

Nosso hacker correspondente **Rainer Brockerhoff** continua sua peregrinação internacional e comparece à maior e mais divertida reunião de MacGeeks do planeta.

Fotos: Jonathan "Wolf" Rentzsch

Um brasileiro na MachHack

Cheguei de Vancouver pouco antes da meia-noite. Para economizar, só reservei hotel a partir do meio-dia de quinta, 20 de junho. Felizmente, encontrei-me com **Daniel Steinberg** (www.oreillynet.com/pub/au/187), guru de Java e gente fina, que conheci durante o cruzeiro MacMania (ver edição anterior). Daniel ofereceu-se para guardar minhas malas e dividir o quarto nos dias seguintes, para recíprocas economias. Enquanto isso, tive que ficar acordado até aparecer uma vaga de quarto com duas camas... Felizmente tinha a meu favor o fuso horário de Vancouver. Ou seja, meu biorritmo ainda pensava que eram apenas 2 horas da madrugada em vez de 5 horas.

O maluco do Brasil

O tema deste ano da conferência anual **MachHack** (www.machhack.com) foi "Hakku no Tetsujin!" ("Hacker de Ferro" em japonês). Faixas vermelhas com o tema escrito em caracteres japoneses serviam para distinguir visualmente os membros da equipe do evento. Ao final do banquete de encerramento (mais detalhes adiante), a organizadora, Carol Lynn, distribuiu algumas extras, com cerimônia e tudo: "Ajoelhe-se, reles geek! Levante-se, Sir Iron Hacker!" Chegando, fui direto à recepção do evento. Como já é minha terceira vez, parece que já me conhecem (provavelmente como "o maluco que vem do Brasil"), e me receberam com abraços e felicitações pelo sucesso do XRay. (Aliás, esse meu humilde software recebeu posição de destaque, além de cinco ratinhos na página 63 da revista Macworld de julho...) Com algum atraso, comecei a keynote de



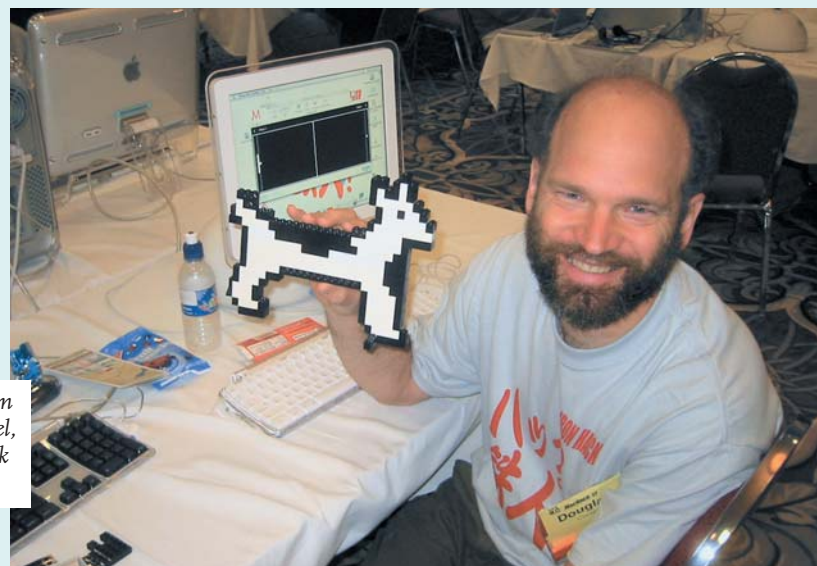
Tim O'Reilly (tim.oreilly.com), CEO da editora O'Reilly & Associates. Tim entrou acompanhado por um escocês em trajes típicos (e participante da MachHack), tocando gaita de fole. No seu discurso, contou como começou a trabalhar como escritor técnico e detalhou a evolução da sua empresa, que hoje é considerada a melhor editora de livros sobre computação. Tim é também usuário do Mac OS X e anunciou que provavelmente toda a editora será convertida para Mac OS X nos próximos 12 meses.

De resto, a keynote foi a mais curta dos anos recentes – começou à 1:00 h da madrugada e terminou às 4:15 h, ficando longe, por exemplo, das quase 6 horas que Steve Wozniak ficou contando "causos" no ano passado, ou das 6 horas e meia que Eric Raymond gastou no ano anterior. Também, o assunto não era tão interessante – que me perdoem os fãs da O'Reilly.

Neste ponto, embora não houvesse mais atividades "oficiais" até as 11 da manhã – quando começam as conferências – a sala de máquinas estava ativa, com cerca de 15 pessoas surfando a Internet ou jogan-

do em rede. O átrio do hotel tinha um pouco menos de gente. Aliás, o átrio e todos os auditórios foram equipados com Internet sem fio; conectei-me com meu iBook sem problemas. Este ano apresentei um trabalho chamado "Cocoa Plug-ins: Making Other People's Software Work for You". Estava marcado para as 15 h do primeiro dia, mas consegui transferir para o último dia – não tive tempo de preparar minha apresentação antes.

