

FreeBSD Status Reports Q2/2007

Source: <http://unix.derkeiler.com/Mailing-Lists/FreeBSD/current/2007-07/msg00403.html>

- *From:* Max Laier <max@xxxxxxxxxxxxxxxx>
 - *Date:* Tue, 10 Jul 2007 12:16:26 +0200
-

FreeBSD Quarterly Status Report

Introduction

This report covers FreeBSD related projects between April and June 2007. Again an exciting quarter for FreeBSD. In May we saw one of the biggest developers summits to date at BSDCan , our 25 Google Summer of Code students started working on their projects – progress reports are available below, and finally the 7.0 release cycle was started three weeks ago.

If your are curious about what's new in FreeBSD 7.0 we suggest reading Ivan Voras' excellent summary at: <http://ivoras.sharanet.org/freebsd/freebsd7.html> and of course these reports.

The next gathering of the BSD community will be at EuroBSDCon in Copenhagen , September 14–15. More details about the conference and the developer summit are available in the respective reports below.

Thanks to all the reporters for the excellent work! We hope you enjoy reading.

Google summer of code

- * A GUI audit analyzer for FreeBSD
- * Apple's MacBook on FreeBSD
- * BSD Bintools project
- * Distributed Logging Daemon
- * finstall
- * FreeBSD–update front end
- * Gvinum improvements
- * http support for PXE
- * Linuxulator update
- * lockmgr rewriting
- * mtund – Magic Tunnel Daemon
- * Multicast DNS and Service Discovery

FreeBSD Status Reports Q2/2007

- * Porting Linux KVM to FreeBSD
- * Porting OpenBSD's sysctl Hardware Sensors Framework to FreeBSD
- * Ports Collection infrastructure improvements
- * Security Regression Test
- * tarfs: A tar File System

Projects

- * FreeBSD/xen
- * HDTV Drivers (ATSC)
- * Kernel contention reduction using mysql
- * Stack trace capture in PMCTools
- * TrustedBSD Audit
- * TrustedBSD MAC Framework
- * USB
- * USB update

FreeBSD Team Reports

- * Ports Collection
- * Problem Report Database
- * Release Engineering
- * Security Officer and Security Team
- * The FreeBSD Foundation

Kernel

- * Fine grain thread locking
- * gvinstor
- * SCHED_SMP and SCHED_ULE
- * TrustedBSD priv(9)

Network Infrastructure

- * 10Gigabit Network Support
- * FAST_IPSEC Upgrade
- * FreeBSD and Wake On Lan
- * Multi-link PPP daemon (MPD)
- * Multiprocessor Network Stack
- * Network Stack Virtualization
- * Wireless Networking

Vendor / 3rd Party Software

- * FreeBSD and Coverity Prevent
- * FreeSBIE
- * OpenBSD packet filter – pf
- * PC-BSD

Miscellaneous

- * EuroBSDcon 2007
 - * EuroBSDCon 2007 Developer Summit
 - * libarchive/bsdtr
 - * The Hungarian Documentation Project
-

10Gigabit Network Support

Contact: Kip Macy <kmacy@xxxxxxxxxxxx>
Contact: Andrew Gallatin <gallatin@xxxxxxxxxxxx>
Contact: Jack Vogel <jfv@xxxxxxxxxxxx>
Contact: Robert Watson <rwatson@xxxxxxxxxxxx>

Support was added for two more 10gigabit network drivers and there were major advances on improving system performance over 10g media.

Kip Macy committed a new driver for the Chelsio adapters. The cxgb driver supports all current 10g adapters, as well as the new four-port gigabit model. The cxgb driver work was supported by Chelsio.

Drew Gallatin made significant improvements to the Myricom 10g driver mxge. With these updates the driver does line rate transfers with less system overhead.

Neterion contributed the nxge driver to support all their Xframe 10Gbe Server/Storage adapters. The initial driver import was done by Sam Leffler; a switch over to vendor support will happen soon.

Jack Vogel is preparing a driver to support the latest Intel 10g hardware devices. The new driver – ixgbe – will complement the existing ixgb driver that supports older Intel 10g cards.

Kip and Drew worked with other folks on performance analysis and tuning. This work improved cpu affinity and reduced overhead for managing network resources. Work is also underway to define a common Large Receive Offload (LRO) infrastructure. LRO is analogous to TSO on the receive side enabling drivers to receive at near line rate with normal sized frames. This common code base will help replace driver-specific code.

A GUI audit analyzer for FreeBSD

URL:

Contact: Dongmei Liu <ldm@xxxxxxxxxxxxxxxxxxxx>

This project is due to provide a GUI audit log analysis tool for FreeBSD. Refer to ethereal/wireshark packet parsing engine and its framework to view and parse audit logs.

