

Article 13861 (3 more) in comp.unix.bsd:
From: Keith Bostic
Newsgroups: comp.unix.bsd, comp.bugs.4bsd
Subject: 4.4BSD Release
Date: 29 Jun 1993 21:57:39 GMT
Organization: University of California, Berkeley
Lines: 529
NNTP-Posting-Host: toe.cs.berkeley.edu

June 1, 1993

Dear Colleague:

We are happy to send you information about our June 1993 release of 4.4BSD. This distribution is the final release that will be done by the Computer Systems Research Group (CSRG). For details on the demise of CSRG, see "The End of BSD from Berkeley" below.

This distribution is intended to be used on production systems; it has been run extensively at several test sites and has proven to be stable and reliable. However, because of the shutdown of the CSRG, there will not be anyone available at Berkeley to assist with problems, so it should not be used by sites without enough local expertise to find and fix any problems that are encountered.

The code in this distribution may be redistributed and used in released products provided that you abide by the due credit requirements listed in your license agreement. We have attempted to make the system as compliant with the POSIX 1003.1 and 1003.2 standards as was possible at the time of its release. We have not been able to run it through any of the verification test suites, thus, you should not claim conformance with either standard without first validating the code.

We had planned on releasing two versions of the software, 4.4BSD-Encumbered and 4.4BSD-Lite. Currently, we are releasing only 4.4BSD-Encumbered. The 4.4BSD-Encumbered distribution is available only to sites with UNIX/32V, System III, or System V source licenses with Western Electric, American Telephone and Telegraph (AT&T), or UNIX Systems Laboratories (USL). The 4.4BSD-Encumbered distribution is a complete distribution in the style of 4.3BSD and contains the complete source for the Berkeley Distribution.

The 4.4BSD-Lite distribution was to have been a distribution that was copyrighted by the University of California and others, but could be freely redistributed. It was to have been available to anyone and require no previous license, either from USL/Novell or The Regents of the University of California. Its license agreement and content would have been similar to that of the two BSD Networking Releases. However, USL has brought a lawsuit against the University and the University has voluntarily decided to withhold the distribution of 4.4BSD-Lite until the lawsuit is resolved.

The enclosed information is designed to serve two purposes. The first purpose is to acquaint you with the details of our distribution so you can decide whether you would like to receive it. The second purpose is to tell you how to obtain our distribution.

What is 4.4BSD?

This software distribution is provided on one 6250bpi 1/2" 9-track tape or one 8mm Exabyte cassette only. The 4.4BSD-Encumbered distribution contains complete source as well as binaries for one of the following three architectures:

- + HP 9000/300 68000-based workstations.
- + DECstation 3100 and 5000 MIPS-based workstations.
- + Sparcstation I & II SPARC-based workstations. Please note that the SPARC kernel will not run on the Sparcstation 10.

If you wish to obtain binaries for more than one architecture, they may be purchased at the same time for an additional \$500.00 fee.

The distribution supports a somewhat wider set of machines than those for which we have built binaries. The architectures that are supported in source form include:

- + HP 9000/300 68000-based workstations
- + Intel 386/486-based machines (ISA/AT or EISA bus only)
- + Sony News MIPS-based workstations
- + Omron Luna 68000-based workstations
- + DECstation 3100 and 5000 MIPS-based workstations
- + Sparcstation I & II SPARC-based workstations

The distribution does not include the machine support for the Tahoe and VAX architectures found in previous BSD distributions. Our primary development environment is the HP9000/300 series machines. The other architectures are developed and supported by people outside the university. Consequently, we are not able to directly test or maintain these other architectures, so cannot comment on their robustness, reliability, or completeness.

The major new facilities available in the 4.4BSD release are a new virtual memory system, the addition of ISO/OSI networking support, a new virtual filesystem interface supporting filesystem stacking, a freely redistributable implementation of NFS, a log-structured filesystem, enhancement of the local filesystems to support files and filesystems that are up to 2⁶³ bytes in size, enhanced security and system management support, and the conversion to and addition of the IEEE Std1003.1 ('POSIX') facilities and many of the IEEE Std1003.2 facilities. In addition, many new utilities and additions to the C library are present as well. The kernel sources have been reorganized to collect all machine-dependent files for each architecture under one directory, and most of the machine-independent code is now free of code conditional on specific machines. The user structure and process structure have been reorganized to eliminate the statically-mapped user structure and to make most of the process resources shareable by multiple processes. The system and include files have been converted to be compatible with ANSI C, including function prototypes for most of the exported functions. There are numerous

other changes throughout the system.