O tema "Iron Hacker" é uma sátira ao "Iron Chef" (www.foodtv.com/foodtv/show/0,6525,IC,00.html), um programa de TV bastante popular aqui, que parece consistir de competições entre cozinheiros. Em alusão a isso houve referências constantes ao mascote extra-oficial do Mac, **Clarus** (também conhecido como "dogcow", "vacadela" ou "Moof" – aquele bichinho que pode ser visto no diálogo de Page Setup do Mac OS clássico), e que seria o "ingrediente secreto" dos hacks. E boa parte dos hacks apresentados efetivamente continha imagens ou sons de Clarus. ▶



Douglas Clarke, com a ajuda do filho Nigel, de 7 anos, fez o hack Clarus All Over

MacHack

continuação

69 horas sem dormir

Minha tentativa de dividir esta reportagem em dias e enviar notícias à redação em intervalos regulares fracassou. Até as 12 h de domingo foram 84 horas contínuas, das quais dormi um total de 15. Fica-se no quarto o mínimo possível, por medo de perder algo ou de deixar de encontrar alguém importante. Apenas no dia seguinte ao encerramento oficial tive tempo para consolidar minhas anotações. Escrevi esta matéria sentado no salão de recepção do hotel. Um salão grande, com uma dúzia de mesas redondas e um conjunto separado de poltronas e sofás, rodeado de salas e salões para conferências e um restaurante. Cada mesa tem um hub Ethernet de 8 portas e dois filtros de linha; como quase todo mundo tem um cartão AirPort no seu computador, os hubs são usados só por quem precisa de mais velocidade.

Para quem não trouxe um iBook ou TiBook, há uma “sala de máquinas” com aproximadamente 50 iMacs – a maioria iMacs G3 – com alguns G4, 10 desktops G4 e quatro servidores. Seis estações AirPort foram espalhadas pelos salões de recepção e conferência, tudo conectado a uma linha DSL de 1,1 megabit.

Tudo funcionou muito bem. Enquanto eu escrevia, toda essa estrutura ia sendo desmontada e embalada; prometeram manter a rede no ar até o último instante possível. Ao meu redor, uma dúzia de pessoas com laptops redigia reportagens ou simplesmente surfava a Internet.

Mais estatísticas

Este ano, participaram 280 pessoas, contra 350 no ano passado; a queda foi menor do que se esperava. Mas uma pesquisa informal revelou que muita gente estava procurando emprego. Havia 53 estudantes (tradicionalmente chamados “yoots”, corruptela de “youths” = “jovens”) e 26 apresentadores. 41 voluntários organizaram e trabalharam na conferência. 12 trabalhos e outro tanto de sessões extras foram apresentadas.

Retornando à ordem cronológica: na noite de quinta assistimos à segunda keynote, desta vez de Rob Malda, mais conhecido como CmdrTaco (cmdrtaco.net), fundador e um dos operadores do site de notícias

Rob Malda (CmdrTaco), do SlashDot, deu a segunda keynote “falando mal de todos os sistemas operacionais”

e discussão SlashDot (<http://slashdot.org>). CmdrTaco, um jovem que fala numa velocidade de vários megabits por segundo, foi uma escolha controversa para essa keynote. O seu site é mais dirigido para Linux, embora recentemente tenha inaugurado uma seção sobre Mac (apple.slashdot.org). Ele trouxe um laptop IBM rodando Linux (e Windows XP para games), coisa talvez inédita numa keynote da MacHack. Depois de avisar que iria “falar mal de todo e qualquer sistema operacional” e resumir brevemente a história do SlashDot, ele ficou cerca de três horas respondendo a perguntas que iam do absurdo (“Quantos biscoitos cabem no Shea Stadium?” “Sete”) ao filosófico. Também contou como conheceu sua noiva Kathleen (www.fent.net; recomendo baixar a “vaca louca”) e vários outros casos engraçados. Finalmente, passou cerca de uma hora demonstrando sua versão customizada do jogo The Sims, que ele denominou “Black Widow Lesbian Killers Neighbourhood”. Hmmm...

Após um breve desmaio no meu quarto (“dormir” não se aplica muito bem), tivemos um dia de conferências. Assisti a algumas e fiz minha própria apresentação a toque de caixa. Foram especialmente interessantes as conferências sobre a arquitetura do iPod e algumas sobre extensões e técnicas de Objective-C.

Sabatinados

Acabei não assistindo, mas houve uma conferência de Keith Stattenfield, líder técnico da equipe do Mac OS 9, sobre “O Futuro do Mac OS 9”. Segundo quem assistiu – e um

pedaço disso está documentado no filme-resumo que vem no CD da MacHack – ele disse, mais ou menos:

“Não fui à WWDC, aí um colega me ligou e disse: ‘Steve matou o g!’ Eu disse ‘Hã? Vocês piraram?’ e ele: ‘Não, ele matou mesmo, caixão no palco e tudo o mais!’ Aí falei ‘Muito engraçado, um caixão! O Steve nem sabe que eu existo, pelamordedeus, e agora ele coloca o g num caixão? No palco? Hahaha!’... Bom, de qualquer maneira, eu vim dizer pra vocês: morreu! Acabou!! It’s DEAD!!!!”

