BSD에 대한 설명

초록

오픈 소스의 세계에서, "리눅스"라는 단어는 "운영체제"라는 단어와 거의 동의어입니다. 하지만 리눅스만이 유일한 오픈 소스 UNIX® 운영체제인 것은 아닙니다. Internet Operating System Counter에 따르면, 1999년 4월을 기준으로 네트워크에 연결된 전세계의 컴퓨터들 중 31.3%가 리눅스를 구동시켰습니다. 14.6%는 BSD UNIX®를 구동시켰습니다. Yahoo!와 같은 세계 최대 규모의 웹 관리 중 일부는 BSD를 통해 이루어집니다. 1999년을 기준으로 세계에서 가장 활발한 FTP 서버(지금은 중단되었습니다)였던 ftp.cdrom.com는 하루 1.4 TB의 데이터를 전송하기 위해 BSD를 사용했었습니다. 이것이 틈새 시장이 아닌 것은 분명합니다: BSD는 잘 유지되어 온 비밀입니다.

무엇이 비밀이라는 뜻일까요? 왜 BSD는 잘 알려져 있지 않을까요? 이 보고서는 이러한 의문과 다른 몇몇의 질문에 대한 것입니다.

이 보고서 전반에 걸쳐, BSD와 리눅스의 차이점은 이렇게 표기될 것입니다.

차례

1. BSD는 무엇일까요?. 1
2. 진짜 UNIX®라고요?. 2
3. 왜 BSD는 잘 알려지지 않았을까요?. 2
4. BSD와 리눅스의 대조. 3

1. BSD는 무엇일까요?

BSD는 "Berkeley Software Distribution"을 나타냅니다. 이는 AT&T의 Research UNIX® 운영체제의 확장판이었던, University of California, Berkeley의 소스 코드 배포판의 이름입니다. 몇몇 오픈 소스 운영체제들은 4.4BSD-Lite라고 알려진 이 소스 코드 릴리즈에 기반을 두고 있습니다. 게다가, 그 운영체제들은 GNU 프로젝트를 비롯한 다른 오픈 소스 프로젝트들로부터 가져온 많은 패키지들도 포함합니다. 전체적인 운영체제는 다음으로 구성되어 있습니다:

• 프로세스 스케줄링, 메모리 관리, 대칭형 다중 처리 (SMP), 장치 드라이버 등을 제어하는 BSD 커널.

리눅스 커널과는 다르게, 서로 다른 기능을 가진 몇몇의 BSD 커널들이 존재합니다.

• 시스템의 기반 API인 C 라이브러리.

BSD C 라이브러리는 GNU 프로젝트가 아니라 Berkeley의 코드를 기반으로 합니다.

• 쉘, 파일 유틸리티, 컴파일러, 링커와 같은 유틸리티들.

이들 중 일부는 GNU 프로젝트로부터 가져왔고, 다른 일부는 그렇지 않습니다.

• 그래픽 디스플레이를 제어하는 X Window 시스템.

대다수의 BSD에서 사용하는 X Window 시스템은 X.Org project에 의해 관리됩니다. FreeBSD는 사용자가 GNOME, KDE, 또는 Xfce와 같은 데스크탑 환경 및 Openbox, Fluxbox, 또는 Awesome과 같은 가벼운 window manager를 선택해서 사용하는 것을 허용합니다.

• 여러 다른 프로그램과 유틸리티들.

2. 진짜 UNIX®라고요?

BSD 운영체제는 클론이 아니라, AT&T의 Research UNIX® 운영체제(이는 현대의 UNIX® System V의 조상이기도 합니다)의 오픈 소스 파생본입니다. 여러분은 이 사실에 놀랄지도 모릅니다. AT&T는 그 코드를 오픈 소스로 공개한 적이 없는데 어떻게 그런 일이 일어날 수 있었을까요?

AT&T UNIX®가 오픈 소스가 아니라는 것과, 상표권의 관점에서 BSD가 엄밀히는 UNIX®가 아니라는 것은 사실입니다. 하지만 한편으로는, AT&T는 캘리포니아에 있는 University of California, Berkeley에 속한 Computer Sciences Research Group (CSRG) 등의 다른 프로젝트로부터 소스를 가져와 사용했습니다. 1976년부터, CSRG는 Berkeley Software Distribution 또는 BSD라고 불린 그들의 소프트웨어 테이프를 배포하기 시작했습니다.

