05     --month=$(date +%-m)          \
06     --year=$(date +%Y))
07 do
08     :
09 done
10 echo $data
11 12-07-2008
```

Tabela 4: Opções de diálogos de advertência

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

padrão, o mês e o ano do calendário que será exibido (veja a **tabela 3**).

Veja o fragmento de script do **exemplo 1**. Nele, me aproveitei da facilidade do botão *CANCEL* devolver um código de retorno (`$?`) diferente de zero para permanecer no *loop* do `until`, que nada faz (comando `:`), e só terminar quando uma data for escolhida. O resultado da escolha (**figura 1**) foi para a variável `$data`.

Neste exemplo, não foi necessário informar os parâmetros –day, –month e –year, pois seus valores padrão (default) são os da data de hoje, e foram justamente estes os valores informados. Mas note o uso das máscaras `+%-d` e `+%-m`, necessárias para evitar a inserção de um zero antes dos números de um algarismo. O zenity precisa receber argumentos inteiros decimais nas opções `--day`, `--month` e `--year`, e o comando `date` `+%m` retornaria, por exemplo `09`, o que pode ser interpretado como um inteiro octal.

Como você viu, a saída padrão é no formato *dd-mm-aaaa*. Para alterar esse formato, podemos usar os mesmos caracteres de formatação do comando `date`. Experimente o seguinte:

```
zenity --calendar \
    --date-format="%Y/%m/%d"
```

O diálogo da **figura 1**, após você selecionar uma data, vai retornar algo como `2008/07/28`.

Advertência

A opção de diálogo `--warning` serve para você fazer uma advertência ao operador. O **exemplo 2** contém um fragmento de programa que precisa ser executado pelo usuário.

As opções específicas desse tipo de diálogo são `--text` e `--no-wrap` (**tabela 4**).

Na **linha 1**, comparamos a variável do interna `$UID`, do operador, e comparamos com zero (qualquer usuário root tem `UID=0`). Caso seja diferente, um diálogo de atenção será disparado (**figura 2**).

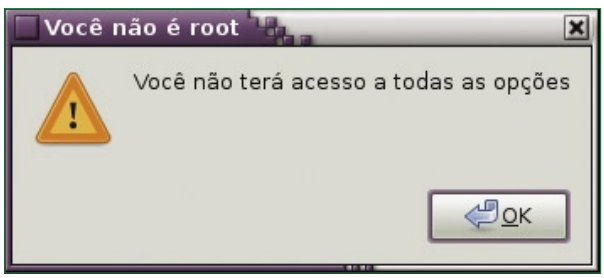


Figura 2 Diálogo de advertência.

Exemplo 2: Diálogo de advertência

```
01 [ $UID -eq 0 ] ||
02     zenity --warning              \
03         --title="Você não é root" \
04         --text="Você não terá acesso
➥ a todas as opções"
```

Pergunta

O diálogo de pergunta (`--question`) possui as mesmas opções que o de advertência (**tabela 4**) e lhe fará uma pergunta e oferecerá como resposta dois botões: *OK* e *CANCEL*. Caso seja pressionado o *OK*, o código de retorno (`$?`) será igual a zero; caso contrário, será diferente de zero.

Aproveitando isso, após o diálogo de advertência mostrado na **figura 2**, poderíamos perguntar ao operador se ele deseja mudar para root (**figura 3**), como faz o **exemplo 3**.

Ao fim da execução, a variável `$Resp` teria obrigatoriamente o valor `s` ou `n`.

As páginas de manual, por incrível que pareça, têm um exemplo bastante engraçado do uso desta opção. Confira.

Temos ainda o parâmetro `--no-wrap`, que pode ser usado caso o texto definido por `--text` seja grande e você não queira dividi-lo em mais de uma linha.

Exemplo 3: Diálogo de pergunta

```
01 Resp=n
02 zenity --question      \
03     --title "Responda" \
04     --text "Deseja mudar para
➥ root?" && Resp=s
```

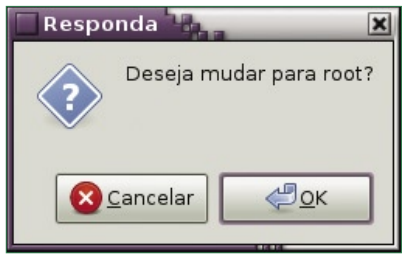


Figura 3 Diálogo de pergunta.

Com entrada

Os diálogos do tipo `--entry` (entrada) abrem uma janela de diálogo requisitando uma entrada de dados ao operador, sendo suas opções descritas na **tabela 5**. Vamos continuar nos exemplos em que o cara deveria ser root para aproveitar todas as facilidades oferecidas por um deter-

Tabela 5: Opções de diálogos de entrada

Opção	Efeito
`--text=TEXTO`	Define o texto da caixa
`--entry-text=TEXTO`	Valor padrão
`--hide-text`	Esconde o valor digitado (senhas)

Exemplo 4: Diálogo de entrada

```
01 if  zenity --question  \
02     --title "Responda" \
03     --text "Deseja mudar para root?"
04 then
05     Senha=$(zenity --entry                \
06         --title "Captura de Senha"        \
07         --text "Informe a senha de root:" \
08         --hide-text)
09 fi
10 echo $Senha
11 123456
```

Exemplo 5: Valor padrão na entrada

```
01 Usuario=$(zenity --entry                                    \
02     --title "Captura nome de usuário"                       \
03     --text "Informe o nome do usuário na máquina remota:" \
04     --entry-text $LOGNAME)
```

Tabela 6: Formato do arquivo de aniversariantes

Campo	Data de Nascimento	Nome	Endereço de e-mail	Telefone
Formato	DD/MM/AAAA	Xxxx Xxxx	Usuário@dominio	(DD) NNNN-NNNN

minado aplicativo, mas, primeiramente, vamos pegar nosso exemplo anterior e inseri-lo nesse contexto (**exemplo 4**).