The new virtual memory implementation is derived from the MACH operating system developed at Carnegie-Mellon, and was ported to the BSD kernel at the University of Utah. The MACH virtual memory system call interface has been replaced with the ``mmap''-based interface described in the ``Berkeley Software Architecture Manual'' (see UNIX Programmer's Manual, Supplementary Documents, PSD:5). The interface is similar to the interfaces shipped by several commercial vendors such as Sun, USL, and Convex Computer Corp. The integration of the new virtual memory is functionally complete, but still has serious performance problems under heavy memory load. The internal kernel interfaces have not yet been completed and the memory pool and buffer cache have not been merged.

The ISO/OSI Networking consists of a kernel implementation of transport class 4 (TP-4), connectionless networking protocol (CLNP), and 802.3-based link-level support (hardware-compatible with Ethernet*). We also include support for ISO Connection-Oriented Network Service, X.25, TP-0. The session and presentation layers are provided outside the kernel by the ISO development environment (ISODE). Included in this development environment are file transfer and management (FTAM), virtual terminals (VT), a directory services implementation (X.500), and miscellaneous other utilities.

A new virtual filesystem interface has been added to the kernel to support multiple filesystems. In comparison with other interfaces, the Berkeley interface has been structured for more efficient support of filesystems that maintain state (such as the local filesystem). The interface has been extended with support for stackable filesystems done at UCLA. These extensions allow for filesystems to be layered on top of each other and allow new vnode operations to be added without requiring changes to existing filesystem implementations.

In addition to the local ``fast filesystem'', we have added an implementation of the network filesystem (NFS) that fully interoperates with the NFS shipped by Sun and its licensees. Because our NFS implementation was implemented using only the publicly available NFS specification, it does not require a license from Sun to use in source or binary form. By default it runs over UDP to be compatible with Sun's implementation. However, it can be configured on a per-mount basis to run over TCP. Using TCP allows it to be used quickly and efficiently through gateways and over long-haul networks. Using an extended protocol, it supports Leases to allow a limited callback mechanism that greatly reduces the network traffic necessary to maintain cache consistency between the server and its clients.

A new log-structured filesystem has been added that provides near disk-speed output and fast crash recovery. It is still experimental in the 4.4BSD release, so we do not recommend it for production use. We have also added a memory-based filesystem that runs in pageable memory, allowing large temporary filesystems without requiring dedicated physical memory.

The local ``fast filesystem'' has been enhanced to do clustering which allows large pieces of files to be allocated contiguously resulting in near doubling of filesystem

throughput. The filesystem interface has been extended to allow files and filesystems to grow to 2^{63} bytes in size. The quota system has been rewritten to support both user and group quotas (simultaneously if desired). Quota expiration is based on time rather than the previous metric of number of logins over quota. This change makes quotas more useful on file servers onto which users seldom login.

The system security has been greatly enhanced by the addition of additional file flags that permit a file to be marked as immutable or append only. Once set, these flags can only be cleared by the super-user when the system is running single user. To protect against indiscriminate reading or writing of kernel memory, all writing and most reading of kernel data structures must be done using a new `sysctl` interface. The information to be accessed is described through an extensible `Management Information Base` (MIB).

The 4.4BSD distribution contains most of the interfaces specified in the IEEE Std1003.1 system interface standard. The biggest area of change is a new terminal driver. The terminal driver is similar to the System V terminal driver with the addition of the necessary extensions to get the functionality previously available in the 4.3BSD terminal driver. 4.4BSD also adds the IEEE Std1003.1 job control interface, which is similar to the 4.3BSD job control interface, but adds a security model that was missing in the 4.3BSD job control implementation. Other additions include IEEE Std1003.1 signals, FIFOs, byte-range file locking, and saved user and group identifiers.