최초의 BSD 릴리즈는 대체로 사용자 프로그램들로 구성되어 있었지만, 그 구성은 Defense Advanced Research Projects Agency(DARPA)가 그들의 네트워크인 ARPANET에서 사용하는 통신 프로토콜을 개선하기 위해 CSRG와 계약을 맺은 이후 극적으로 변화했습니다. 새로운 프로토콜들은 Internet Protocols라고 알려졌으며, 가장 중요한 프로토콜들인 TCP/IP의 구성 요소가 되었습니다. 그 프로토콜의 구현이 최초로 널리 배포된 것은 1982년에 4.2BSD에 포함된 것이었습니다.

1980년대에, 많은 수의 새로운 워크스테이션 회사들이 갑자기 나타났습니다. 대다수는 그들만의 운영체제를 개발하는 것보다 UNIX®를 라이선스받는 것을 선호했습니다. 특히 Sun Microsystems는 UNIX®를 라이선스받아서 SunOS™라고 불리는, 4.2BSD의 한 버전을 만들었습니다. AT&T가 UNIX®를 상업적으로 판매하도록 허가받았을 때, 그들은 System III라고 불리는 구현을 바닥부터 시작했고, 이는 곧 System V로 발전했습니다. System V 코드는 네트워킹을 포함하지 않았기 때문에, 모든 구현은 TCP/IP 소프트웨어 및 csh 쉘이나 vi 에디터와 같은 추가적인 BSD 소프트웨어들을 포함했습니다. 이러한 기능 향상은 일괄적으로 Berkeley Extensions로 알려졌습니다.

BSD 테이프는 AT&T 소스 코드를 포함했기 때문에 UNIX® 소스 라이선스가 필요했습니다. 1990년에, CSRG의 재정은 바닥나고 있었고, 폐쇄를 앞두고 있었습니다. 모임의 일부 회원들은 오픈 소스였던 BSD 코드를 AT&T의 상업 코드 없이 릴리즈하기로 결정했습니다. 이는 마침내 Net/2라고 알려진 Networking Tape 2가 되었습니다. Net/2는 완전한 운영체제가 아니었습니다: 커널 코드의 약 20% 정도가 부족했습니다. CSRG 회원의 한 명이었던 William F. Jolitz는 부족했던 코드를 작성했고 1992년 초에 386BSD라는 이름으로 배포했습니다. 같은 시기에, 전 CSRG 회원들로 구성된 모임이 Berkeley Software Design Inc.라는 상업 회사를 조직했고, 386BSD와 같은 코드 기반을 가지는 운영체제 BSD/386의 베타 버전을 릴리즈했습니다. 이 운영체제의 이름은 나중에 BSD/OS로 바뀌었습니다.

386BSD는 안정적인 운영체제가 되지는 못했습니다. 그 대신, 1993년에 그로부터 두 개의 다른 프로젝트가 파생되었습니다: NetBSD와 FreeBSD가 그것입니다. 두 프로젝트는 386BSD의 개선을 기다리는 인내심의 차이에서 갈라졌습니다: NetBSD의 사람들은 새해 일찍 작업에 착수했고, FreeBSD의 첫 버전은 연말까지 준비되지 않았습니다. 한편, 둘의 코드베이스는 다시 합쳐지기 어려울 정도로 크게 갈라졌습니다. 게다가, 각 프로젝트는 서로 다른 목표를 가지고 있었습니다(이에 대해서는 나중에 다룰 것입니다). 1996년에, NetBSD로부터 OpenBSD가 갈라져 나왔고, 2003년에는 FreeBSD로부터 DragonFlyBSD가 갈라져 나왔습니다.

3. 왜 BSD는 잘 알려지지 않았을까요?

여러 이유들로 인해, BSD는 상대적으로 덜 알려져 있습니다:

1. BSD 개발자들은 대개 홍보하는 일보다 코드를 개선하는 일에 더 관심이 많습니다.

2. 리눅스 인기의 많은 부분은 언론 또는 리눅스 서비스를 제공하는 회사 등 리눅스 프로젝트 외부의 요인에 의한 것입니다. 최근까지, 오픈 소스 BSD는 그러한 후원자들이 없었습니다.