O comando `if` testou o diálogo de pergunta e, caso o botão clicado seja `OK`, o programa lançará a caixa de entrada de dados da **figura 4** para capturar a senha. Repare que a opção `--hide-text` não deixa o texto que está sendo digitado aparecer na tela.

Também poderíamos oferecer um valor padrão com a opção `--entry-text`, como mostram o **exemplo 5** e a **figura 5**.

– Pois é, amigo, está gostando?

– É, por enquanto o negócio parece ser bom e conciso como você falou...

Figura 4 Diálogo de entrada de dados.

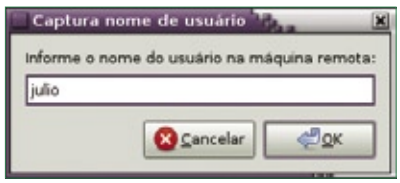


Figura 5 Diálogo com valor padrão para o texto.

– Então, de brincadeira, vamos desenvolver um "sistemeco" para nos lembrar dos aniversários dos amigos. Dessa vez, vou te deixar a incumbência de fazer um script para montar o arquivo de aniversariantes, que tem o layout descrito na **tabela 6**.

– E o separador entre os campos é o dois-pontos (:). O script será feito em shell, porém, todas as interações com o usuário, isso é, leitura dos dados, avisos de erros e advertências, serão gráficas.

– Chico, fecha a conta que eu vou nessa, mas não demoro a voltar... ■

Mais informações

[1] Baixe todas as edições da série Papo de Botequim: http://www.linuxmagazine.com.br/noticia/1729

Sobre o autor

Julio C. Neves é um dos mais antigos colaboradores do SL no Brasil e trabalha atualmente no SERPRO ao lado de Marcos Mazoni, contribuindo para que a principal empresa de TI do Brasil mantenha-se cada vez mais na liderança latino americana no uso e difusão de SL.

Crie websites eficientes com AJAX

AJAX: ágil e a jato

No princípio, páginas web se comportavam como livros. Graças ao AJAX, os sites modernos são mais semelhantes a aplicativos de verdade.

por Carsten Zerbst

Os padrões para uma boa apresentação na Internet são bem diferentes hoje do que eram quando Tim Berners-Lee criou as primeiras páginas da Web. Os sites se assemelham, cada vez mais, a aplicativos interativos de desktop, deixando para trás o antiquado visual de material impresso. O AJAX é uma tecnologia baseada no *JavaScript* que adiciona conveniência por meio de menus *drop-down*, tabelas ordenáveis e páginas de entrada de dados interativas. O principal avanço é a ausência de atrasos geralmente experimentados enquanto as páginas eram recarregadas.

Longo caminho

Antes de renderizar um website, o navegador e o servidor web cumprem diversas etapas (**figura 1**):

- O navegador envia uma requisição de página ao servidor.
- O servidor processa a requisição e serve o texto HTML e as imagens. Isso pode levar alguns segundos se a carga for pesada. A velocidade de transmissão da rede decide a rapidez de entrega do conteúdo. O tempo necessário ainda é perceptível, mesmo em intranets rápidas.
- Por último, o navegador lê a resposta e exibe a página. A mesma seqüência se repete para cada imagem antes de o navegador conseguir renderizar a versão final da página.

Esses três passos costumam levar vários segundos. No caso de páginas HTML sem tecnologia AJAX, os passos são repetidos até para as menores alterações.

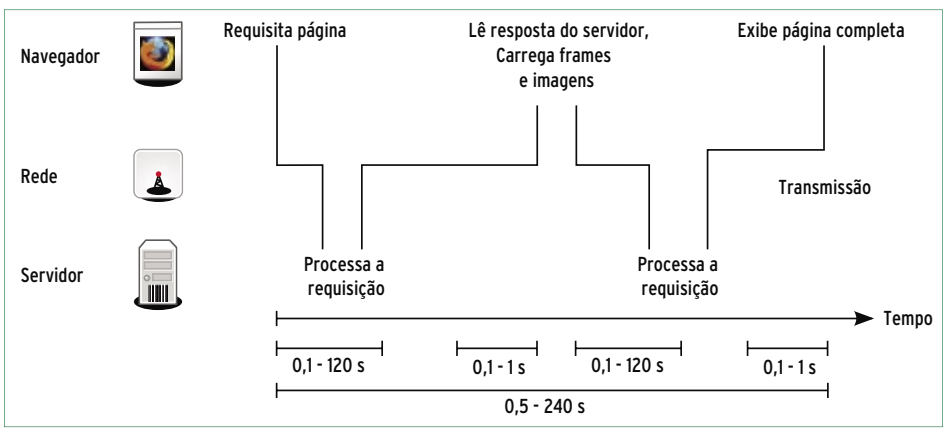


Figura 1 Pingue-pongue: a troca entre o navegador e o servidor web que ocorre para cada alteração da página sem AJAX dura vários segundos.

Diferentemente das *Rich Internet Applications*, isso afeta consideravelmente a experiência do usuário: menus que se abrem sem atraso, ordenação de tabelas com cliques do mouse ou arrastar-e-soltar não são fáceis de implementar em razão das lentas recargas de página. As páginas HTML que oferecem esses tipos de recursos precisam ser autônomas, como programas locais; ou seja, não se deve depender de uma conexão com o servidor.