Open tasks:

1. Get a GUI framework using GTK2.0 include menu bar, toolbar, list view and tree view.
 2. Parse and display audit log in the trailer file in the list view and tree view.
 3. Online capture audit log and parse and display them in the list view and tree view
 4. Add the filter mechanism
 5. Add the statistic mechanism
 6. Remote audit log analysis mechanism
-

Apple's MacBook on FreeBSD

URL:

<http://repoman.freebsd.org/depotTreeBrowser.cgi?FSPC=//depot/projects/soc2007/rpaulo%2dmacbook/>

URL: <http://wiki.freebsd.org/AppleMacbook>

Contact: Rui Paulo <rpaulo@xxxxxxxxxxx>

Apple's MacBook computers are nicely designed and have neat features that other laptops don't. While Mac OS X is a nice operating system, UNIX folks (like me) would prefer to run other operating systems like FreeBSD. This project aims to bring bug fixes and new drivers to FreeBSD that would help running this OS on this platform.

Open tasks:

1. Write drivers or fix issues for/with the touchpad, keyboard, remote control IR reciever, Bluetooth.
 2. Fix reboot, halt, suspend/resume issues.
-

BSD Bintools project

URL: <http://wiki.freebsd.org/BSDBintools>

Contact: Kai Wang <kaiw27@xxxxxxxxxxx>

A basic implementation of ar(1) (include ranlib) was finished and available in the perforce repository. Currently it provides all the main functions an ar(1) should have and it is based on the libarchive and libelf library thus is expected to have a better and simpler structure than the GPL'ed version. The work left in this part of the project is to perform a elaborate test and add additional functions.

Distributed Logging Daemon

FreeBSD Status Reports Q2/2007

URL:

<http://docs.freebsd.org/cgi/getmsg.cgi?fetch=232192+0+/usr/local/www/db/text/2007/freebsd-hackers/20070527.freebsd-hackers>

URL:

<http://perforce.freebsd.org/depotTreeBrowser.cgi?FSPC=//depot/projects/soc2007/karma%5faudit/dlog&HIDEDEL=NO>

Contact: Alexey Mikhailov <karma@xxxxxxxxxxxx>

Contact: Bjoern Zeeb <bz@xxxxxxxxxxxx>

The basic idea behind this project is to implement secure and reliable log file shipping to remote hosts. While the implementation focuses on audit logs, the goal is to build tools that will make it possible to perform distributed logging for any application by using a simple API and linking with a shared library.

Open tasks:

1. Network protocol implementation
2. Spooling
3. SSL support

EuroBSDcon 2007

URL: <http://2007.EuroBSDCon.org/>

Contact: EuroBSDCon 2007 Organizing Committee <info@xxxxxxxxxxxx>

The sixth EuroBSDCon will take place at Symbion in Copenhagen, Denmark on Friday the 14th and Saturday 15th of September 2007.

The programme is ready and online at the webpage. Registration is open. Details about tutorials and Legoland trip are ready too. The keynote will be John Hartman: Real men's pipes

If you share a room with friends at the hostel, then lodging is really inexpensive, and the lounge has high speed Internet access. Staying at the hostel is of course optional, and the area has several hotels.

KD85.com and O'Reilly will each have a booth at the conference.

We are still looking for more sponsors.

A public IRC channel #eurobsdcon on EFnet has been created for discussion and questions about the conference.

EuroBSDCon 2007 Developer Summit

URL: <http://wiki.freebsd.org/200709DevSummit>

FreeBSD Status Reports Q2/2007

Contact: Poul-Henning Kamp <phk@xxxxxxxxxxx>

The next developer summit will be different from the previous ones.

Very different.

Gone are the auditorium style seating, beamers, endless presentations and soggy sandwiches.

Instead we head out to a an old village school in the beautiful Danish countryside, we hang around all over the place, sleep in the old science room, cook our own food and hack the living daylights out of anything we care for.

September 17th and 18th, right after EuroBSDcon2007 in Copenhagen. (Well, right after the optional trip to legoland...)

Be there!

PS: Yes, it's not uncivilized, there is a full speed ADSL and WLAN.

FAST_IPSEC Upgrade

Contact: George Neville-Neil <gnn@xxxxxxxxxxx>

Contact: Bjoern Zeeb <bz@xxxxxxxxxxx>

FAST_IPSEC has now replaced Kame IPsec as the IPsec stack in HEAD. This will be part of the 7.0 release. The merge happened in early July with George handling the kernel bits and Bjoern handling user space.

The kernel option IPSEC is now the ONLY option for IPsec support in the FreeBSD kernel.