3. BSD 개발자들은 리눅스 개발자들보다 더 숙련된 경향이 있고, 시스템을 사용하기 쉽게 만드는 데에는 상대적으로 덜 관심있어합니다. 초보자들은 리눅스를 좀더 편안해하는 경향이 있습니다.

4. 1992년에, AT&T는 BSD/386의 공급자인 BSDI를 고소했는데, 해당 제품이 AT&T가 저작권을 가지고 있는 코드를 포함하고 있다는 것이 그 이유였습니다. 1994년에 그 소송은 타협을 보기로 합의하는 것으로 일단락되었지만, 그 소송의 유령은 사람들을 계속 괴롭혔습니다. 2000년 3월에 들어 웹에 게시된 글은 해당 소송 사건이 "최근에 해결되었다"고 주장했습니다.

소송에서 분명해진 한 가지 요소는 바로 이름입니다: 1980년대에, BSD는 "BSD UNIX®"라고 알려져 있었습니다. BSD에서 AT&T 코드의 마지막 흔적까지 제거했기 때문에, 그것은 UNIX®라는 이름에 대한 권리 역시 잃었습니다. 그렇기 때문에 여러분은 책 제목 등에서 "4.3BSD UNIX® 운영체제" 그리고 "4.4BSD 운영체제"와 같은 표현을 찾아볼 수 있을 것입니다.

4. BSD와 리눅스의 대조

그래서, 예를 들어, Debian 리눅스와 FreeBSD의 실제적인 차이점은 무엇일까요? 일반적인 사용자들에게, 차이점은 놀랄 만큼 적습니다: 둘 모두 UNIX® 계열 운영체제입니다. 둘 모두 비영리 프로젝트에 의해 개발됩니다(이는 물론 다른 많은 리눅스 배포판에는 적용되지 않는 사실입니다). 아래 항목에서, 우리는 BSD를 살펴 보고 리눅스와 대조해볼 것입니다. 설명은 BSD 설치본의 약 80%를 차지하는 FreeBSD를 기준으로 작성되었지만, NetBSD, OpenBSD 및 DragonFlyBSD에도 차이는 거의 없습니다.

4.1. 누가 BSD를 소유하고 있나요?

BSD는 개인 또는 하나의 회사에 의해 소유되지 않습니다. BSD는 전문적인 기술을 가지고 있는 전세계의 헌신적인 기여자들로 이루어진 커뮤니티에 의해 개발되고 배포됩니다. BSD의 일부 구성 요소는 그들 고유의 오픈 소스 프로젝트에 속해 있고 다른 프로젝트 관리자들에 의해 개발됩니다.

4.2. BSD는 어떻게 개발되고 업데이트되나요?

BSD 커널은 다음의 오픈 소스 개발 모델에 따라 개발되고 업데이트됩니다. 각 프로젝트는 Subversion (SVN)을 통한 공개 소스 트리를 유지합니다. 여기에는 문서 및 다른 부가적인 파일들을 포함한 프로젝트의 모든 소스 파일들이 포함되어 있습니다. SVN은 사용자들이 시스템의 원하는 버전을 "check out" (다시 말하면, 복사본을 추출)하는 것을 허용합니다.

전세계의 많은 개발자들이 BSD의 개선에 기여하고 있습니다. 이들은 세 가지 분류로 나뉘어질 수 있습니다:

• 공헌자들은 코드 또는 문서를 작성합니다. 이들은 소스 트리에 직접 커밋(코드를 추가)하는 것은 허용되지 않습니다. 그들의 코드를 시스템에 포함시키기 위해서는, 먼저 커미터 로 알려진, 등록된 개발자에 의해 검토받아야 합니다.

• 커미터들은 소스 트리에 쓰기 권한을 가진 개발자입니다. 커미터가 되기 위해서, 개인은 그들이 활동하고 있는 영역에 대한 능력을 보여 주어야 합니다.

소스 트리에 수정 사항을 커밋하기 전에 허가를 받아야 하는지는 커미터 개인의 재량입니다. 일반적으로, 숙련된 커미터는 명백해 보이는 수정에 대해서는 다른 사람과 의견을 조율하지 않고 커밋할 수 있습니다. 예를 들어, documentation project 커미터는 오타나 문법 오류 등을 수정할 때 검토받을 필요가 없습니다. 한편, 많은 부분을 또는 복잡하게 수정하는 개발자는 커밋하기 전에 그들의 수정을 제출해서 검토받아야 합니다. 극단적인 경우에, Principal Architect와 같은 권한을 가진 코어 팀 구성원이 수정한 내용을 트리에서 제거할 것(backing out이라고 부릅니다)을 요청할 수도 있습니다. 모든 커미터들은 각각의 커밋에 대한 메일을 받게 되므로, 비밀리에 커밋하는 것은 불가능합니다.