Bom, depois disso ele relutantemente (“não estive aqui e vocês não me ouviram falar isso”) afirmou que ainda pode haver um ou outro updatezinho para acomodar novos Macs que hipoteticamente possam surgir, ou consertar algum bugzinho necessário para rodar o Classic sob o Jaguar. OK, eu não escrevi isso e vocês não leram, certo? Mais sobre o Keith no final deste relatório...

Logo depois houve a tradicional sessão “Bash Apple”, na qual os funcionários da Apple que estão na conferência se aglomeram no palco e respondem a perguntas ou insultos da audiência. Aliás, devo dizer que todos os funcionários da Apple foram extremamente amigáveis, dispostos a conversar por horas com qualquer um, dar dicas, ajudar a debugar qualquer coisa... e quase todos pagaram, do seu próprio bolso, para comparecer, sem qualquer ajuda financeira ou institucional da Apple.

Isto posto, Tim Holmes, presença constante em congressos de desenvolvedores, onde geralmente é o representante-mor da Apple, iniciou a sessão lembrando que todos eles esta-



vam ali absolutamente extra-oficialmente e que o único comentário que ele poderia fazer como funcionário da Apple seria que toda pergunta ou sugestão estava sendo anotada para transmissão posterior a “alguém responsável”. Disse também que, como nenhum dos presentes queria perder seu emprego, pedia aos jornalistas presentes para não publicarem os seus comentários. E, de quebra, disse: *“de qualquer modo, não estivemos aqui, mesmo!”* No que vou atendê-lo, com exceção de alguns comentários que acho reproduzíveis. Um programador se levantou e fez uma longa pergunta, que pode ser resumida como: *“A Apple pretende estender, no futuro, o Cocoa e o Mac OS X para incluir novas funções como A, B, C, etc...?”* Aí um dos funcionários presentes, cujo nome não revelarei, respondeu: *“Não. Depois do Jaguar acabamos, ficou pronto, todo mundo vai pra casa. Vocês podem ir embora agora... estamos olhando outros mercados. Que tal um iEspremedor?”* :-)

Outra: *“Vocês já tentaram convencer o Steve que obviamente X, Y ou Z?”* Resposta: *“Não. Próxima pergunta...?”* Aliás, eles respondem assim tão *vapt-vupt* que deve haver cursos específicos para isso em Cupertino...

Apesar disso, o que ouvimos a seguir, por quase duas horas, foram informações interessantes e que, mesmo sem revelar qualquer coisa publicável sobre futuros Macs ou coisas do gênero, me deram uma idéia muito melhor do que acontece na Apple – ou melhor, por que acontecem certas coisas.

TidBits

A seguir, fui assistir a conferência *“Hacking the Press”* (“Haqueando a Imprensa”) de Adam Engst, dono do ilustre boletim TidBits (www.tidbits.com). Era para durar uma hora, mas acabou se estendendo por mais de três horas. Adam explicou como, na sua opinião, um desenvolvedor deve interagir com a imprensa. Analisou casos reais de *press releases* e homepages de empresas de vários portes; depois, fez o mesmo com as empresas dos desenvolvedores presentes. Felizmente éramos apenas meia dúzia, senão teria levado ainda mais tempo. Não é necessário dizer que fiz muitas anotações e, no meu copioso tempo livre, vou seguir a maioria dos conselhos de Adam.

Sorteio de Macmanias

Exatamente à meia-noite, começou o famoso concurso de hacks. A parede que separa a sala de máquinas do salão principal foi recolhida e, de um lado, fez-se uma fila enorme de gente, com parafernalias variadas, para apresentá-los à platéia. Enquanto isto, alguns permaneceram na sala de máquinas, na audiência ou até mesmo na fila, fazendo um último ajuste

no seu programa ou apresentação.

Tem gente que trabalha o ano inteiro para levar um hack; outros formam uma equipe, geralmente com gente que não conhecia anteriormente, durante a conferência; alguns chegam a fazer o hack integralmente enquanto estão na fila!

Devo admitir que, novamente, me acovardei e preferi fazer contatos com amigos, assistir a algumas conferências, preparar minha própria apresentação e até dormir alguns minutos a apresentar um hack. Da próxima vez, quem sabe?

A cena foi bem complexa. Havia dois telões onde, nos intervalos, colocam breves vídeos e telas engraçadas ou esquisitas. Scott Boyd (www.machax.com), o mestre de cerimônias, fez comentários sarcásticos, apresentou veteranos, comentou sobre esta ou aquela tradição da MacHack e jogou (com força!) embalagens de *fast food* e camisetas na platéia. Portanto, era aconselhável ficar acordado; um pacote de Twinkies na cara pode ter certo impacto. Incrivelmente, até hoje não houve notícia de feridos, mas desta vez uma cadeira quebrou sob o peso de um participante mais corpulento pulando para agarrar uma camiseta. Aliás, um pacote de Twinkies, descrevendo uma parábola perfeita, alojou-se num dos dois candelabros do salão; enquanto escrevo isto, ainda parece estar lá. Há apostas sobre se o descobrirão antes da MacHack 2003, sobre qual classe de roedores ou insetos isso vai atrair (o consenso é que nenhuma barata que se preze comeria aquilo), ou se Scott vai “afinar” e pedir uma escada emprestada ao hotel.