Open tasks:

1. Test test test!!!!
-

Fine grain thread locking

Contact: Jeff Roberson <jeff@xxxxxxxxxxx>

Contact: Attilio Rao <attilio@xxxxxxxxxxx>

Contact: Kris Kennaway <kris@xxxxxxxxxxx>

Over the past 6 months several developers undertook an effort to replace the global scheduler lock with a finer-grain interface modeled on the Solaris container lock approach. This significantly reduces contention on higher-end multiprocessor machines.

FreeBSD Status Reports Q2/2007

This patch went into 7.0-CURRENT and has proven to be very stable. The last remaining bugs are in rusage and effect only process time accounting statistics.

finstall

URL: <http://wiki.freebsd.org/finstall>

Contact: Ivan Voras <ivoras@xxxxxxxxxxx>

Project "finstall" aims to create a next-generation FreeBSD installer that will make use of the newest features present in the system. The project should yield something usable for 7.0-RELEASE, but the intention is to keep it as a "second" installer system during 7.x, alongside sysinstall. In any case, sysinstall will be kept for architectures not supported by finstall (e.g. all except i386 and amd64).

Open tasks:

1. The work is progressing well and on plan. There's a small setback currently with X11 applications executing of a read-only file system (at least that's the currently recognizable symptom).
 2. Any interested testers are very much welcome!
-

FreeBSD and Coverity Prevent

Contact: Pawel Jakub Dawidek <pjd@xxxxxxxxxxx>

Contact: David Maxwell <dmaxwell@xxxxxxxxxxx>

FreeBSD's static analysis scans have been updated with a recent version of Coverity Prevent. Coverity is providing additional advice on configuration of the analysis to maximize the benefit from the tools.

At BSDCan2007, Coverity provided FreeBSD with a license for an additional analysis tool called Extend, which allows writing custom FreeBSD specific code checkers. David Maxwell presented training material for interested FreeBSD developers. Some applications of custom checkers have been considered, and more results will be forthcoming as they are implemented and tested.

FreeBSD and Wake On Lan

URL: <http://stsp.name/wol/>

URL: <http://stsp.name/wol/README.txt>

URL: <http://www.freebsd.org/cgi/query-pr.cgi?pr=83807&cat=kern>

Contact: Stefan Sperling <stsp@xxxxxxxx>

FreeBSD Status Reports Q2/2007

I have been working on making wake on lan (WOL) work with FreeBSD. Contrary to popular believe OS support is required for WOL to work properly. In particular network card drivers need to configure network cards for WOL during system shutdown, else the cards won't wake up. WOL is not just a BIOS issue.

This is work in progress. Currently the following cards/chipsets are supported:

- * NatSemi DP83815 (if_sis)
- * Via Rhine (if_vr, only VT6102 and up chips support WOL)
- * Nvidia nForce (if_nve, needs testing)
- * 3Com Etherlink XL and Fast Etherlink XL (if_xl, needs testing , only 3c905B type adapters support WOL)

I would be glad to get more feedback on my patch. I can add support for more chipsets but I need testers for hardware I don't have. I would appreciate access to data sheets for any NIC chipsets that are supported by FreeBSD and have WOL support.

I would especially appreciate technical feedback on the patch, preferably by a committer who is willing to nitpick the patch to make it ready for inclusion in -CURRENT. I currently maintain the patch against RELENG_6_2 for my own use but I would port it to -CURRENT for inclusion.

FreeBSD-update front end

URL: <http://wiki.freebsd.org/FreeBSDUpdateFrontend>

Contact: Andrew Turner <andrew@xxxxxxxxxxxx>

The project is split up with a front end to interact with the user and a back end to interact with freebsd-update. The back and front ends are able to communicate with each other using an XML protocol. The GUI is almost at the point it can take a command from the user and send it to the back end. The back end is able to detect when updates are ready.

FreeBSD/xen

Contact: Rink Springer <rink@xxxxxxxxxxxx>

Work is well under way to finish Kip Macy's FreeBSD/xen port, and get it into a shape which is suitable for inclusion in 7.0.

Generally, the port is stable and performs quite well. The major bottleneck is the inability to work with GCC 4.2, this is the last major TODO before the work can be committed.

Open tasks:

1. Fix the port to correctly work with GCC 4.2.
 2. Port the Xen drivers to newbus.
 3. Test/fix PAE support.
 4. Start on amd64 support.
-

FreeSBIE

URL: <http://www.freesbie.org>

URL: <http://liste.gufi.org/mailman/listinfo/freesbie>

Contact: Matteo Riondato <matteo@xxxxxxxxxxxx>

Contact: FreeSBIE Staff <staff@xxxxxxxxxxxx>

Contact: FreeSBIE ML <freesbie@xxxxxxxx>

After the success of FreeSBIE-2.0.1-RELEASE, development slew down a bit, but we have a big task for the summer: enable unionfs again and trying the new efficient memory filesystem, tmpfs.