Sem requisições lentas

Para melhorar a experiência do usuário, mais e mais aplicativos web estão começando a processar a entrada dos usuários diretamente no navegador, reduzindo assim as requisições lentas ao servidor.

Somente duas das várias técnicas de processamento de dados no lado do navegador alcançaram grande sucesso: JavaScript e *Flash*. Ambas estão disponíveis em mais de 90% de todos os computadores. Isso significa que desenvolvedores web não precisam ter medo de usá-las.

Outras soluções, à exceção do relativamente difundido plugin *Java*, foram incapazes de atingir um sucesso multi-plataforma tão significativo. Mas Flash e JavaScript adotam técnicas completamente diferentes entre si.

Flash

O plugin proprietário do Flash executa aplicativos Flash binários no navegador. O plugin é embutido na página da mesma forma que uma imagem *bitmap*, exceto pelo fato de que oferece ao usuário uma interface interativa. O plugin do Flash possui excelentes capacidades gráficas e praticamente não oferece restrição à criatividade do desenvolvedor. Entretanto, por falta de uma alternativa de código aberto equivalente, não há alternativas às ferramentas da Adobe para sua criação.

JavaScript

Em contraste, o JavaScript não é restrito a áreas isoladas da página. O interpretador integrado ao navegador executa o programa e converte a página inteira em uma interface modificável dinamicamente. Para isso, os scripts criam ou modificam o código HTML na página, modificam os estilos CSS e até desenham imagens.

Os scripts propriamente ditos são textos não compilados. Diferentemente do desenvolvimento em Flash, programar em JavaScript não requer ferramentas especiais. Um simples editor de texto é suficiente, ao menos para começar.

Boas ferramentas

Como sempre, boas ferramentas facilitam a programação. Um editor com suporte a HTML , CSS e JavaScript é bastante útil. Ele também deve ser capaz de lidar com código-fonte que mistura todos os três[1][2]. O plugin *Web Developer*[3], feito por Chris Pederick para o *Firefox*, é a escolha óbvia para analisar e depurar programas. Ele revela falhas em HTML, CSS e JavaScript, investiga cookies e exibe o código HTML modificado dinamicamente, não apenas a versão original entregue pelo servidor.

Tabelas

Um número crescente de aplicativos, como sistemas de processamento de pedidos ou de gestão empresarial (ERPs), usam interfaces web: listas de componentes e outras visões em listas são os que mais se vê.

Direitos	Tipo	Dono	Grupo	Tamanho	Alteração	Nome
lrwxrwxrwx	1	root	root	18	11.02.08	printcap
-rw-r--r--	1	root	root	149	26.11.06	hosts.deny
-rw-r--r--	1	root	root	188	26.11.06	hosts.equiv
-rw-r--r--	1	root	root	191	26.11.06	hosts.lpd
-rw-r--r--	1	root	root	530	27.01.07	group.YaST2save
-rw-r--r--	1	root	root	536	27.01.07	group.old
-rw-r--r--	1	root	root	551	27.01.07	group
-rw-r--r--	1	root	root	677	19.12.06	hosts.YaST2save
-rw-r--r--	1	root	root	715	27.12.06	hosts
-rw-r--r--	1	root	root	1357	02.10.08	passwd
-rw-r--r--	1	root	root	2639	26.11.06	hosts.allow

Figura 2 Nova ordem: Clicar no cabeçalho da tabela faz o *JavaScript* reordenar a tabela, sem obrigar o servidor a participar.

Um aspecto importante é a capacidade do usuário de ordenar essas listas. Se a interface web for baseada em HTML estático, o servidor precisará gerá-la novamente e servir a versão modificada. Obviamente, a ordenação baseada em código JavaScript no lado cliente torna o processo bem mais veloz.

A **figura 2** dá um exemplo de listagem de diretório implementada numa tabela HTML. Clicar no cabeçalho da tabela ordena-a pela coluna selecionada. A **figura 3** mostra como fazer isso com JavaScript. Uma tabela HTML compreende elementos `<tr>` e `<td>` aninhados, que podem ser referenciados via DOM[4].

O script inicia apagando dinamicamente todos os elementos `<tr>` (ou seja, as linhas da tabela) da página. As linhas só existem como um vetor no JavaScript. A função em JavaScript então ordena o vetor na coluna pedida e escreve a nova ordem entre as tags vazias `<table>` e `</table>`. Por último, o script desenha uma seta, um caractere de seta Unicode [5], antes do cabeçalho da coluna que funcionou como base para a ordenação.