• 코어 팀. FreeBSD와 NetBSD는 각각 프로젝트를 관리하는 코어 팀을 가지고 있습니다. 코어 팀은 프로젝트의 진행 도중에 만들어졌고, 그들의 역할은 항상 잘 정의되어 있지는 않습니다. 코어 팀 구성원이 되기 위해서 개발자일 필요는 없지만, 그런 경우가 일반적입니다. 코어 팀에 대한 규칙은 프로젝트마다 다르지만, 일반적으로 그들은 프로젝트의 진행 방향에 대해 다른 이들보다 큰 영향력이 있습니다.

아래 정리된 내용은 리눅스와 여러 방면에서의 차이점을 나타냅니다:

1. 한 사람이 시스템의 구성 요소를 통제하지는 않습니다. 실제로는, Principal Architect가 코드를 back out할 것을 요구할 수 있기 때문에 이 차이점이 과대평가되기도 합니다. 심지어 리눅스 프로젝트에서도 여러 사람들이 수정할 권한이 있습니다.

2. 한편, 중앙 저장소는 하나인데, 이 곳은 전체 운영체제의 시스템 소스를 이전 버전까지 포함하여 모두 찾아볼 수 있는 통합된 장소입니다.

3. BSD 프로젝트는 단지 커널만이 아니라 "운영체제" 전체를 관리합니다. 이 차이점은 아주 약간의 의미만 있습니다: BSD도 리눅스도 애플리케이션 없이는 유용하지 않습니다. BSD에서 사용되는 애플리케이션들은 리눅스에서 사용되는 애플리케이션과 같은 경우가 많습니다.

4. 하나의 공식 SVN 소스 트리를 관리하는 결과로, BSD의 개발은 명확하게 이루어지고 있으며, 릴리즈 넘버 혹은 날짜를 통해 시스템의 어떤 버전에라도 접근할 수 있습니다. SVN은 또한 시스템의 증분 업데이트를 지원합니다: 예를 들어, FreeBSD 저장소는 하루에 약 100회 업데이트됩니다. 이러한 수정 사항의 대부분은 작은 것들입니다.

4.3. BSD 릴리즈

FreeBSD, NetBSD 그리고 OpenBSD는 시스템을 세 가지 다른 "릴리즈"의 형식으로 제공합니다. 리눅스와 같이, 릴리즈들은 1.4.1 또는 3.5와 같은 번호를 부여받습니다. 더불어, 버전 번호는 시스템의 목적을 나타내는 접미사를 가지고 있습니다.

1. 개발자 버전의 시스템은 CURRENT이라고 불립니다. FreeBSD는 FreeBSD 5.0-CURRENT와 같은 형태로 CURRENT에 숫자를 할당합니다.

NetBSD는 약간 다른 작명법을 사용하여 내부 인터페이스에 변화를 나타내는 한 글자의 접미사를 붙입니다. 예를 들어 NetBSD 1.4.3G와 같습니다. OpenBSD는 숫자를 할당하지 않습니다("OpenBSD-current"). 시스템의 새로운 개발은 모두 이 가지로 들어갑니다.

2. 일 년에 두 번에서 네 번 정도의 적당한 간격으로, 각 프로젝트는 CD-ROM 또는 FTP 무료 다운로드 형식으로 제공되는 RELEASE 버전 시스템을 제공합니다. 예를 들어 OpenBSD 2.6-RELEASE 또는 NetBSD 1.4-RELEASE와 같습니다. RELEASE 버전은 일반적인 버전의 시스템으로 최종 사용자를 위한 것입니다. NetBSD는 세 번째 자리수와 함께 patch release들을 제공합니다. 예를 들어 NetBSD 1.4.2와 같습니다.

3. REALEASE 버전에서 버그가 발견되면, 그것들은 고쳐지고, 고쳐진 내용은 SVN 트리에 반영됩니다. FreeBSD의 경우, 그 결과적인 버전은 STABLE 버전이라고 부르고, NetBSD와 OpenBSD의 경우 계속 RELEASE 버전이라고 부릅니다. CURRENT 가지에서 일정 기간 테스트된 이후에 작은 새 기능들이 이 가지에 추가될 수도 있습니다.