Devo ressaltar que Scott também distribuiu uma dúzia de exemplares recentes da Macmania, que foi por ele apresentada como “a melhor revista de Mac do mundo!” e, depois que algumas capas foram colocadas num retroprojeto, a revista foi aplaudida com grande entusiasmo.



Tom Zerucha e Troy Benjegerdes com o hack AirPort Extender Blender. O liquidificador é acionado via AirPort, amplificado pela antena feita com uma lata vazia de batatas Pringles

Para o sorteio das Macmanias, Scott falava coisas como *“quem tem um número de inscrição primo?”* Como os números de inscrição têm 4 dígitos, não era imediatamente óbvio; mesmo assim, várias pessoas respondiam rapidamente e ganharam uma revista. Um, mais entusiasmado, foi interpelado para provar que seu número era primo, e (pasmê!) ele andava com uma tabela de todos os números primos de até 5 dígitos *no bolso!* Isso é que é geek...

Liquidificadores sem fio

Enquanto isso, havia um grupo de voluntários ajudando os participantes a “uploadar” seus programas para o servidor principal, montar os equipamentos, conectar projetores e tudo o mais para sua apresentação. Cada um podia trazer seu próprio laptop ao palco ou usar um dos iMacs e G4s colocados na mesa principal. Enquanto um apresentava um trabalho, o próximo já ia montando sua tralha do lado. E digo “tralha” de propósito. Não eram apenas programas. Muitos dos “yoots” presentes usaram o sistema Lego para seus hacks. Houve também hacks de hardware, como por exemplo o *“Airport Extender Blender”*, que parecia conter um laptop surpreendido em conjunção carnal com um emaranhado absurdo de placas e cabos, terminando numa antena direcional improvisada com uma lata de batatas fritas apontando para outro emaranhado de parafernália a alguns metros, que por sua vez culminava (não há outro verbo apropriado) num liquidificador que podia ser acionado – sem fio e via AppleScript – a partir do laptop.

Como eu disse lá no começo, muitos hacks envolviam Clarus, a *dogcow* mascote, devidamente documentada na *“TechNote 31”* (<http://developer.apple.com/products/techsupport/dogcow/tn31.html>). Aparecia como ícone de menus, galopava sobre a tela, mugia/latia em diversas circunstâncias etc. O hack de Nigel Clarke, um menino de 7 anos (e aparentando menos), consistia num Finder alterado para inserir Clarus em vários ícones no Dock –

MacHack

continuação

isso com a assistência de alguém para localizar onde esses ícones ficam. Quando ele, no final, mostrou uma lata de lixo cheia, derrubada, com o rabo de Clarus aparecendo de dentro da lata, foi aplaudido de pé. Outro engraçado foi o “Iron Hack” que reproduziu um famoso *sketch* de TV do Monty Python – um cara volta à loja de animais alegando que o papagaio recém-comprado está morto. Mas no caso, o comprador era Keith Stattenfield (claro!) comprando o Mac OS 9, apropriadamente representado numa caixa preta em vez de branca, e o vendedor ficava dizendo “mas eu vejo ele respirar” e Keith ficava repetindo “está morto!”

Do CD da MacHack (www.machack.com/BestHackCDForm.html) constam 63 hacks, e a competição foi até cerca de 5 horas da madrugada, então infelizmente não dá pra falar de todos. Alguns outros que gostei: *LCD Degauss*, que sacudia a imagem num iBook como acontece quanto se desmagnetiza um monitor com tubo de imagem; *iBacklight*, um plug-in para iTunes que pisca a maçã na tampa do iBook em ritmo com a música; *Somersault*, que faz os ícones do Dock darem uma enorme cambalhota enquanto se lança um aplicativo; e *AniMac*, que usa uma placa USB protótipo e vários motores para fazer um iMac G4 mexer com o braço, como aparece nos comerciais da Apple (este infelizmente não ficou pronto a tempo). Consegui ajudar alguns autores de hacks em Cocoa, o que me deixou bem satisfeito.

Num intervalo, quase no final, Adam Engst apareceu para contar sobre o famoso “bit” que vem sendo usado para testar o próprio hotel como sistema de armazenamento. No concurso de 2000, Adam executou um hack que foi contemplado pelos árbitros com um prêmio especial: uma tora de madeira de quase um metro e meio (vou explicar o sistema de premiação mais à frente). A intenção foi, claro, causar problemas no embarque de Adam no aeroporto. Mas ele descobriu que o console do banheiro, no quarto do hotel, tinha a dimensão exata para, *ummmh*, acomodar a tora sem deixá-la visível. Em 2001, Adam conseguiu miraculosamente reservar o mesmo quarto e – milagre! – a tora ainda estava lá, e prontamente virou objeto do novo hack *Hotel Bit Storage*. Antes de ele sair, a tora foi recolocada e, pasme, ainda estava lá em 2002! Dessa vez, o quarto foi ocupado por um incauto que prontamente foi designado para correr lá e verificar...

Primeiro lugar!