Sistema modular

Esse método pode ser implementado com poucas centenas de linhas de código; porém, é mais fácil usar soluções já existentes. Há dezenas de bibliotecas JavaScript populares na Internet que incluem funções para tarefas freqüentemente necessárias,

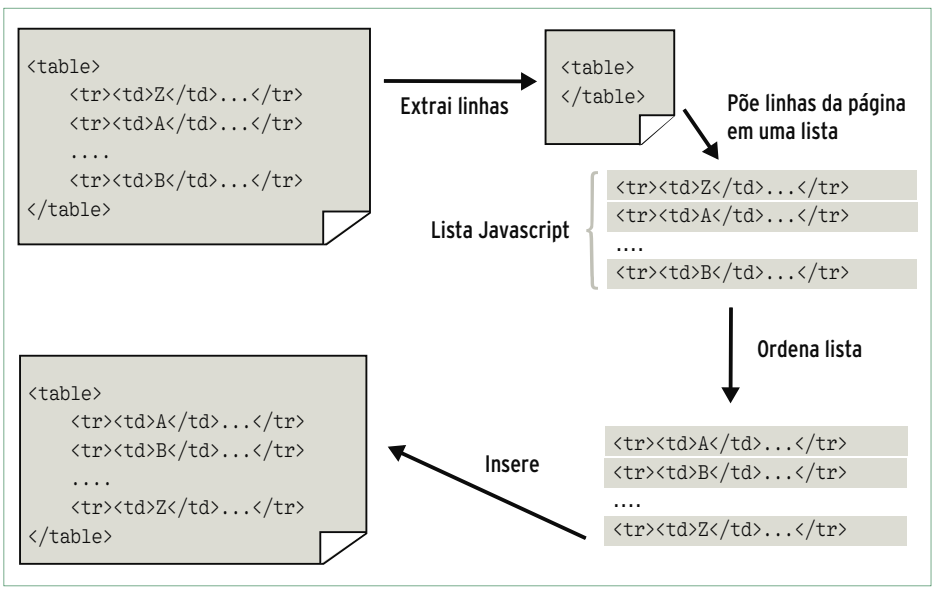


Figura 3 O *JavaScript* ordena tabelas removendo dinamicamente as linhas da tabela, fazendo *cache* dos resultados num vetor e reinserindo os dados na nova ordem.

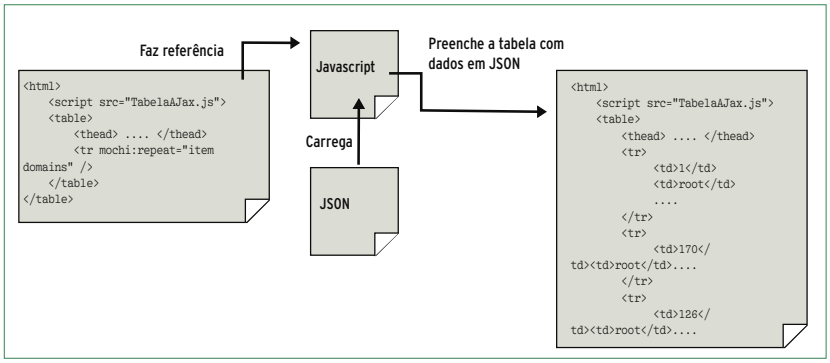


Figura 4 Mais dinâmico: se o *JavaScript* receber do servidor os dados em formato JSON, eles poderão ser ordenados e a tag <table> atualizada sem recarregar a página.

como ordenação de tabelas, criação de cantos arredondados em HTML e desenho de diagramas em árvore.

As vantagens das bibliotecas, em comparação com soluções do tipo "faça você mesmo", são o enorme escopo de funções e a compatibilidade garantida com navegadores populares. Apesar de seguir o padrão ECMA[6], as implementações de JavaScript em navegadores para Linux, Mac OS X e Windows diferem em aspectos muito importantes. A maioria das bibliotecas atuais abstraem essas diferenças para facilitar o trabalho dos desenvolvedores.

O **exemplo 1** mostra o código HTML de uma tabela ordenável baseada na biblioteca aberta *Mochikit*[7]. Além dos dois `includes` na **linha 6** que chamam o Mochikit, o código-fonte HTML é um pouco diferente de uma tabela estática. A tag `table` precisa de um ID único (`tabela Ordenavel`) assim como no exemplo de menu. Os atributos específicos do Mochikit `mochi:format="int"` e `mochi:format="gdate"` nas tags `<th>` ajudam o Mochikit a ordenar colunas numéricas e de datas corretamente.

Dados dinâmicos

O exemplo anterior ordena o conteúdo da tabela no código-fonte HTML; o exemplo a seguir vai um passo além: o aplicativo obtém o conteúdo da tabela sem recarregar a página do servidor. Links, botões e os itens do menu permitem que o usuário preencha a tabela com valores diferentes. Além disso, também é possível exibir resultados de busca sem recarregar a página.

A **figura 4** ilustra a tecnologia exibida aqui. O servidor fornece os dados em *JavaScript Object Notation* (JSON), um formato de texto que utiliza colchetes e vírgulas como separadores.

Exemplo 1: Tabela ordenável

```
01 <!DOCTYPE HTML PUBLIC "- /W3C//DTD HTML 4.01 Transitional//EN">
02 <html>
03   <head>
04     <title>Tabela ordenável</title>
05     <meta http-equiv="Content-Type" content="text/html;charset=UTF-8">
06     <link href="tabelle.css" rel="stylesheet" type="text/css" />
07     <script type="text/javascript" src="lib/MochiKit/MochiKit.js"></script>
08     <script type="text/javascript" src="Tabela ordenável.js"></script>
09   </head>
10   <body>
11     <TABLE id="Tabela ordenável"class="datagrid">
12       <THEAD>
13         <TH>Direitos</TH> ....
<TH mochi:format="int">Tamanho</TH><TH mochi:format="gdate">Alteração</TH>
14       </THEAD>
15       <TBODY>
16         <TR>
17           <TD>-rw-r--r</TD><TD>1</TD><TD>root</TD><TD>root</TD><TD>551
  </TD><TD>27.01.07</TD><TD>group</TD>
18         </TR>
19         <TR>
20           ...
21         </TR>
22       </TBODY>
23     </TABLE>
24   </body>
25 </html>
26
```

JSON

XML é uma alternativa popular ao JSON. A vantagem desse último é o menor *overhead* quando comparado ao desnecessariamente verboso XML. A função JavaScript `eval()` converte o código JSON para objetos JavaScript normais.