대조적으로, 리눅스는 두 개의 코드 트리를 별도로 관리하고 있습니다: stable 버전과 development 버전이 그것들입니다. stable 버전은 2.0, 2.2, 2.4와 같이 짝수의 minor 버전 숫자를 가지고 있습니다. development 버전은 2.1, 2.3, 2.5와 같이 홀수의 minor 버전 숫자를 가지고 있습니다. 각각의 경우에, 이 숫자 다음에는 정확한 릴리즈를 지칭하는 자릿수가 더 따라옵니다. 더불어, 각 공급자는 그들 고유의 사용자 프로그램 및 유틸리티를 추가하기 때문에, 배포판의 이름 역시 중요합니다. 각 배포판 공급자는 배포판에 대한 버전 숫자도 할당하기 때문에, 전체적인 묘사는 "TurboLinux 6.0 with kernel 2.2.14"와 같이 됩니다.

4.4. BSD에는 어떤 종류가 있나요?

수많은 종류의 리눅스 배포판들과는 대조적으로, 오픈 소스 BSD에는 오직 네 가지 주요한 버전만이 있습니다. 각 BSD 프로젝트는 고유한 소스 트리와 커널을 유지합니다. 그러나 실제로는, userland 코드는 리눅스의 경우에서보다 차이가 적습니다.

각 프로젝트의 목표를 분류하기는 어렵습니다: 차이는 매우 주관적입니다. 기본적으로,

• FreeBSD는 높은 성능 그리고 end user가 손쉽게 사용하는 것을 추구하며, 웹 컨텐츠 제공자들이 선호합니다. FreeBSD는 다양한 플랫폼에서 작동될 수 있으며 다른 프로젝트에 비해 현저히 많은 사용자를 가지고 있습니다.

• NetBSD는 이식성을 최우선으로 하여 개발되고 있습니다: "of course it runs NetBSD". NetBSD는 palmtop에서부터 대형 서버에서까지 구동될 수 있으며, 심지어는 NASA의 space mission에도 사용되었습니다. 오래된 비 Intel® 하드웨어를 구동시킬 때 특히 좋은 선택입니다.

• OpenBSD는 보안과 코드 품질을 최우선으로 하여 개발되고 있습니다: OpenBSD는 오픈 소스와 엄격한 코드 검사를 통해 명백히 correct한 시스템을 만듭니다. 이를 통해 은행, 증권 거래소 및 US 정부 기관 등과 같이 보안이 중시되는 곳에서 사용될 수 있도록 합니다. NetBSD와 마찬가지로, 이는 다양한 플랫폼에서 구동될 수 있습니다.

• DragonFlyBSD는 single-node UP 시스템에서부터 대규모 클러스터 시스템에 이르기까지의 환경에서 높은 성능과 확장성을 추구합니다. DragonFlyBSD는 몇몇의 장기적인 기술 목표들을 가지고 있지만, 핵심은 이해하고 유지하고 개발하기 쉬운 대칭형 다중 처리(SMP) 기반을 제공하는 것에 있습니다.

이와 더불어 오픈 소스가 아닌 두 개의 BSD UNIX® 운영체제들도 있습니다. BSD/OS와 Apple의 Mac OS® X가 그것들입니다:

• BSD/OS는 가장 오래된 4.4BSD 파생본입니다. 이것의 소스 코드는 상대적으로 낮은 가격에 얻을 수 있었지만, 오픈 소스는 아니었습니다. 이것은 FreeBSD와 여러 방면에서 유사합니다. BSDi가 Wind River Systems에 의해 인수된 지 2년 후, BSD/OS는 독립적인 제품으로 살아남는 데에는 실패했습니다. 소스 코드 및 지원은 Wind River에서 여전히 제공했지만, 모든 새로운 개발은 VxWorks 임베디드 운영체제에 집중되었습니다.

• Mac OS® X는 Apple®의 Mac®을 위한 운영체제의 최신 버전입니다. 이 운영체제의 BSD 코어인 Darwin은 x86 및 PPC 컴퓨터를 위한 완전한 기능의 오픈 소스 운영체제로 사용이 가능합니다. 그러나 Aqua/Quartz 그래픽 시스템 및 Mac OS® X의 다른 많은 상용 부분은 클로즈드 소스로 남아 있습니다. 몇몇 Darwin 개발자들은 FreeBSD의 committer이기도 합니다.

