Explicando o BSD

Resumo

No mundo do código aberto, a palavra "Linux" é praticamente sinônimo de "Sistema Operacional", mas ele não é o único sistema operacional UNIX® de código aberto.

Então, qual é o segredo? Por que o BSD não é mais conhecido? Este artigo aborda esta e outras questões.

Ao longo deste artigo, as diferenças entre BSD e Linux serão destacadas *desta forma*.

Índice

1. O que é o BSD? 1
2. O que, um verdadeiro UNIX®? 2
3. Por que o BSD não é mais conhecido? 3
4. Comparando BSD e Linux 3

1. O que é o BSD?

BSD é a sigla para "Berkeley Software Distribution". É o nome do código fonte distribuído pela Universidade da Califórnia, Berkeley, que era originalmente uma extensão do UNIX® desenvolvido pela área de pesquisa da AT&T. Diversos projetos de sistemas operacionais de código aberto foram baseados em uma versão deste código, conhecido como 4.4BSD-Lite. Além disso, eles incluem vários pacotes de outros projetos de código aberto, com destaque para os do projeto GNU. O sistema operacional geralmente abrange:

- O kernel BSD, que lida com o agendamento de processos, gerenciamento de memória, multi processamento simétrico (symmetric multi-processing ou SMP), drivers de dispositivos, etc.
- A biblioteca C, a API base do sistema.

  A biblioteca C do BSD é baseada no código de Berkeley, não no projeto GNU.

- Utilitários como shells, gerenciadores de arquivos, compiladores e linkers (conversores de arquivos compilados em executáveis).

  Alguns utilitários são derivados do projeto GNU, outros não são.

- O sistema X Window, que gerencia a interface gráfica.

  O sistema X Window usado na maioria das versões do BSD é mantido pelo Projeto X.Org. O FreeBSD permite ao usuário escolher a partir de uma variedade de ambientes de desktop, tais como o Gnome, KDE ou Xfce; e gerenciadores gráficos (gerenciadores de janelas) mais leves,

como Openbox, Fluxbox ou Awesome.

- Diversos outros programas e utilitários.

2. O que, um verdadeiro UNIX®?

Os sistemas operacionais BSD não são cópias ou clones, mas sim derivações de código aberto do sistema operacional UNIX® da AT&T, que também é o ancestral do moderno UNIX® System V. Isto pode surpreendê-lo. Como isto é possível, uma vez que a AT&T nunca liberou seu código como código aberto?

É verdade que o UNIX® da AT&T não é um sistema de código aberto e no sentido de licenciamento o BSD definitivamente *não é* UNIX®, mas por outro lado a AT&T importou fontes de outros projetos, principalmente do Grupo de Pesquisa de Ciências da Computação (Computer Sciences Research Group ou CSRG) da Universidade da Califórnia em Berkeley, CA. A partir de 1976, o CSRG começou a liberar fitas de seu software, chamando ele de *Berkeley Software Distribution* ou *BSD*.

As primeiras distribuições do BSD consistiam principalmente em programas de usuários mas isso mudou radicalmente quando o CSRG firmou um contrato com a Agência de Pesquisa de Projetos de Defesa Avançados (Defense Advanced Research Projects Agency ou DARPA) para atualizar os protocolos de comunicação de sua rede, a ARPANET. Os novos protocolos ficaram conhecidos como *Internet Protocols*, posteriormente *TCP/IP* em virtude dos protocolos mais importantes. A primeira implementação amplamente distribuída foi parte do 4.2BSD, em 1982.

Durante a década de 80 surgiram muitas empresas produtoras de estações de trabalho. A maioria preferiu licenciar o UNIX® ao invés de desenvolver seus próprios sistemas operacionais. Em particular, a Sun Microsystems licenciou o UNIX® e implementou uma versão do 4.2BSD, a qual eles chamaram de SunOS™. Quando a própria AT&T começou a comercializar o UNIX®, eles começaram com uma implementação de certa forma simples chamada de System III, que logo transformou-se no System V. O código base do System V não incluía comunicação em rede, então todas as implementações incluíram software adicional do BSD, inclusive o software TCP/IP, e também utilitários como o shell *csh* e o editor *vi*. Estes aprimoramentos ficaram conhecidos como *Berkeley Extensions*.