For all new ISO images we will be following RELENG_7, with the hope to release a stable image once 7.0-RELEASE have been released.

Open tasks:

1. Build and test an ISO image with FreeSBIE+unionfs+tmpfs.
-

Gvinum improvements

URL: <http://folk.ntnu.no/lulf/patches/freebsd/gvinum/soc2007>

URL: <http://blogs.freebsdish.org/lulf/>

URL: <http://wiki.freebsd.org/UlfLilleengen/SOC>

Contact: Ulf Lilleengen <lulf@xxxxxxxxxxxx>

My previous status reports contained a lot of code that updated gvinum with the old vinum features.

This year gvinum has been significantly rewritten. Lukas Ertl began rewriting the way gvinum is organized from using a multi consumer/provider model, to use a single consumer and provider, and having an event-system that first handles user-requests, and then runs normal I/O operations (Much like other GEOM classes). This makes the code easier to read, and perhaps there will be less bugs :)

1. setstate on plexes and volumes.
2. attach/detach command now works.
3. concat/stripe/mirror commands. The previous code conflicted more than I expected with the new gvinum system, but it should work now.
4. (Mounted) rebuilds possible.

5. (Mounted) sync possible.
6. Some refactoring of old code (Basically updating old code to use the new event system, and add some abstractions where possible)

And of course, some time has gone to work out how things should be done, and to fix other bugs. I hope some of you are interested in trying this out (all the work has been in perforce so far), a patch can be found in the URL section. . This is a bit experimental, and although I've done much testing to hunt down bugs, there are most probably bugs left.

I have other goals this summer as well. However, since some parts of gvinum was rewritten, I might not be able to do all of these, but growing is already working for the concatenated volumes (and also mirrored). I'd also like to implement growing for Raid5 arrays as well. Logging plexes would also be cool to have, but this is not really needed, since we have g_journal. Both these features will be addressed after I've made sure gvinum does all old vinum does, and also perhaps better. As I might have some extra time on my hands this summer, so I gladly accept suggestions on what else I might fix or implement "while I'm at it".

Open tasks:

1. Stability, stability, stability. I want gvinum to work really well. To accomplish that I have several test-machines I'm going to do different tests on. I sort of have a little test-plan in the working that I'll be using.
2. A gvinumadmin tool that would make gvinum easier to use for unexperienced users. Perhaps integrate this into the installer. This is now probably something I'll do at the end, when hopefully everything works :) I might poke Ivan Voras a bit on this.
3. Documenting gvinum and it's differences to vinum better. I take notes on where I need to document, so this is in progress.
4. Implementing growing and shrinking of volumes.
5. Implement logging plexes. Log all parity data being written.

gvirstor

URL: <http://wiki.freebsd.org/gvirstor>

Contact: Ivan Voras <ivoras@xxxxxxxxxxxx>

Gvirstor is a GEOM class which provides virtual storage capacity (something like virtual memory for storage devices). It's ready to be committed to HEAD (the plan is for it to get into 7.0-RELEASE).

Open tasks:

1. Any interested testers are welcome!

HDTV Drivers (ATSC)

URL:

<http://perforce.freebsd.org/fileSearch.cgi?FSPC=%2F%2Fdepot%2Fuser%2Fjimg%2Fbktrau%2F...&ignore=GO%21>

URL:

<http://perforce.freebsd.org/fileSearch.cgi?FSPC=%2F%2Fdepot%2Fuser%2Fjimg%2Fcxd%2F...&ignore=GO%21>

Contact: John-Mark Gurney <jmg@xxxxxxxxxxxx>

This entry was previously the Bt878 Audio Driver (aka FusionHDTV 5 Lite driver) announcement, but as work expanded slightly, it's a bit more generic now.

A few bugs in bktrau has been fixed since January. If you have been running an earlier version, it is recommended to upgrade as the driver could panic. The driver works with multiple cards in the same machine (tested with two).

FusionHDTV 5 Lite --- Due to lack of documentation from DVICO and LG, I have copied magic values from the Linux driver to get ATSC capturing working.

ATI HDTV Wonder --- After years of trying to get into the ATI developer program, they have finally suspended it, so no support from ATI. I have started work on a driver, cxd, for the Conexant CX2388x based cards. The ATI HDTV Wonder uses ATI's own demodulator, and I was able to get it to tune, after cribbing from the Linux driver. When capturing, I get some valid data, but not all the data. Due to lack of support from ATI and linux-dvb the project has been put on indefinite hold.