O último dia da MacHack começou um pouco mais tarde e devagar do que de costume. Como já tinha acabado o concurso de hacks, a sala de máquinas ficou relegada aos jogadores de Quake III, enquanto outros colocavam os emails e reportagens em dia. Apenas depois do almoço

ainda foram apresentadas algumas sessões.

Curiosamente, uma sessão que sempre foi enorme e concorrida dessa vez estava só com uma dúzia de gatos pingados – a “*Bash Metrowerks*”. O ambiente de desenvolvimento CodeWarrior, da Metrowerks, demorou para ser adaptado ao Mac OS X. A versão 8, recém-publicada, parece que finalmente recuperou o nível das versões clássicas, mas muita gente (inclusive eu) mudou para o ProjectBuilder. Aproveitei a calma para sair e fazer umas compras. Convenientemente, há lojas da CompUSA e Office Depot ao lado do hotel, e um shopping com uma livreria enorme a um quilômetro. Comprei expansões para meu iBook e um roteador *wireless* da SMC para colocar em casa – acostumei-me a essa história de poder acessar a Internet de uma poltrona sem fios. No início da tarde começou o banquete de encerramento. Podia-se escolher entre três pratos principais. Foi uma ocasião boa, também, para sentar e conversar com gente desconhecida. Enquanto serviam a sobremesa, vários membros da comissão organizadora fizeram discursos, publicando as estatísticas do evento, pedindo palmas para os outros membros da comissão, anunciando a comissão de 2003 e coisas do gênero.

Foi também anunciado o resultado da competição de trabalhos – no último dia os participantes votam, no que acharam mais relevante. A competição estava muito apertada. Foi a primeira vez que dois trabalhos empataram em primeiro lugar: o meu próprio “Cocoa Plug-ins” e o de Lisa Lippincott, “The Nitrogen Manifesto”.

Dois outros trabalhos empataram em segundo lugar, com apenas um voto de diferença! Os vencedores terão inscrição grátis na MacHack 2003. (Aliás, outro modo de não se pagar a inscrição é ter um trabalho aceito na competição.) Uma pequena interrupção foi causada pela circulação, na platéia, de um carrinho osten-



O australiano Quinn “The Eskimo!”, autor do Internet Config e vencedor do concurso com o hack FireStarter

tando uma criação chamada “*The Geek Dessert from Hell*”, uma bandeja contendo imitação de sorvete decorado com amostras aleatórias de todos os petiscos e “fast foods” – tudo colorido, fluorescente, iridescente etc. – que foram servidos durante a MacHack. A criadora Maurita Plouff, que na vida real é *chef* de restaurante, advertiu todos os presentes que a ingestão de qualquer pedaço dessa

sobremesa acarretaria a morte instantânea; e realmente ninguém se aventurou a tocar naquilo. A seguir, iniciou-se a distribuição de prêmios do concurso de hacks. Primeiro, houve os prêmios especiais para os “*yoots*”; todos ganharam uma camiseta extra, um livro de programação da O’Reilly e um poster especial do TiBook. A

empresa que publica o Fetch (software de FTP) deu um cachorrinho de pelúcia – o mascote que aparece no ícone do programa – para todos os participantes mais novos que o próprio software, criado em 1989.

A premiação mais louca do mundo

Os prêmios da competição necessitam uma explicação à parte. Desde a primeira MacHack, é tradicional: alguns membros da comissão visitam uma loja de ferragens próxima, chamada Duke’s Hardware, e compram itens obscuros, baratos, de preferência encailhados e empoeirados, cujo nome ou etiqueta aludem aos hacks. O primeiro prêmio é uma ratoeira chamada pelo seu fabricante de *A-Trap*; coincidentemente, é o mesmo nome que a Motorola usava para as chamadas de sistema nos antigos Macs 68K. A ratoeira é sempre gravada com o nome do vencedor. Nos primeiros anos, os vendedores do Duke’s quase enlouqueciam tentando adivinhar quem eram aqueles caras estranhos e por que insistiam em comprar coisas mais estranhas ainda – e nunca sabiam o que queriam, mas pediam para revistar a loja inteira. Mas nos anos recentes – inclusive, suponho, para evitar acionar o FBI e similares – o pessoal do



Duke's foi devidamente informado. Este ano, o Duke's comemorou 50 anos de existência e foi devidamente homenageado com uma placa, entregue à esposa do velho Duke, que não pôde vir por motivos de saúde. Ela achou tudo muito estranho, mas ficava exclamando "então era pra isso!..." :-)

Primeiro, foram distribuídos alguns prêmios especiais. Os autores do hack *AirPort Extender Blender* – aquele que aciona um liquidificador à distância sem fio – ganharam um forno de microondas e o desafio de incorporá-lo num hack em 2003. O autor de *Informed Choice* ganhou uma edição universitária do Metrowerks CodeWarrior versão 8; este hack, que alterava o Chooser do System 7.6 para assinalar servidores de AppleTalk sobre TCP/IP, normalmente incompatíveis com o 7.6, foi feito usando CodeWarrior versão 2 num PowerBook Duo 280 rodando System 7.6!