As fitas do BSD continham código fonte da AT&T e portanto necessitavam da licença dos fontes do UNIX®. Por volta de 1990, os recursos financeiros do CSRG's estavam acabando, e ele foi encerrado. Alguns membros do grupo decidiram liberar o código do BSD, que era código aberto, sem o código proprietário da AT&T. Isso finalmente aconteceu com a fita *Networking Tape 2*, também conhecida como *Net/2*. O Net/2 não era um sistema operacional completo: faltava cerca de 20% do código fonte do kernel. Um dos membros do CSRG, William F. Jolitz, escreveu o código que faltava e o liberou no início de 1992 sob o nome de *386BSD*. Ao mesmo tempo, um outro grupo de ex integrantes do CSRG formaram uma empresa comercial chamada Berkeley Software Design Inc. e liberaram uma versão beta de um sistema operacional chamado BSD/386, o qual era baseado nos mesmos fontes. Mais tarde o nome deste sistema operacional foi alterado para BSD/OS.

O 386BSD nunca chegou a ser um sistema operacional estável. Ao invés disso, dois outros projetos surgiram a partir dele em 1993: NetBSD e FreeBSD. Este dois projetos divergiam originalmente na questão da espera pelas melhorias no 386BSD: o pessoal do NetBSD iniciou no começo daquele ano,

e a primeira versão do FreeBSD não ficou pronta antes do final do ano. Neste meio tempo o código base ficou diferente um do outro o suficiente para tornar difícil sua fusão. Além disso, os projetos tinham objetivos diferentes, como veremos adiante. Em 1996, o OpenBSD surgiu a partir do NetBSD, e em 2003, o DragonFlyBSD surgiu a partir do FreeBSD.

3. Por que o BSD não é mais conhecido?

Por uma série de razões, o BSD é relativamente desconhecido:

1. Os desenvolvedores do BSD estão mais interessados em aprimorar o seu código do que em divulgá-lo.
2. Grande parte da popularidade do Linux se deve a fatores externos ao projeto Linux, como a mídia e empresas que foram criadas para prover serviços Linux. Até pouco tempo atrás os BSD de código aberto não tinham este tipo de proposta.
3. Em 1992 a AT&T processou a BSDI, que comercializava o BSD/386, alegando que o produto continha código protegido por direitos autorais da AT&T. O caso foi encerrado fora dos tribunais em 1994, mas o fantasma do litígio continua assombrando. Em março de 2000 um artigo publicado na web afirma que o caso foi "recentemente encerrado".

   Um detalhe que o processo civil não deixa claro refere-se ao nome: nos anos 80 o BSD era conhecido como "BSD UNIX®". Com a eliminação dos últimos vestígios do código da AT&T do BSD, ele também perdeu o direito ao nome UNIX®. Desta forma você verá referências em livros para o "sistema operacional 4.3BSD UNIX®" e o "sistema operacional 4.4BSD".

4. Comparando BSD e Linux

Então, qual é realmente a diferença entre, digamos, o Debian Linux e o FreeBSD? Para a maioria dos usuários, a diferença é surpreendentemente pequena: Ambos são sistemas operacionais estilo UNIX®. Ambos são desenvolvidos por projetos não comerciais (isto não se aplica a diversas outras distribuições Linux, é claro). Nas próximas sessões vamos olhar para o BSD e compará-lo ao Linux. As descrições se aplicam principalmente ao FreeBSD, que representa aproximadamente 80% das instalações de BSD, mas as diferenças do NetBSD, OpenBSD e Dragon FlyBSD são pequenas.

4.1. Quem é o dono do BSD?

Nenhuma pessoa ou empresa é proprietária do BSD. Ele é criado e distribuído por uma comunidade de colaboradores altamente técnica e comprometida espalhada ao redor do mundo. Alguns dos componentes do BSD são projetos de código aberto com seus próprios licenciamentos e gerenciados por diferentes mantenedores.