If someone has another CX2388x based card, it shouldn't be too hard to take the driver and get it working with a different tuner.

A Python module is available for both drivers/cards, along w/ a sample capture application using it. The module is now known to work well with threads so that tuning (expensive due to i2c ioctl's) can happen in another thread without causing program slow down. The module is working well with a custom PVR backend.

Open tasks:

1. Provide support for NTSC and FM tuning.
2. Add support for other cards and tuners that use the Bt878 chip.
3. Add support for other cards and tuners that use the CX2388x chip.

http support for PXE

URL:

http://perforce.freebsd.org/depotTreeBrowser.cgi?FSPC=//depot/projects/soc2007/taleks-pxe_http

URL: http://wiki.freebsd.org/http_support_for_PXE

Contact: Alexey Tarasov <taleks@xxxxxxxxxxxx>

Main goal of project is to introduce code working in PXE preboot environment, able to download from web server via direct connection or http proxy and prepare booting of FreeBSD kernel.

Already implemented, but haven't thoroughly tested: PXE wrappers core code, ARP, ICMP echo request/reply, sockets code similar to common sockets (UDP and TCP modules). On base of sockets: simple DHCP client, DNS client.

Currently working on http client, TCP testing, kernel booting and documenting main concepts of project modules.

Open tasks:

1. Testing PXE API related code in different PXE implementations.
2. Testing of implemented protocols.

Kernel contention reduction using mysql

URL: <http://jeffr-tech.livejournal.com/>

Contact: Jeff Roberson <jeff@xxxxxxxxxxxx>

FreeBSD developers have been using mysql as a testbed to find kernel contention hotspots in the kernel. As a result of this we have seen a 5x performance improvement over 6.0 on 8way machines. Recent changes include finer locking in fcntl(), removing Giant from flock and fcntl F_SETLK. These changes will be available in 7.0 and primarily improve write performance. Experimental changes to select() have also been discussed on arch@ that solve contention issues there however these will not be ready in the 7.0 timeframe.

libarchive/bsdtar

URL: <http://people.freebsd.org/~kientzle/libarchive/>

Contact: Tim Kientzle <kientzle@xxxxxxxxxxxx>

Contact: Colin Percival <cperciva@xxxxxxxxxxxx>

Both libarchive 2 and bsdtar 2 are now in -CURRENT and will be in 7.0. Libarchive 1.9 and bsdtar 1.9 should be in 6-STABLE in time for 6.3.

libarchive 2 is much faster writing to disk than libarchive 1. It also supports new formats, has several minor API/ABI corrections, is more portable, and has many fewer bugs. Of special note is "libarchive_test", a new program that exercises much of the libarchive functionality; anyone interested in working on libarchive should become familiar with this test suite. bsdtar 2 is less ambitious, but does have a number of bug fixes and takes advantage of several new features in libarchive 2.

libarchive 1.9 is identical to libarchive 2 except it maintains the old API/ABI. Similarly, bsdtar 1.9 is nearly identical to bsdtar 2, lacking only a few features that would prevent it from being used with existing libarchive 1 libraries.

Open tasks:

1. Tim Kientzle has started work on a libarchive-based cpio implementation that should be ready for inclusion with FreeBSD 8.
2. Volunteer needed: We want a libarchive-based pax to replace our out-of-date pax implementation.
3. Volunteer needed: pkg_add should use libarchive instead of forking an external tar; this could eventually make it much faster.
4. Volunteer(s) needed: libarchive should write more cpio variants (easy); libarchive should read and writemtree format (not difficult); libarchive should write GNUtar 1.0 format sparse tar entries (tricky); bsdtar should support --metadata=<archive> to read names and properties from one archive, with data from disk, to create a new archive (mtree support in libarchive would make this very useful); bsdtar should preserve sparseness when creating archives.

Linuxulator update

URL: <http://wiki.freebsd.org/linux-soc2007>

Contact: Roman Divacky <rdivacky@xxxxxxxxxxxx>

Contact: Konstantin Belousov <kib@xxxxxxxxxxxx>

Just like last year I got the opportunity to work on updating the Linuxulator to Linux version 2.6. This year I work on finishing futexes, *at syscalls and epoll/inotify.

I, cooperating with Konstantin Belousov, have managed to fix futexes to the state of passing the official futex testing program. The fix was committed and 7.0R will ship with correct futex implementation. Work is planned on removing Giant locking from futexes. This only needs some careful review and testing.

These days I mostly focus on *at syscalls, the patch is almost finished

FreeBSD Status Reports Q2/2007

for committing and I hope that it will make it into 7.0R. As a part of this work I implemented native FreeBSD syscalls as well. Watch arch mailing list as I post the patch there.