A seguir, entrou-se nos prêmios "normais", se é que se pode usar essa palavra. Para ilustrar a mente poluída dos organizadores, nada melhor que o prêmio dado aos autores do hack *Open Girl*. Esse hack mostra um cubo giratório, programado em OpenGL, que baixa da Internet fotos de garotas em maiôs e as aplica às faces do cubo. O prêmio foi uma geringonça designada, na etiqueta, de "No Smoke 2-Stroke Oil & Adjustable Rod Holder", um nome passível de várias interpretações pornográficas. O jornalista **Andy Ihnatko**, cujo hack consistiu em um discurso inflamado (e muito engraçado) sobre seu absoluto merecimento de um prêmio, ganhou uma geringonça chamada "Heavy-Duty Snap-On Meltguard" – ou seja, algo para proteger não-sei-o-quê de derretimentos.

O hack *Metadata*, de Allon Stern, que inclui os códigos de Type e Creator na extensão dos arquivos, foi premiado com meia dúzia de lixas de unha com cabo "Aqua".

Dúzias de prêmios similares foram distribuídos, com alguns autores recebendo um bônus extra: um sabonete cafeinado, especial para hackers. (Dizem que a absorção da cafeína pela pele é mais eficiente que por via oral...) Finalmente, Scott anunciou que este ano, excepcionalmente, seriam distribuídos dois "A-Traps", pois consideraram dois hacks como tecnicamente empatados. O primeiro foi entregue a Adam Atlas, um jovem de 12 (!) anos que, além de escrever um programa em Cocoa que consulta automaticamente uma lista de servidores e deles abstrai uma lista-gem clicável de manchetes, ainda ajudou vários outros participantes – a maioria adultos! – a resolver problemas técnicos do Cocoa. Adam foi aplaudido de pé, e ainda ganhou uma cópia do Photoshop 7, três livros sobre programação e uma inédita bolsa para que possa participar da MacHack 2003 sem custos!

O A-Trap "tradicional" foi entregue a **Quinn "The Eskimo!"** Ele é uma figura bem conhecida na comunidade. Há vários anos, emigrou da Austrália para ir trabalhar na Apple. Foi o criador do Internet Preferences do Mac OS clássico e co-autor do programa de FTP Anarchie (hoje Interarchy), entre outras coisas.

O hack de Quinn foi o FireStarter; é um programa em Open Firmware que joga aquele efeito de "fogo" do QuickTime direto na RAM de vídeo de qualquer máquina, mesmo que esta tenha sido iniciada de um CD-ROM, através da interface FireWire.

Quase inacreditável, demonstra fartamente o profundo conhecimento que Quinn tem das tecnologias do Mac.

Tumulto no cinema

Encerrado o banquete, a maioria dos participantes foi a um cinema próximo para assistir ao filme "The Sum Of All Fears".

Como de costume, bloqueou-se o cinema inteiro para que os hackers pudessem conversar à vontade, comentar sobre o filme e gritar palavras sarcásticas sobre este ou aquele absurdo que aparecia na tela. Via de regra, quanto pior o filme, mais engraçados os comentários. Desta vez, como eu tinha lido o livro, minha apreciação da ruindade do filme foi prejudicada.


A melhor parte do filme vem depois do retorno ao hotel. Enquanto se serve o também tradicional sorvete da meia-noite, Keith Stattenfeld (sim, aquele que é gerente do defunto Mac OS 9) encena ao vivo uma edição do seu programa de TV "Keith Explains" (www.keithexplains.com), que vai ao ar semanalmente em Cupertino. No caso, Keith explica a todos o que acabam de ver no cinema. "...Agora, aparece o vilão! Como sabemos que é o vilão? Porque ele tem RUGAS! Sim! TODOS os vilões neste filme têm rugas proFUNdas na face!! Os mocinhos têm a pele mais lisa que bunda de nenê, mas os vilões têm RUGAS!!!..." E por aí vai, ao longo de meia hora. Qualquer inconsistência, clichê ou ilogicidade hollywoodiana é dissecada sem dó nem piedade.

E com isso, encerrou-se a MacHack 2002 e minha reportagem. O CD contendo os hacks, trabalhos e apresentações, e um conjunto de fotos e filmes, está à venda no site www.machack.com, onde também se acham mais detalhes sobre o evento. Até o ano que vem... **M**

Metendo a mão no Unix

Na lição anterior, aprendemos um pouco sobre o shell, e agora vamos descobrir um pouco mais sobre ele e conhecer os redirecionamentos e os *pipes*.