O **exemplo 2** exibe o código HTML. Em vez do conteúdo da tabela, o código possui apenas um elemento `tbody` para marcar o lugar. O arquivo JavaScript referido no cabeçalho da tabela, `TabelaAjax.js`, substitui o `tbody` pelo novo conteúdo com valores obtidos do arquivo JSON (**exemplo 3**).

Se as listas forem mais compridas, fica mais fácil carregar apenas as primeiras 25 linhas. Essa técnica reduz o tempo de espera do usuário, assim como a carga do servidor.

Imagens

Os exemplos até aqui se restringiram a elementos HTML simples, como listas e tabelas. Widgets e funções não fornecidos nativamente pelo HTML podem ser programados em JavaScript em combinação com CSS.

Consultar a posição do mouse permite que o desenvolvedor implemente textos auxiliares e também recursos como arrastar-e-soltar. A maioria das bibliotecas JavaScript inclui implementações.

Isso não elimina as capacidades do HTML dinâmico, incluindo imagens que o HTML legado só conseguiria implementar embutindo bitmaps.

A **figura 5** mostra uma estrutura de árvore desenhada com JavaScript. Imagens dinâmicas no lado cliente têm várias vantagens quando comparadas a bitmaps: a resolução não é fixa e portanto pode ser modificada para refletir o tamanho pré-configurado de fonte do navegador.

Dados entrados pelo usuário podem ser visualizados em interação com o servidor, reduzindo a carga sobre este. Normalmente, boa parte da banda de um servidor web é consumida pelo serviço de imagens.

Extensões HTML

Existem basicamente duas técnicas para se usar JavaScript na criação de imagens. A linguagem SVG[8], baseada em XML, é a que oferece o maior número de recursos, em comparação com a *Vector Markup Language* (VML)[9]. Elas podem ser embutidas em HTML como imagens normais, mas também podem acrescentar elementos gráficos típicos – como linhas, áreas ou texto em tempo de execução. Portanto, elas podem responder interativamente à entrada de dados pelo usuário, da mesma forma que o HTML dinâmico modificado por JavaScript.

Programas de desenho, como o *Inkscape*[10] ou o *Karbon*[11], podem ser úteis na criação de rascunhos.

Infelizmente, não há como garantir a compatibilidade multi-navegador: nenhum dos navegadores mais populares suporta todos os três formatos.

O Firefox suporta SVG e *Canvas*[12], o Safari apenas Canvas, e o Internet Explorer apenas VML.

O Google possui duas bibliotecas JavaScript: uma para SVG[13] e outra para Canvas[14]; entretanto, não é garantido o suporte na plataforma Windows.

Ferramentas atestadas

Muitos diagramas e imagens podem ser criados com JavaScript e as ferramentas HTML e CSS padrão, como o diagrama em árvore da **figura 5**.

As caixas são compostas por elementos CSS `div` livremente posicionados. Se for usado `em` em vez de `px` (pixel) como unidade para localização e tamanho, as dimensões e a posição da caixa serão baseadas no tamanho da letra *m*.

Exemplo 2: Tabela gerada dinamicamente

```
01 <!DOCTYPE HTML PUBLIC " //W3C//DTD HTML 4.01 Transitional//EN">
02 <html>
03   <head>
04     <link href="tabelle.css" rel="stylesheet" type="text/css" />
05     <script type="text/javascript" src="lib/ MochiKit/MochiKit.js"></script>
06     <script type="text/javascript" src="TabelaAjax.js "></script>
07   </head>
08   <body>
09     <a href="index.html">Exemplos</a>
10     <hr>
11     <h4>Ajax Table</h4>
12     <p>
13       <a href="top.json" mochi:dataformat="json">Recarregar Tabela 1</a><br>
14       <a href="top2.json" mochi:dataformat="json">Recarregar Tabela 2</a>
15     </p>
16     <table id="sortable_table" class="datagrid">
17       <thead>
18         <tr>
19           <th mochi:sortcolumn="PID int">PID</th>
20           <th mochi:sortcolumn="USER str">USUÁRIO</th>
21           [...]
22           <th mochi:sortcolumn="COMMAND str">COMANDO</th>
23         </tr>
24       </thead>
25       <!   substituído pelo conteúdo do arquivo JSON   >
26       <tbody class="mochi template">
27         <tr mochi:repeat="item domains">
28           <td mochi:content="item.PID"></ td>
29           <td mochi:content="item. USER"></td>
30           [...]
31           <td mochi:content="item. COMMAND"></td>
32         </tr>
33       </tbody>
34     </table>
35   </body>
36 </html>
```

Exemplo 3: Dados JSON

```
01 {
02   "columns": [ "PID", "USER","PR","NI", "VIRT", ➥"RES", "SHR",
➥"S", "CPU","MEM", "TIME", "COMMAND"],
03   "rows": [
04     [ "6620","cz","15","0","166912","57344","40960",
➥"S","11.6","5.6", "0:02.40", "soffice.bin"],
05     [ "3701","root","15","0","241664","225280","17408",
➥"S","4","21.8", "4:23.50", "X"],
06     [ "4496","cz","15","0","63828","20480","15360",
➥"R","2","2", "0:16.02", "gnome-panel"],
07     [ "4506","cz","15","0","70480","18432","10240",
➥"R","2","1.8", "0:04.14", "gnome-terminal"],
08   ]
09 }
```

A figura inteira será redimensionada para refletir o tamanho do texto, sem qualquer esforço por parte do desenvolvedor.