4.2. Como o BSD é desenvolvido e atualizado?

Os kernels dos BSDs são desenvolvidos e atualizados seguindo o modelo de desenvolvimento Open Source. Cada projeto mantém uma *árvore com código fonte* publicamente acessível, que contém todo o código fonte do projeto, incluindo documentação e outros arquivos incidentais. Os usuários

podem obter uma cópia completa de qualquer versão.

Um grande número de desenvolvedores por todo o mundo contribuem com melhorias ao BSD. Eles estão divididos em três categorias:

- *Contributors* escrevem código ou documentação. Eles não têm permissão para adicionar código diretamente ao repositório principal de código fonte. Para que seu código seja incluído no sistema, ele deve ser revisado e verificado por um desenvolvedor registrado, conhecido como *committer*.
- *Committers* são desenvolvedores com acesso de gravação no repositório principal de código fonte. Para se tornar um committer, um indivíduo deve mostrar habilidade na área em que está ativo.

  Fica a critério do bom senso individual de cada committer a decisão se eles devem obter ou não um consenso antes de enviar alterações para o repositório de código fonte. Em geral, um committer experiente pode fazer alterações que sejam inquestionavelmente corretas sem obter consenso. Por exemplo, um committer do projeto de documentação pode corrigir erros tipográficos ou gramaticais sem revisão. Por outro lado, espera-se que os desenvolvedores que realizam mudanças complexas ou muito extensas enviem suas alterações para revisão antes de enviá-las para o repositório de código fonte. Em casos extremos, um membro do Core Team com uma função tal como a de arquiteto principal, pode ordenar que as alterações sejam removidas do repositório, num processo conhecido como *backing out*. Todos os committers recebem emails que descrevem cada commit individual, portanto não é possível enviar alterações para o repositório de código fonte em segredo.

- O *Core Team*. O FreeBSD e o NetBSD possuem uma equipe principal (Core team) que gerenciam o projeto. As equipes principais evoluíram ao longo dos projeto e a sua função nem sempre está bem definida. Não é necessário ser um desenvolvedor para ser um membro da equipe principal, embora isto seja normal. As regras para a equipe principal variam de um projeto para o outro, mas no geral elas têm mais voz ativa sobre a direção do projeto do que os demais membros tem.

Esse arranjo difere do Linux de várias maneiras:

1. Ninguém controla o conteúdo do sistema. Na prática, essa diferença é superestimada, uma vez que o arquiteto principal pode exigir que o código seja removido ou substituído, e mesmo no projeto Linux, várias pessoas podem fazer alterações.
2. Por outro lado, *existe* um repositório central, um lugar único no qual você pode encontrar todo o código fonte do sistema operacional, incluindo todas as versões mais antigas.
3. Os projetos BSDs mantêm todo o "Sistema Operacional", e não apenas o kernel. Essa distinção é apenas marginalmente útil: nem o BSD e nem o Linux são úteis sem aplicativos. Os aplicativos usados no BSD são frequentemente os mesmos aplicativos usados no Linux.
4. Como resultado da manutenção formal de um único repositório Git com o código fonte, o desenvolvimento do BSD é claro e é possível acessar qualquer versão do sistema por número de release ou por data. O Git também permite atualizações incrementais no sistema: por exemplo, o repositório do FreeBSD é atualizado cerca de 100 vezes por dia. A maioria dessas mudanças é pequena.