There are several new tools and utilities included in this release. A new version of `make` allows much-simplified makefiles for the system software and allows compilation for multiple architectures from the same source tree (which may be mounted read-only). Notable additions to the libraries include functions to traverse a filesystem hierarchy, database interfaces to `btree` and hashing functions, a new, fast implementation of `stdio` and a radix sort function. The additions to the utility suite include greatly enhanced versions of programs that display system status information, implementations of various traditional tools described in the IEEE Std1003.2 standard, and many others.

We have been tracking the IEEE Std1003.2 shell and utility work and have included prototypes of many of the proposed utilities. Because most of the traditional utilities have been replaced with implementations conformant to the POSIX standards, you should realize that the utility software may not be as stable, reliable or well documented as in traditional Berkeley releases. In particular, almost the entire manual suite has been rewritten to reflect the POSIX defined interfaces, and in some instances it does not correctly reflect the current state of the software. It is also worth noting that, in rewriting this software, we have generally been rewarded with significant performance improvements. Most of the libraries and header files have been converted to be compliant with ANSI C. The default compiler (`gcc`) is a superset of ANSI C, but supports traditional C as a command-line option. The system libraries and utilities all compile with either ANSI or traditional C.

Work has also progressed in several other areas. Several important enhancements have been added to the TCP/IP

protocols including TCP header prediction and serial line IP (SLIP) with header compression. The routing implementation has been completely rewritten to use a hierarchical routing tree with a mask per route to support the arbitrary levels of routing found in the ISO protocols. The routing table also stores and caches route characteristics to speed the adaptation of the throughput and congestion avoidance algorithms.

The Kerberos (version 4) authentication software has been integrated into much of the system (including NFS) to provide the first real network authentication on BSD.

This release includes several important structural kernel changes. The kernel uses a new internal system call convention; the use of global ('`u-dot'') variables for parameters and error returns has been eliminated, and interrupted system calls no longer abort using non-local goto's (longjmp's). A new sleep interface separates signal handling from scheduling priority, returning characteristic errors to abort or restart the current system call. This sleep call also passes a string describing the process state, which is used by the ps(1) program. The old sleep interface can be used only for non-interruptible sleeps. The sleep interface (tsleep) can be used at any priority, but is only interruptible if the PCATCH flag is set. When interrupted, tsleep returns EINTR or ERESTART.

Many data structures that were previously statically allocated are now allocated dynamically. These structures include mount entries, file entries, user open file descriptors, the process entries, the vnode table, the name cache, and the quota structures.

The End of BSD from Berkeley

For the following three reasons, the CSRG clearly could not continue in its present form.

Funding had become increasingly time-consuming and difficult. We were spending more and more of our time obtaining funding, time that we would have preferred to spend working on BSD. As many of you are intimately aware, computer corporations are actively seeking ways to reduce discretionary outlays. Also, as UNIX vendors have developed their own research groups, the work of the CSRG became less necessary to them. Finally, making BSD freely redistributable resulted in fewer distributions sold, as other organizations sold our releases for less money.

Support within the University of California declined as BSD became less widely used internally. Victims of our own success, many of the features once found only in BSD are now available from every vendor.

The system has become too large and complex for a group of four to architect and maintain. In the last few years it became obvious to us that we had to expand the size of our group if we wanted to continue developing and distributing a complete UNIX system. Expansion was impossible given the external funding environment and the space constraints imposed by the university.

BSD has always been a community effort, and, as a community effort, does not rely on a small group of people in Berkeley to keep it going. BSD will not go away, but will

live on through the free software and commercial efforts of many people. We thank you for your support over the years, your funding, and, of course, the software you've contributed to make the BSD system what it is today!

How to obtain 4.4BSD-Encumbered

To obtain 4.4BSD-Encumbered we require execution of the Berkeley License Agreement (6/92). In addition, foreign licensees must execute Addendum Number One for Foreign Licensees in ordering 4.4BSD-Encumbered. The fee is \$2500.00 for 4.4BSD-Encumbered.

Because we are a research and development organization and not a commercial organization, we make our research results available for a small license fee. We distribute only the whole system "As Is" and cannot send individual pieces of the system. We are required by the University of California to have a formal license arrangement with each organization to which we distribute. In addition, for 4.4BSD-Encumbered, we are required to secure a copy of the USL (or AT&T or Western Electric) Software Agreement with your organization and confirm it with USL before the software can be shipped.