I also finished writing my master thesis describing how the Linuxulator works and Gabor Kovesdan is working on integrating it into official FreeBSD articles.

No work has happened in the epoll/inotify area but I hope to work on it right after I finish the *at syscalls.h

Open tasks:

1. Finishing *at syscalls.
2. Start the epoll/inotify work.
3. Finish removal of Giant from futexes.

lockmgr rewriting

URL: <http://wiki.freebsd.org/AttilioRao>

Contact: Attilio Rao <attilio@xxxxxxxxxxxx>

Contact: Jeff Roberson <jeff@xxxxxxxxxxxx>

The project consists in a rewriting of the lockmgr(9) interface on a lighter basis, using atomic instructions and direct usage of the sleepqueue interface. This should lead to a faster primitive, a saner interface and an higher maintainability of the code.

So far, 3 newly files called kern/kern_lockng.c, sys/_lockmgrng.h and sys/lockmgrng.h have been created for the new primitive and an initial implementation has been committed into the perforce branch:

//depot/user/attilio/attilio_lockmgr/...

The implementation contains a good set of code intended to replace old lockmgr. Actually it only misses the support for lock draining that will be committed after an initial phase of testing and the inclusion of a better wake-up algorithm (which will simplify draining a lot and will improve performance on wakeup).

Open tasks:

1. Need some testing

mtund – Magic Tunnel Daemon

URL: <http://wiki.freebsd.org/SuperTunnelDaemon>

Contact: Matus Harvan <mharvan@xxxxxxxxxxxx>

IP can easily be tunneled over a plethora of network protocols at various layers, such as IP, ICMP, UDP, TCP, DNS, HTTP, SSH. While a direct connection may not always be possible due to a firewall, the IP packets could be encapsulated as payload in other protocols, which would get through. However, each such encapsulation requires the setup of a different program and the user has to manually probe different encapsulations to find out which of them works in a given environment.

mtund is a tunneling daemon using run-time loadable plugins for the different encapsulations. It automagically selects the best encapsulation in each environment and fails over to another encapsulation in case the environment changes. There already is running code available, capable of tunneling via TCP and UDP with a working failover mechanism. As this is a Summer of Code project, rapid changes and addition of new features can be expected during the summer. Please see the wiki page for more details and up-to-date information.

Note that the project originally started under the name of Super Tunnel Daemon, but was later renamed to mtund for Magic Tunnel Daemon.

Open tasks:

1. I am always happy to hear from others trying out the code and providing feedback, both positive and negative.

Multi-link PPP daemon (MPD)

URL: <http://sourceforge.net/projects/mpd/>

URL: <http://mpd.sourceforge.net/doc/mpd5.html>

Contact: Alexander Motin <mav@xxxxxxxxxxxx>

Mpd-4.2 has been released. It includes many new features, performance improvements and fixes.

The most significant and unique new feature is a link repeater functionality. It allows mpd to accept incoming connection of any supported type and forward it out as same or different type outgoing connection. As example, this functionality allows mpd to implement real LAC with accepting incoming PPPoE connection from client and forwarding it using L2TP tunnel to LNS. All other software L2TP implementations I know is only a LAC emulators without real incoming calls forwarding abilities.

Also mpd-4.2 presents:

- * PPTP listening on multiple different IPs,
- * L2TP tunnel authentication with shared secret,
- * fast traffic filtering, shaping and rate-limiting using ng_bpf and ng_car,

- * new 'ext-auth' auth backend as full-featured local alternative to 'radius-auth',
- * NetFlow generation for both incoming and outgoing packets same time.

Replacing external ifconfig and route calls with their internal implementations and other optimizations in 4.2 gave significant performance boost in session management. Newly implemented overload protection mechanism partially drops incoming connection requests for periods of critical load by monitoring daemon's internal message queue. As result, simple 2GHz P4 system is now able to accept, authenticate and completely process spike of 1000 concurrent PPPoE connections in just a 30 seconds.

Open tasks:

1. Implement dynamic link/bundle creation.
2. Auth proxying support in repeater mode. It is required for some LAC/PAC and Tunnel Switching Aggregator (TSA) setups.
3. Remove static phys – link – bundle and phys – repeater relations. Implement ability to differentiate incoming connections processing depending on user login, domain and/or other parameters.