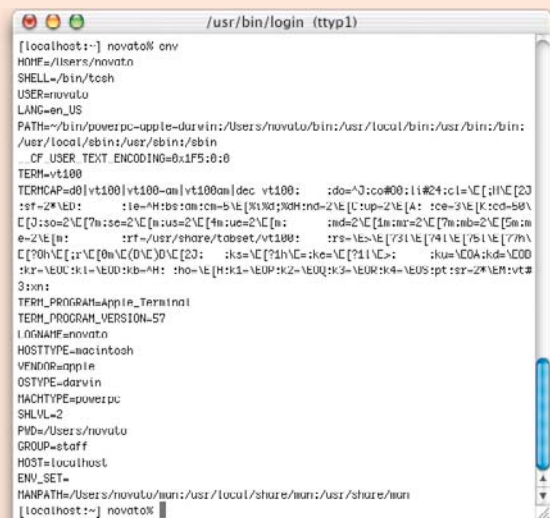
Já que vimos quais são os shells do Mac OS X. Vamos verificar o arquivo que os define, vendo rapidamente o conteúdo do arquivo `/etc/shells`. Para isso, usa-se o comando `cat`:



```
~/usr/bin/login (tty1)
Welcome to Darwin!
[localhost:~] novato% cat /etc/shells
# List of acceptable shells for chpass(1).
# Ftprd will not allow users to connect who are not using
# one of these shells.

/bin/bash
/bin/csh
/bin/eh
/bin/lcsh
/bin/zsh
[localhost:~] novato% █
```

Podemos confirmar que o arquivo de shells confere com o que aprendemos anteriormente. Vamos analisar também alguns aspectos de nosso shell, utilizando o comando `env`:



```
~/usr/bin/login (tty1)
[localhost:~] novato% env
HOME=/Users/novato
SHELL=/bin/tcsh
USER=novatu
LANG=en_US
PATH=/bin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/sbin:/usr/bin:/usr/sbin:/usr/local/bin/...
[localhost:~] novato% █
```

Os dados acima podem parecer estranhos, mas é muito importante compreender como o seu sistema está configurado para posteriormente alterá-lo e fazer com que ele faça exatamente o que você deseja. Analise cuidadosamente os termos em maiúsculas e tente reconhecer alguns deles. Um exemplo fácil de reconhecer é o `TERM_PROGRAM` (Programa de Terminal), que está definido como `Apple Terminal` (versão 57).

Observação – Um fato interessante que podemos notar é a presença dos parâmetros `USER` e `LOGNAME`. Até certa época, os sistemas Unix possuíam ou um ou outro desses dois parâmetros, sendo necessário que os programas fossem criados de modo a verificar ambos os parâmetros para descobrir qual deles estaria disponível. Os sistemas **MacPRO-52**

mas baseados no BSD utilizavam `LOGNAME`, enquanto os baseados no SVR3 utilizavam `USER`. Notamos então que houve uma evolução por parte do Unix em geral.

Podemos adicionar outros parâmetros não-presentes no Mac OS X, mas que podemos configurar para obter um melhor resultado na utilização do shell. Um exemplo é o parâmetro `EDITOR`, que define o editor de texto preferencial do usuário (se você é dos que acham que o `vi` é o Cão Chupando Manga). Mas por enquanto vamos nos limitar a reconhecer os parâmetros já existentes. É sempre bom lembrar que o Unix em geral é *case sensitive* – isto é, digitar um comando em maiúsculas ou minúsculas afeta o resultado.

Redirecionamento de arquivos

Até agora, todos os comandos que aprendemos recebem informações pela linha de comando e retornam um resultado na tela.

Na verdade, o Unix é muito mais poderoso do que apenas isso. Toda entrada pode vir tanto do teclado como de um arquivo, assim como o resultado pode ser exibido na tela como armazenado em um arquivo.

E como podemos tirar proveito dessa função? Bem... Essa é uma questão complexa, pois isso vai depender muito da sua criatividade!! Você pode, por exemplo, adicionar uma linha ou comandos a um arquivo de texto

sem precisar abrir um editor de texto no Terminal, executar uma sequência de comandos consecutivos, entre muitas outras possibilidades.

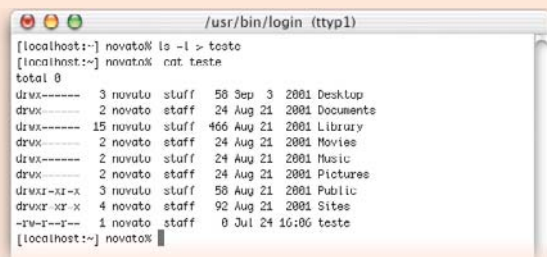
1. Redirecionamento de saída

No redirecionamento de saída, utilizaremos um arquivo como saída de resultados. Para isso utilizamos os comandos `>` e `>>`. O `>` é usado para indicar que a saída do programa deve ser redirecionada para um arquivo. O arquivo será criado se já não existir, ou seu conteúdo será totalmente substituído pelo novo conteúdo da saída, se existir. O `>>` produz o mesmo resultado que `>`, com a única diferença de não sobrescrever o conteúdo do arquivo. Apenas adiciona o resultado

da saída ao fim do conteúdo do arquivo. Vamos testar essa função. Digite:

```
[localhost:~] novato% ls -l > teste
```

Veja que nenhum resultado é apresentado na tela. Mas utilizando o `vi` podemos abrir o arquivo `teste` recém-criado e visualizar o resultado. No entanto, vamos aprender mais um novo comando. Para a visualização rápida do conteúdo do arquivo `teste`, utilizaremos o comando `cat`:

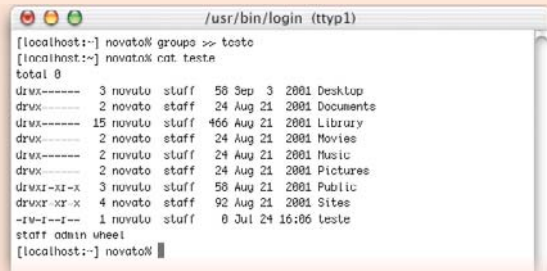


```
~/usr/bin/login (tty1)
[localhost:~] novato% ls -l > teste
[localhost:~] novato% cat teste
total 0
drwx----- 3 novatu stuff 58 Sep 3 2001 Desktop
drwx----- 2 novatu staff 24 Aug 21 2001 Documents
drwx----- 15 novatu stuff 466 Aug 21 2001 Library
drwx----- 2 novatu staff 24 Aug 21 2001 Movies
drwx----- 2 novatu stuff 24 Aug 21 2001 Music
drwx----- 2 novatu staff 24 Aug 21 2001 Pictures
drwx----- 3 novatu stuff 58 Aug 21 2001 Public
drwx----- 4 novatu staff 92 Aug 21 2001 Sites
-rw-r--r-- 1 novatu staff 0 Jul 24 16:06 teste
[localhost:~] novato% █
```

O que temos no arquivo `teste` é exatamente o que deveríamos ter obtido na tela do Terminal. Mas, devido ao redirecionamento na saída, o resultado é inserido e apresentado no arquivo `teste` ao invés de na tela do Terminal. Vamos agora inserir mais alguma coisa ao arquivo `teste`.

```
[localhost:~] novato% groups >> teste
```

Utilize o `cat` novamente para verificar a alteração no arquivo `teste`:



```
~/usr/bin/login (tty1)
[localhost:~] novato% groups >> teste
[localhost:~] novato% cat teste
total 0
drwx----- 3 novatu stuff 58 Sep 3 2001 Desktop
drwx----- 2 novatu staff 24 Aug 21 2001 Documents
drwx----- 15 novatu stuff 466 Aug 21 2001 Library
drwx----- 2 novatu staff 24 Aug 21 2001 Movies
drwx----- 2 novatu stuff 24 Aug 21 2001 Music
drwx----- 2 novatu staff 24 Aug 21 2001 Pictures
drwx----- 3 novatu stuff 58 Aug 21 2001 Public
drwx----- 4 novatu staff 92 Aug 21 2001 Sites
-rw-r--r-- 1 novatu stuff 0 Jul 24 16:06 teste
staff admin wheel
[localhost:~] novato% █
```

Podemos verificar a adição de uma linha com o resultado do último comando executado.

Observação – Se utilizássemos o comando `>`, ao invés de adicionarmos teríamos eliminado todo o conteúdo preexistente, substituindo-o pelo resultado da saída do último comando executado.

2. Redirecionamento de entrada

No redirecionamento de entrada, utilizaremos um arquivo como entrada de dados, assim

como utilizamos o teclado até agora. Para isso o comando é `<`.

O `<` é usado para indicar que a entrada do comando ou programa deve *receber* o conteúdo de um arquivo. (O arquivo obrigatoriamente já deve existir.)

Podemos utilizar o redirecionamento de entrada para obter informações diretamente de arquivos previamente criados. Assim, é possível reutilizar informações por quantas vezes forem necessárias sem necessidade de digitá-las novamente.

Pipe (sequência de comandos)

O *pipe* faz com que, em uma mesma linha, comandos separados sejam executados em sequência, de forma tal que a saída de um comando seja a entrada do seguinte.

O *pipe* é representado pelo caractere `|`. O formato do comando é:

`comando1 | comando2 | comando | comandoN`

Vejamos:

```

/usr/bin/login (tty1)
[localhost:~] novato% ls -l | sort -r | grep D
Documents
Desktop
[localhost:~] novato%
    
```

Aqui temos um exemplo em que o resultado é uma relação de todos os arquivos e diretórios que contenham a letra D maiúscula, apresentados em ordem inversa.

```

/usr/bin/login (tty1)
[localhost:~] novato% ps aux | grep od
111 ?? Ss  0:01.77 autodiskmount
223 ?? Ss  0:01.02 netinfod
273 ?? S  0:00.02 nfsiod
274 ?? S  0:00.00 nfsiod
275 ?? S  0:00.00 nfsiod
276 ?? S  0:00.01 nfsiod
[localhost:~] novato%
    
```

Este outro é um exemplo em que o resultado é uma relação dos processos

sendo executados pelo Mac OS X que contenham `od` no seu nome.

Conclusão

Você viu alguns comandos novos, como `cat` (visualizar arquivos texto rapidamente), `sort` (organizar linhas em arquivos de texto), `grep` (definir um

padrão na visualização), assim como um pouco mais sobre o shell, sobre redirecionamentos e *pipes*.

Muito se pode fazer com todos esses comandos, mas apenas a experiência de uso será realmente capaz de fazer você compreender todas essas possibilidades.

Estaremos sempre apresentando mais exemplos práticos de cada um dos comandos demonstrados com o decorrer de nossas lições de Unix. **M**

ALBERTO V. M.

Agradecimentos especiais a Oswaldo V. C. Bueno.