Specifically, for 4.4BSD-Encumbered, we must receive from your organization the following material before the distribution can be sent:

- + Two copies of the current Software Agreement between your company or institution and USL (or AT&T or Western Electric) that authorize you as a source licensee for UNIX/32V, System III, or System V. Note that a complete copy of the agreement up to the Schedule is required, not just the cover and/or signature page. Letters authorizing additional CPUs are not necessary in this process; however, it is your legal responsibility to obtain an additional CPU authorization from USL.
- + Two original signed and executed copies of the Berkeley License Agreement (6/92) between your company or institution and The Regents of the University of California along with Exhibit A properly filled out. For Foreign licensees, there is an Addendum to the License Agreement that must also be executed. The name of the organization on the Berkeley License Agreement must be the same as that which appears on the Software Agreement with USL (or AT&T or Western Electric). The Berkeley License Agreement (6/92) must be signed by a duly authorized person who holds a position that is at the same level or a higher level of authority as that which appears on the USL (or AT&T or Western Electric) Software Agreement. Please have this person's name and title typed in the available space in addition to the signature. This license agreement applies to all the CPUs covered by the Software Agreement with USL (or AT&T or Western Electric) that you have provided. One signed copy of the Berkeley License Agreement will be returned to you after it has been executed by The Regents of the University of California.
- + A check from a U.S. bank for \$2500.00 must be received before the distribution can be sent. Checks should be made payable to "The Regents of the University of California, Computer Systems Research Group." If you must issue a Purchase Order, together with your prepayment,

please issue one that is blank-backed. If this is not possible, insert and initial in the body of the Purchase Order the following clause: "The terms and conditions of this Purchase Order are not accepted by The Regents of the University of California. The revised Berkeley License Agreement (6/92) prevails." Wire transfers are strongly discouraged.

- + The attached Site Information Form completely filled out. Your copy of the signed 4.BSD-Encumbered License Agreement will be sent to the person listed as the administrative contact. The distribution itself will be sent to the technical contact. All information is kept confidential; it is for our use in notifying you of important bug fixes and the availability of 4.BSD-Lite should it become available. Please note that we cannot ship to post office boxes; therefore, please have the technical contact's address supplied without use of a post office box.

A checklist is included to aid you in assembling this material. All the above material must be sent to:

Pauline Schwartz, Distribution Coordinator
Computer Systems Research Group
Computer Science Division, EECS
University of California
Berkeley, California 94720

Once all these items have been received and are in proper order, the distribution will be sent to the technical address listed on the Site Information Form. We cannot provide delivery dates. Once the material is assembled and packaged, the distribution is shipped by commercial carrier. Order of shipment will be based on time of arrival of the properly completed paperwork and confirmation with USL if necessary. Because of the differential in costs of shipping outside the United States, we ask that organizations beyond the North American continent pay the collect shipping charges. If the destination is one where collect shipment cannot be made by the carrier, then advance payment of the shipping charges will be required.

The most expedient way to ensure that your full distribution is sent as quickly as possible is to include in a single package two original copies of the appropriate Berkeley License Agreement completed and properly signed (without change), two complete copies of your USL (or AT&T or Western Electric) Software Agreement, the appropriate check properly made out to "The Regents of the University of California, Computer Systems Research Group" and a completely filled out Site Information Form and to send this single package to the address noted above.

Please note that if you modify the Berkeley License Agreement, you may experience a delay of three months or more before receiving an acceptance or denial of the changes. We reserve the right to cancel your application if we have not received the requested paperwork within 60 days from the date it was sent to us.

Special Cases

University of California Sites. If you are a part of the University of California, the following requirements apply: To run 4.4BSD-Encumbered on any CPU, you must have a

CPU authorization under the Regents of the University of California Software Agreement with USL. This can be obtained by contacting Pam True at (510) 642-6348 in Berkeley Campus Materiel Management for an application. A copy of this should be sent to us. In addition, the following items must be sent to the Computer Systems Research Group: 1) a letter of authorization signed by the Director or Head of Department requesting 4.4BSD-Encumbered, stating that you have read and understood the Berkeley License Agreement (6/92) and that your organization will abide by it; 2) an IOC for \$2,500.00; and 3) a Site Information Form.