Multicast DNS and Service Discovery

URL: <http://wiki.freebsd.org/MulticastDNS>

Contact: Fredrik Lindberg <fli@xxxxxxxxxxx>

This project aims to create a multicast DNS daemon and service discovery utilities suitable for the base system. Multicast DNS is a part of Zero Configuration Networking (Zeroconf) and provides the ability to address hosts using DNS-like names without the need of an existing (unicast), managed DNS server. Work on the responder daemon is well underway and the only large missing piece of the puzzle is a way for local clients to do queries. The code can be found in the p4 branch projects/soc2007/fli-mdns_sd if anyone would like to give it a spin, even though it's incomplete. The project plan can be found on the wiki.

Multiprocessor Network Stack

URL: <http://www.FreeBSD.org/projects/netperf/>

Contact: Robert Watson <rwatson@xxxxxxxxxxx>

Contact: <net@xxxxxxxxxxx>

The custom file descriptor array lock has been replaced with an optimized sx lock, resulting in 2x-4x improvement in MySQL transaction rates on 8-core MySQL benchmarks. This improvement is due to moving to

shared locking for frequent fd lookup operations, as well as significant optimization of the case where the filedesc lock is highly contended (as occurs in the threaded MySQL server performing constant socket I/O).

The custom socket buffer I/O serialization lock (sblock), previously created by interlocking SB_WANT and SB_LOCK flags with the socket buffer mutex, has been replaced with an optimized sx lock, leading to a 10% performance improvement in MySQL and PostgreSQL benchmarks on 8-core systems. As part of this change, sx locks now have interruptible sleep primitives to allow the SB_NOINTR flag to work properly.

These changes also correct a long-standing bug in socket buffer lock contention and SB_NOWAIT reported by Isilon; a simpler patch has been merged to 6.x to fix this bug without merging locking changes.

TCP debugging is now properly synchronized using a new tcp_debug_mtx.

UMA allocation counters are now used for pipes rather than custom atomic counters, resulting in lowered overhead for pipe allocation and free.

Significant code cleanup, commenting, and in some cases MFC'ing, has taken place with respect to the network stack and synchronization. Additional DDB debugging commands for sockets of various sorts have been added, allowing listing of socket state from DDB without the use of GDB.

Certain non-MPSAFE subsystems have been removed or will be removed from FreeBSD 7.0, including IPX over IP tunneling (not general IPX/SPX support, just the tunneling over IP), KAME IPSEC (FAST_IPSEC is MPSAFE and now now supports IPv6), i4b, netatm (two other ATM stacks are still present), and ng_h4. Some of these features will be reintroduced in FreeBSD 7.1, but by removing them now, we are able to remove the NET_NEEDS_GIANT compatibility infrastructure that significantly complicates and obfuscates the socket and network stack code.

Other measurement and optimization projects continue; however, the 7.0 locking/synchronization work for the network stack is essentially complete.

Open tasks:

1. New work to parallelize the netisr thread (netisr2) as well as distribute UDP and TCP processing over multiple CPUs by connection, rather than just by input source as in 7.0, was presented at BSDCan. This work will be targeted at the 8-CURRENT branch.
2. Complete netatm and NET_NEEDS_GIANT removal for 7.0.
3. Complete MPSAFE locking of mld6 and nd6 IPv6 subsystems, which currently run under a global lock.

Network Stack Virtualization

URL: <http://imunes.tel.fer.hr/virtnet/>

Contact: Marko Zec <zec@xxxxxx>

The network stack virtualization project aims at extending the FreeBSD kernel to maintain multiple independent instances of networking state. This will allow for complete networking independence between jails on a system, including giving each jail its own firewall, virtual network interfaces, rate limiting, routing tables, and IPSEC configuration.

I believe that the prototype, which is kept in sync with FreeBSD -CURRENT, is now sufficiently stable for testing. It virtualizes the basic INET and INET6 kernel structures and subsystems, including IPFW and PF firewalls, and more. In the next month I plan to have the IPSEC code fully virtualized, and refine and document the management APIs. The short-term goal is to deliver production-grade kernel support for virtualized networking for FreeBSD 7.0-RELEASE (as a snap-in kernel replacement), while continuing to keep the code in sync with -CURRENT for possible merging at a later date.

OpenBSD packet filter – pf

Contact: Max Laier <mlaier@xxxxxxxxxxxx>

pf in HEAD (soon to be FreeBSD 7.0) has been updated to OpenBSD 4.1 bringing in a couple of new features:

- * ftp-proxy has been rewritten, and a tftp version, tftp-proxy, has been added
- * pf(4) now supports Unicast Reverse Path Forwarding (uRPF) checks for simplified ingress filtering
- * The pflog(4) interface is now clonable. pf(4) can log to multiple pflog interfaces now, each rule can specify which pflog interface to log to
- * pflogd(8) can now be told which pflog interface to work with
- * pfctl(8) can now expire table entries
- * keep state is now the default for pf.conf(5) rules, as is the flags S/SA option on TCP connections. no state and flags any can be used to disable stateful filtering or TCP flags checking
- * The pfctl(8) ruleset optimiser can be enabled in pf.conf(5)
- * pf(4) anchors can now be loaded inline in the main pf.conf(5) and can be printed recursively
- * Allow pf(4) rules inside anchors to have their counters reset, and make counter read & reset an atomic operation

Some patches that went into OpenBSD after 4.1 and improve performance significantly will be merged later.