4.3. Releases do BSD

O FreeBSD, o NetBSD e o OpenBSD fornecem o sistema em três diferentes "releases". Como no Linux, os releases recebem um número como 1.4.1 ou 3.5. Além disso, o número da versão tem um sufixo indicando sua finalidade:

1. A versão de desenvolvimento do sistema é chamada de *CURRENT*. O FreeBSD atribui um número a CURRENT, por exemplo, FreeBSD 5.0-CURRENT. O NetBSD usa um esquema de nomenclatura ligeiramente diferente e acrescenta um sufixo de uma única letra que indica mudanças nas interfaces internas, por exemplo, o NetBSD 1.4.3G. O OpenBSD não atribui um número ("OpenBSD-current"). Todo novo desenvolvimento no sistema entra neste branch.
2. Em intervalos regulares, entre duas e quatro vezes por ano, os projetos lançam uma versão *RELEASE* do sistema, a qual é disponibilizada por meio de CD-ROMs e por meio de download gratuito em sites FTP, por exemplo, OpenBSD 2.6-RELEASE ou NetBSD 1.4-RELEASE. A versão RELEASE destina-se a usuários finais e é a versão normal do sistema. O NetBSD também fornece *versões de correção* (Patch Releases) com um terceiro dígito, por exemplo, o NetBSD 1.4.2.
3. A medida que os erros são encontrados em uma versão RELEASE, eles são corrigidos e as correções são adicionadas ao repositório Git. No FreeBSD, a versão resultante é chamada de *STABLE*, enquanto no NetBSD e OpenBSD continua sendo chamada de versão RELEASE. Novos recursos menores também podem ser adicionados a essa branch após um período de teste na branch CURRENT. Patches de segurança e outras correções de bugs importantes também são aplicadas a todas as versões RELEASE suportadas.

Por outro lado, o Linux mantém duas árvores de código separadas: a versão estável e a versão de desenvolvimento. Versões estáveis têm um número de versão menor par, como por exemplo 2.0, 2.2 ou 2.4. Versões de desenvolvimento têm um número de versão menor ímpar, como por exemplo 2.1, 2.3 ou 2.5. Em cada caso, o número é seguido por um outro número que designa a release exata. Além disso, cada fornecedor adiciona seus próprios programas e utilitários de área de usuário, portanto, o nome da distribuição também é importante. Cada fornecedor de distribuição também atribui números de versão à distribuição, portanto, uma descrição completa seria algo como "TurboLinux 6.0 com kernel 2.2.14 ".

4.4. Quais versões do BSD estão disponíveis?

Em contraste com as numerosas distribuições do Linux, existem apenas quatro grandes distribuições BSD de código aberto. Cada projeto BSD mantém seu próprio repositório de código fonte e o seu próprio kernel. Porém na prática, parece haver menos divergências do código entre os projetos BSD do que no Linux.

É difícil categorizar os objetivos de cada projeto: as diferenças são muito subjetivas. Basicamente,

- O FreeBSD visa o alto desempenho e a facilidade de uso pelos usuários finais, e é um dos favoritos dos provedores de conteúdo da web. Ele pode ser executado em diversas plataformas e tem significativamente mais usuários do que os outros projetos.
- O NetBSD visa a máxima portabilidade: "é claro que roda o NetBSD". Ele pode ser executado em diversas plataformas de hardware, de palmtops até grandes servidores, e até mesmo já foi usado em missões espaciais da NASA. É uma escolha particularmente boa para rodar em

hardware antigo que não seja Intel®.

- O OpenBSD visa a segurança e a pureza de código: ele usa uma combinação do conceito de código aberto ao de revisões rigorosas de código para criar um sistema que seja comprovadamente correto, tornando-o a escolha preferida de organizações preocupadas com segurança, tais como bancos, bolsas de valores e departamentos do governo dos EUA. Tal como o NetBSD, ele pode ser executado em várias plataformas.
- O DragonFlyBSD tem como objetivo o alto desempenho e a escalabilidade sob todos os aspectos, desde um sistema de um único nó até um sistema altamente clusterizado. O DragonFlyBSD tem várias metas técnicas de longo prazo, mas o foco está em fornecer uma infraestrutura compatível com SMP que seja fácil de entender, manter e desenvolver.