Os usuários também podem redimensionar facilmente a figura, na maioria dos navegadores, com a roda do mouse.

Ligue as linhas

Desenhar linhas de conexão é um pouco mais difícil. Como o JavaScript não tem suporte a elementos gráficos como linhas ou círculos, o truque nesse ponto é desenhar os elementos gráficos, pixel por pixel, como elementos `div`. O *jsGraph* de Walter Zorn abstrai esse processo complexo, dando ao desenvolvedor formas básicas como linhas, círculos e polígonos.

A função `connect()` se baseia nelas para desenhar duas caixas com linhas. Ela assegura as posições da caixa em tempo de execução para permitir que a imagem seja redimensionada para o tamanho da fonte.

A função `displayHierarchy()`, em seguida, redesenha as linhas de conexão.

O *TextResizeDetector*[15] de Lawrence Carvalho chama essa função sempre que o usuário altera o tamanho da fonte. O resultado é um widget AJAX que pode ser aproximado sem perdas, feito para estruturas de árvore que não podem ser implementadas com bitmaps.

Como ele se baseia inteiramente em elementos HTML, não há necessidade de extensões como Canvas ou VML que não estejam disponíveis para alguns navegadores. Pelo menos a caixa de texto é legível em navegadores sem suporte JavaScript e CSS.

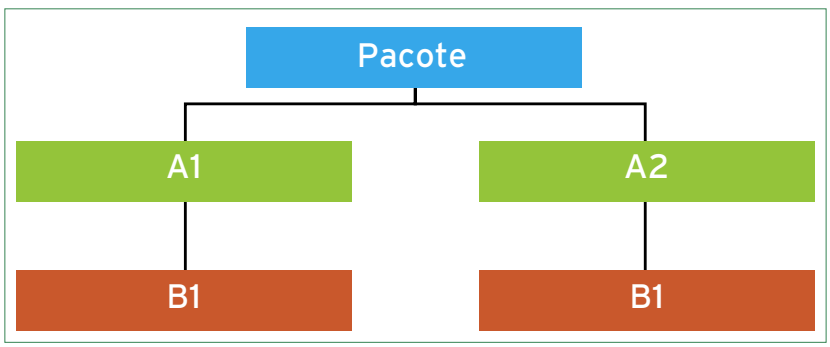


Figura 5 Melhor que bitmaps: A imagem em *JavaScript* reage dinamicamente a mudanças de tamanho de fontes e economiza banda.

Prós e contras

Muitos sites se beneficiam do uso de JavaScript e AJAX em relação à usabilidade. Menores tempos de resposta e a independência de recarregamento de páginas são muito bem recebidos pelos usuários. Entretanto, o uso de AJAX pode causar problemas que não ocorrem em páginas estáticas. Por exemplo, os usuários não podem simplesmente clicar nos botões *Avançar* e *Voltar* para navegar.

As páginas modificadas dinamicamente por JavaScript aumentam as expectativas dos usuários. O problema afeta até os simples menus dinâmicos citados acima. Se os usuários clicarem para abrir um submenu, o botão *Voltar* não os levará ao estado anterior da página; em vez disso, ele abrirá a página visitada anteriormente.

Isso talvez não seja um grande problema para um menu, mas se o script no lado cliente alterar substancialmente a página, fazendo-a parecer uma nova página do ponto de vista do usuário, é provável que haja confusão.

O navegador não consegue detectar as mudanças de estado que ocorrem numa página AJAX porque a URL continua a mesma. A lógica JavaScript do lado cliente aplica as mudanças sem recarregar, o que explica a incapacidade do navegador em capturar o status da página. Se o website utilizar a navegação baseada em AJAX, os bookmarks simplesmente levarão o usuário à página inicial.

A primeira coisa a ser levada em conta é o que é mais útil para o usuário: um histórico funcional e a capacidade de adicionar subpáginas aos favoritos ou a resposta

rápida. Apesar de ser sempre elegante o uso de código JavaScript no lado cliente para abrir menus ou reordenar tabelas sem recarregar páginas, a usabilidade de uma página de loja online ou catálogo com centenas de subpáginas seria seriamente afetada caso os usuários perdessem a capacidade de navegar com os botões de avanço e retrocesso.

Soluções

Há soluções para os problemas do histórico e dos favoritos. Por exemplo, o Google Maps fornece uma URL alternativa com os parâmetros do `GET` para a seção específica do mapa.

Os usuários não conseguem adicionar aos favoritos a página carregada no navegador. Em vez disso, um clique direito no link mostrado na página adiciona um favorito.

Outras soluções usam a âncora HTML normalmente usada para armazenar posições específicas de uma página em uma URL. A âncora é a parte da URL após o sinal de tralha (#).

Assim como a própria URL, a âncora pode ser modificada por meio do JavaScript sem recarregar a página. Se o navegador não conseguir encontrar uma tag de âncora para o texto após a tralha, a tela permanecerá inalterada. A âncora, portanto, é perfeita para guardar informações de estado: a parte da âncora é a única na URL que pode ser modificada sem exigir o recarregamento.