4.5. BSD 라이선스는 GNU Public license와 어떻게 다른가요?

리눅스는 closed 소스 소프트웨어를 제거하도록 디자인된 GNU General Public License(GPL)에 따라 배포됩니다. 특히, GPL의 파생물은 모두 사용자가 소스 코드를 요청하면 이를 제공해야 합니다. 이와는 대조적으로, BSD 라이선스는 제약 사항이 적습니다: 바이너리만 배포하는 것이 허용됩니다. 이는 임베디드 애플리케이션을 개발할 때 특히 매력적인 선택입니다.

4.6. 알아야 할 다른 무언가가 있나요?

BSD에서 사용 가능한 애플리케이션의 수가 리눅스의 경우보다 적기 때문에, BSD 개발자들은 리눅스 프로그램을 BSD에서 구동할 수 있도록 해 주는 리눅스 호환성 패키지를 만들었습니다. 이 패키지는 리눅스 시스템 콜을 올바르게 수행하기 위한 커널 수정과, C 라이브러리와 같은 리눅스 호환성 파일들을 포함하고 있습니다. 같은 장비에서 리눅스를 설치해서 리눅스 애플리케이션을 구동할 때와 BSD를 설치해서 리눅스 애플리케이션을 구동할 때 실행 속도는 눈에 띄지 않을 정도로 차이가 작습니다.

BSD의 "all from one supplier" 환경은 리눅스에서 종종 문제가 되기도 하는 업그레이드를 훨씬 쉽게 할 수 있다는 것을 의미합니다. BSD는 라이브러리 버전을 업그레이드할 때 이전 라이브러리 버전에 대한 호환성 모듈을 제공하기 때문에, 몇년 전의 바이너리를 문제없이 실행하는 것이 가능합니다.

4.7. BSD와 리눅스 중에 무엇을 사용해야 하나요?

실제로 이건 무슨 뜻일까요? 누가 BSD를 사용해야 하고, 누가 리눅스를 사용해야 할까요?

이는 대답하기 대단히 어려운 질문입니다. 여기에 약간의 조언이 있습니다:

• "망가지지 않았다면, 고치지 마라": 만일 여러분이 오픈 소스 운영체제를 사용하고 있고, 이에 대해 만족하고 있다면, 아마 굳이 그걸 바꾸어야 할 이유는 없을 겁니다.
• BSD 시스템, 특히 FreeBSD는 리눅스보다 월등히 높은 성능을 발휘할 수 있습니다. 하지만 항상 그런 것은 아닙니다. 대개의 경우에, 성능 차이는 없거나 아주 작습니다. 일부 경우에는, 리눅스가 FreeBSD보다 더 나은 성능을 발휘할 수도 있습니다.
• 일반적으로, BSD 시스템은 보다 성숙한 코드 베이스를 가지고 있기 때문에, 그 안정성이 더 잘 알려져 있습니다.
• BSD 프로젝트는 문서의 품질과 완성도로 잘 알려져 있습니다. 다양한 문서 프로젝트들은 시스템의 모든 방면을 다루는 문서를 다양한 언어로 꾸준히 갱신하는 것을 목표로 하고 있습니다.
• BSD 라이선스는 GPL보다 매력적인 선택일 수 있습니다.
• 리눅스는 BSD 바이너리를 실행할 수 없는 반면, BSD는 대부분의 리눅스 바이너리를 실행할 수 있습니다. 많은 BSD 구현들은 다른 UNIX® 계열 시스템의 바이너리 또한 실행할 수 있습니다. 결과적으로, 다른 시스템에서 BSD로 마이그레이션하는 것은 리눅스로 하는 것보다 수월할 수 있습니다.

4.8. 누가 BSD에 대한 지원, 서비스, 그리고 교육을 제공하나요?

BSDi / FreeBSD Mall, Inc.는 거의 10년 동안 FreeBSD에 대한 지원 계약을 제공해 오고 있습니다.

더불어, 각 프로젝트는 고용에 대한 상담의 목록을 가지고 있습니다: FreeBSD, NetBSD, 그리고 OpenBSD.