Também existem dois sistemas operacionais BSD UNIX® que não são de código aberto, o BSD/OS e Mac OS® X da Apple:

- O BSD/OS foi o mais antigo dos sistemas derivados do 4.4BSD. Não era um sistema de código aberto, embora as licenças do código-fonte estivessem disponíveis a um custo relativamente baixo. Assemelhava-se ao FreeBSD de várias maneiras. Dois anos após a aquisição da BSDi pela Wind River Systems, o BSD/OS não conseguiu sobreviver como um produto independente. O suporte e o código-fonte ainda podem estar disponíveis por parte da Wind River, mas todo desenvolvimento novo está focado no sistema operacional embarcado VxWorks.
- O Mac OS® X é a versão mais recente do sistema operacional para os equipamentos Mac® da Apple®. O núcleo BSD deste sistema operacional, Darwin, está disponível como um sistema operacional de código aberto totalmente funcional para computadores x86 e PPC. No entanto, o sistema gráfico Aqua/Quartz e muitos outros aspectos proprietários do Mac OS® X continuam fechados. Vários desenvolvedores do Darwin também são committers do FreeBSD, e vice-versa.

4.5. Como a licença BSD difere da licença GNU Publica?

O Linux está disponível sob a Licença Pública Geral GNU (GPL), que é projetada para eliminar o software de código fechado. Em particular, qualquer trabalho derivado de um produto lançado sob a GPL também deve ser fornecido com o código fonte, se solicitado. Por outro lado, a licença BSD é menos restritiva: é permitida a distribuição somente dos binários. O que é particularmente atraente para aplicativos embarcados.

4.6. O que mais eu deveria saber?

Uma vez que há menos aplicativos disponíveis para o BSD em comparação ao Linux, os desenvolvedores do BSD criaram um pacote de compatibilidade com o Linux, que permite que programas do Linux sejam executados no BSD. Esse pacote inclui modificações no kernel para realizar corretamente chamadas de sistema do Linux, além de arquivos de compatibilidade do Linux, como a biblioteca C. Não há diferença perceptível na velocidade de execução entre um aplicativo do Linux rodando em uma máquina Linux e um aplicativo do Linux rodando em uma máquina BSD de mesma velocidade.

A natureza do BSD de ser um sistema em que tudo é provido por "um único fornecedor" significa que as atualizações são muito mais fáceis de se lidar do que frequentemente ocorre no caso no Linux. O BSD lida com as atualizações das versões das bibliotecas fornecendo módulos de

compatibilidade para as versões anteriores, portanto, é possível executar binários bastante antigos sem problemas.

4.7. Qual devo usar, BSD ou Linux?

O que tudo isso significa na prática? Quem deve usar o BSD, quem deve usar o Linux?

Esta é uma pergunta muito difícil de responder. Aqui estão algumas diretrizes:

- "Se não está quebrado, não conserte": Se você já usa um sistema operacional de código aberto e está feliz com ele, provavelmente não existe nenhuma razão para mudar.
- Os sistemas BSD, em particular o FreeBSD, podem ter um desempenho notavelmente superior ao Linux. Mas isto não é uma verdade absoluta. Em muitos casos, há pouca ou nenhuma diferença no desempenho. E em alguns casos, o Linux pode ter um desempenho melhor que o FreeBSD.
- Em geral, os sistemas BSD têm a reputação de oferecer uma melhor confiabilidade, principalmente como resultado de ter uma base de código mais madura.
- Os projetos BSD têm uma reputação melhor pela qualidade e completude da sua documentação. Os vários projetos de documentação visam fornecer uma documentação que é atualizada constantemente, disponibilizada em muitos idiomas, e que cobre todos os aspectos do sistema.
- A licença BSD pode ser mais atraente que a GPL.
- O BSD pode executar a maioria dos binários do Linux, já o Linux por sua vez não pode executar binários do BSD. Muitas implementações do BSD também podem executar binários de outros sistemas semelhantes ao UNIX®. Como resultado, pode ser mais fácil migrar de outros sistemas para o BSD do que seria migrar para o Linux.

4.8. Quem fornece suporte, serviços e treinamento para o BSD?

iXsystems, Inc. fornece contratos de suporte para FreeBSD.

Além disso, o website de cada um dos projetos possui uma lista de consultores disponíveis para contratação: FreeBSD, NetBSD, e OpenBSD.