As capacidades de imagens em JavaScript são bem espartanas quando comparadas ao Flash: HTML e CSS só conseguem desenhar quadrados e texto. Bibliotecas como a jsGraphics adicionam outras formas, como círculos e polígonos. Imagens SVG ou Canvas, um elemento do futuro padrão web HTML 5, adicionam mais possibilidades quando embutidas na página. Porém, nenhuma delas é adequada para sites publicamente acessíveis, pois não estão disponíveis para todos os navegadores web. ■

Mais informações

[1] jEdit: http://www.jedit.org

[2] NetBeans: http://www.netbeans.org

[3] Plugin Web Developer para o Firefox: http://chrispederick.com/work/web-developer

[4] Document Object Model: http://www.w3.org/DOM

[5] Setas Unicode: http://www.alanwood.net/unicode/arrows.html

[6] Padrão ECMA: http://www.ecma-international.org/publications/standards/Ecma-262.htm

[7] MochiKit: http://www.mochikit.com

[8] SVG: http://www.w3.org/Graphics/SVG

[9] VML: http://www.w3.org/TR/1998/NOT-VML-19980513

[10] Inkscape: http://www.inkscape.org

[11] Karbon: http://www.koffice.org/karbon

[12] Elemento Canvas do HTML: http://www.w3.org/html/wg/html5/#the-canvas

[13] SVG2VML: http://code.google.com/p/svg2vml

[14] ExplorerCanvas: http://excanvas.sourceforge.net

[15] TextResizeDetector: http://www.alistapart.com/articles/fontresizing

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	✔				✔	✔

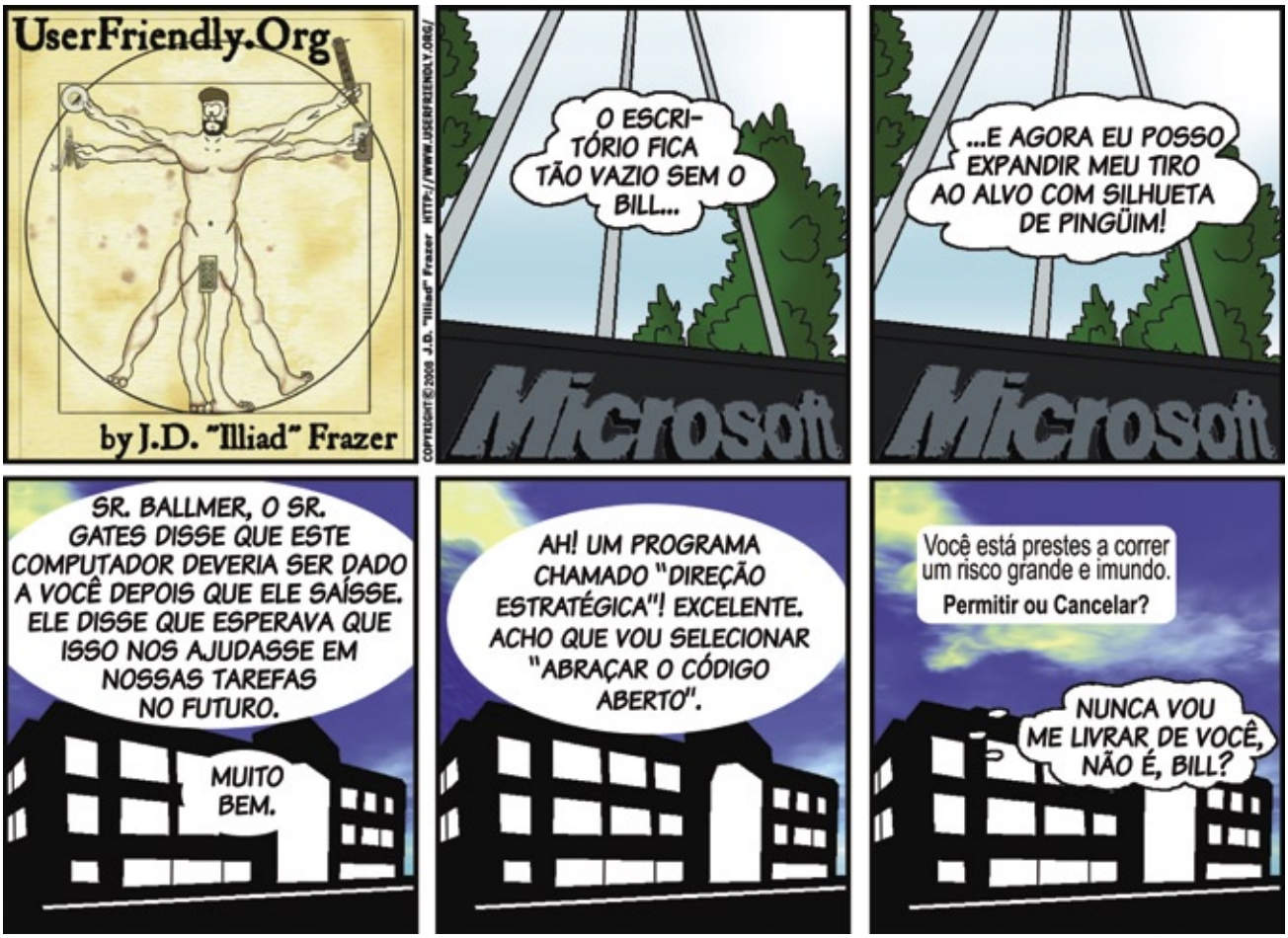
Empresa	Cidade	Endereço	Telefone	Web	1	2	3	4	5	6
São Paulo (continuação)										
Delix Internet	São José do Rio Preto	Rua Voluntário de São Paulo, 3066 9º – Centro – CEP: 15015-909	11 4062-9889	www.delixhosting.com.br	✔		✔		✔	
4Linux	São Paulo	Rua Teixeira da Silva, 660, 6º andar – CEP: 04002-031	11 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	Rua Santa Catarina, 1 – Tatuapé – CEP: 03086-025	11 6097-3000	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	✔	✔			✔	

Calendário de eventos

Evento	Data	Local	Website
PyConBrasil 2008	18 a 20 de setembro	Rio de Janeiro, RJ	www.pyconbrasil.com.br
Linux Park	23 de setembro	Brasília, DF	www.linuxpark.com.br
Festival Software Livre DF	3 e 4 de outubro	Brasília, DF	www.festivalsoftwarelivre.org/evento
Rails Summit Latin America	15 e 16 de outubro	São Paulo, SP	www.locaweb.com.br/rails
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

Índice de anunciantes

User Friendly – Os quadrinhos mensais da Linux Magazine



LINUXPARK 2008

O ECOSSISTEMA DE NEGÓCIOS EM SOFTWARE LIVRE NO BRASIL

O Linux Park 2008 é o evento que vai definir o futuro do mercado de tecnologias abertas no Brasil. Compareça e compartilhe suas experiências com os principais decisores e influenciadores do mercado.