Government Agencies and Government Contractors.

- + The U.S. Government has a UNIX Source Software Agreement with AT&T dated Sept. 1, 1975. If you are a government agency operating under the 1975 Software Agreement, you do not need a copy of the aforementioned Software Agreement; instead you must send a copy of your additional CPU authorization from AT&T. The Berkeley License Agreement for 4.4BSD-Encumbered (6/92) should be signed by the appropriate Contracting Officer.
- + Several government agencies have acquired their own AT&T UNIX Software Agreement. Here, we need a copy of this Software Agreement with USL or AT&T. The Berkeley License Agreement (6/92) must be signed by the same officeholder (or replacement) whose signature appears on the Software Agreement with USL or AT&T. The government agency shall be identified as the Licensee in the Berkeley License Agreement (6/92).
- + If you are a contractor of the Government and have obtained an additional CPU authorization from USL or AT&T for your contract work, the Berkeley License Agreement (6/92) must be signed by the appropriate Contracting Officer for the contract. The contractor should address a letter to the Contracting Officer stating that the contractor agrees to abide by the terms and conditions of the Berkeley License Agreement (6/92) for 4.4BSD and ask that the Contracting Officer sign the Berkeley License Agreement (6/92) for 4.4BSD. The Contracting Officer should then return the signed Berkeley License Agreement (6/92) directly to the Computer Systems Research Group with a cover letter stating that the contractor is hereby authorized to receive a copy of 4.4BSD-Encumbered.

A Special Note

The procedures and rules set out in this document are University and USL constraints that must be followed for the distribution of software to be possible. The Computer Systems Research Group has no control over these constraints and must reject your application if material submitted is not in order. If you have questions about the licensing process after reading this letter, you may call Pauline Schwartz at (510) 642-7780, write to her, or contact her via electronic mail at pauline@cs.berkeley.edu.

Sincerely yours,

Marshall Kirk McKusick
Research Computer Scientist

UNIX, UNIX/32V, UNIX System III, and UNIX System V are registered trademarks of USL/Novell in the USA and some other countries.

Ethernet is a trademark of the Xerox Corporation. June 1, 1993

Colleague:

We are happy to send you information about our June release of 4.4BSD. This distribution is the final release that will be done by the Computer Systems Research Group (CSRG). For details on the demise of CSRG, see "The End of BSD from Berkeley" below.

This distribution is intended to be used on production systems. It has been run extensively at several test sites and has proven to be stable and reliable. However, because of the shutdown of the CSRG, there will not be anyone available at Berkeley to assist with problems, so it should not be used by sites without enough local expertise to find and fix any problems that are encountered.

The code in this distribution may be redistributed and used in released products provided that you abide by the due credit requirements listed in your license agreement. We have attempted to make the system as compliant with the POSIX 1003.1 and 1003.2 standards as was possible at the time of its release. We have not been able to run it through any of the verification test suites, thus, you should not claim conformance with either standard without first validating the code.

We had planned on releasing two versions of the software, 4.4BSD-Encumbered and 4.4BSD-Lite. Currently, we are releasing only 4.4BSD-Encumbered. The 4.4BSD-Encumbered distribution is available only to sites with UNIX/32V, System III, or System V source licenses with Western Electric, American Telephone and Telegraph (AT&T), or UNIX Systems Laboratories (USL). The 4.4BSD-Encumbered distribution is a complete distribution in the style of 4.3BSD and contains the complete source for the Berkeley Distribution.

The 4.4BSD-Lite distribution was to have been a distribution that was copyrighted by the University of California and others, but could be freely redistributed. It was to have been available to anyone and require no previous license, either from USL/Novell or The Regents of the University of California. Its license agreement and content would have been similar to that of the two BSD Networking Releases. However, USL has brought a lawsuit against the University and the University has voluntarily decided to withhold the distribution of 4.4BSD-Lite until the lawsuit is resolved.

The enclosed information is designed to serve two purposes. The first purpose is to acquaint you with the details of our distribution so you can decide whether you would like to receive it. The second purpose is to tell you how to obtain our distribution.