Para mais informações, visite o site:
www.linuxpark.com.br

- Modelos de negócios com Software Livre em vários segmentos
- Cases de sucesso
- Keynotes
- Conteúdo específico para o segmento de revendas de Software Livre
- Provas de certificação

Patrocínio Diamond: Itautec

Patrocínio Gold: QualityWare, Kenos Sistemas de Gestão Integrada

Organização e realização: Linux New Media, The Pulse of Open Source

Promoção: Linux Magazine

PREVIEW

Na Linux Magazine #47

DESTAQUE

Impeça ataques

Após aprender a consertar e reparar os danos causados pelo invasor, aprenda na próxima edição da Linux Magazine as técnicas para proteger sua rede dos cibercriminosos.

De rootkits a ataques de força bruta, é muito importante aprender como agem os invasores e como usar suas ferramentas para defender seus sistemas e, sempre que possível, desferir um contra-ataque.

Apresentaremos também alguns dos melhores sistemas de prevenção de invasão – os famosos IPSs. Essas sentinelas virtuais podem significar a diferença entre danos reais e um agressor frustrado. ■

REDES

Cluster Apache

Várias tecnologias suportam balanceamento de carga em servidores web. Balanceadores de carga vêm em todos os tamanhos e formatos, desde simples técnicas baseadas em DNS até sistemas proprietários vastos e versáteis. No entanto, em alguns casos, os recursos de balanceamento de carga necessários podem já estar disponíveis até no próprio Apache.

O servidor web mais usado do planeta já traz interessantes recursos, como *caching*, compressão, reescrita de URLs e processamento de cabeçalhos – na verdade, alguns dos módulos responsáveis por isso até já são carregados por padrão. ■

Na EasyLinux #13

DESTAQUE

Portáteis!

Os laptops e os notebooks sempre deram aos seus possuidores um certo glamour, status de pessoas antenadas e, claro, fama de gente que tinha "grana" – afinal, um bom portátil, até dois anos atrás, não custava nada barato. Com o passar do tempo, no entanto, fabricantes foram criando uma nova geração de notebooks: menores, sem partes mecânicas quebráveis em boa parte de suas versões e mais baratos. Esses aparelhinhos, chamados de sub-notebooks, vêm ganhando espaço, tanto entre os clientes "tradicionais" dos laptops quanto entre pessoas que jamais imaginaram ter um portátil. ■

OFICINA

Com jeito de cinema

A qualidade de aúdio e vídeo da TV digital, recentemente iniciada no Brasil, é um atrativo para quem deseja assistir a programas em alta resolução e até gravar alguns de seus preferidos. Se os receptores de TV ainda estão muito caros, receptores USB para computador oferecem a imagem de cinema e qualidade de som de DVD. E, o que é melhor, usando Linux. ■

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

Rails Summit Latin America

by LOCAWEB

OS MAIORES NOMES DO MUNDO EM RAILS ESTARÃO AQUI.

A Locaweb realiza no Brasil o Rails Summit Latin America, que completa o calendário de eventos Rails no mundo. São mais de 10 palestrantes internacionais e os melhores especialistas brasileiros reunidos durante dois dias para trocar informações e novidades sobre Ruby on Rails.

DAVID HEINEMEIR HANSSON - videoconferência
Criador do Ruby on Rails e sócio da 37signals, Hansson mudou a forma de pensar o desenvolvimento de aplicações em web, definindo a essência de produtos Web 2.0 com Ruby on Rails.

CHAD FOWLER
Responsável pelas atuais conferências RubyConf nos Estados Unidos, Fowler é um dos fundadores da organização RubyCentral e ajudou efetivamente a elevar a qualidade da comunidade Ruby.

OBIE FERNANDEZ
Participante ativo da comunidade Rails on Ruby, Fernandez foi consultor da renomada ThoughtWorks e criou recentemente a HashRocket. Também lançou o livro *The Rails Way*, um dos melhores sobre o assunto.

FABIO AKITA
Escritor do primeiro livro original em português sobre Rails, Akita é Gerente de Produtos Rails na Locaweb e um dos maiores difusores do tema no país.

CHARLES NUTTER E THOMAS ENEBO
Criadores do projeto JRuby, Nutter e Enebo são engenheiros da Sun e com seu trabalho alcançaram um nível de produtividade e integração sem precedentes no mundo Java.

SÃO PAULO, 15 E 16 DE OUTUBRO – ANHEMBI – AUDITÓRIO ELIS REGINA

VAGAS LIMITADAS. INSCREVA-SE JÁ.

Realização: LOCAWEB

www.locaweb.com.br/railssummit