FreeBSD Status Reports Q2/2007

Work to support pf and netgraph interaction is underway and will be imported after 7.0. As all required ABI changes have been made during the update, we will be able to MFC this work for 7.1 later on.

PC-BSD

URL: <http://www.pcbbsd.org/>

Contact: Kris Moore <kris@xxxxxxxx>

The last major updates are currently being made to PC-BSD 1.4, which will include KDE 3.5.7, Beryl, Flash, Intel Wireless, Nvidia Drivers and more! This release will also include new utilities to make running PC-BSD on the desktop easier than ever, including:

- * Network Manager with WIFI Support
- * Add / Remove Components
- * Firewall Manager for PF
- * Xorg Display setup wizard

Once any final major issues are resolved, we will be issuing a public beta of PC-BSD 1.4 to ensure compatibility across a variety of platforms.

Porting Linux KVM to FreeBSD

URL: <http://wiki.freebsd.org/FabioChecconi/PortingLinuxKVMToFreeBSD>

Contact: Fabio Checconi <fabio@xxxxxxxxxxxx>

Contact: Luigi Rizzo <luigi@xxxxxxxxxxxx>

The Linux kernel-based Virtual Machine (KVM) is a mechanism to exploit the virtualization extensions present in some modern CPUs (e.g., Intel VT and AMD-V). Virtualization extensions let ordinary processes execute a subset of privileged instructions in a controlled way at near-native speed. This in turn may improve the performance of system emulators such as qemu, xen, vmware, vkernel, User Mode Linux (UML), etc.

This project consists in porting to FreeBSD the Linux KVM, implemented as a loadable module, lkvm.ko. We use the approach in ports/devel/linux-kmod-compat to reuse the original Linux source code almost unmodified. We will also port a modified version of qemu which exploits the facilities made available by the Linux KVM to speed up emulation.

The URL above links to progress report detailing the exact project goals, milestones reached, and commit log details.

As of end of June 2007, we have mainly extended linux-kmod-compat to support the kernel API used by the Linux KVM code. The required

functions have been implemented at various degrees, from simple stubs to fully functional ones. We have also imported the modified qemu and the libraries that are used to build the Linux KVM userspace client. In the second half of the SoC work we plan to complete the implementation of the kernel API and have a fully functional Linux KVM module, together with its client (qemu).

Porting OpenBSD's sysctl Hardware Sensors Framework to FreeBSD

URL: <http://mojo.ru/us/GSoC2007.FreeBSD.cnst-sensors.proposal.html>

URL: <http://cnst.livejournal.com/tag/GSoC2007>

URL: <http://cnst.livejournal.com/data/atom?tag=GSoC2007>

URL:

<http://perforce.freebsd.org/depotTreeBrowser.cgi?FSPC=//depot/projects/soc2007/cnst-sensors/>

Contact: Constantine A. Murenin <cnst@xxxxxxxxxxxx>

Contact: Shteryana Shopova <syrix@xxxxxxxxxxxx>

OpenBSD includes sysctl hw.sensors framework since 2003; since 2005 the framework supports raid drives and most known i2c sensors; since 2006 the framework is redesigned with a sensor device concept in mind to accommodate continued growth. Consists of kernel api, sysctl(3)/sysctl(8), sensorsd(8), ntpd(8), systat(1), ports/sysutils/symon and 51 drivers as of 2007-07-07.

This GSoC2007 project is to port the underpinnings of this unified hardware monitoring interface to FreeBSD. Whilst it won't be possible to port all of the drivers due to architecture differences, we aim at porting all other parts of the framework and accompanying userland utilities.

At this time, lm(4) at isa and some kernel api have already been ported. The next big step is to complete sysctl(3) glue code so that further work on porting userland utilities could be accomplished. Details about sysctl are being discussed on arch@.

Open tasks:

1. sysctl(3) glue code
-

Ports Collection

URL: <http://www.freebsd.org/ports/>

URL:

http://www.freebsd.org/doc/en_US.ISO8859-1/articles/contributing-ports/

URL: <http://people.freebsd.org/~fenner/portsurvey/>

URL: <http://portsmon.FreeBSD.org/index.html>

URL: <http://www.freebsd.org/portmgr/index.html>

FreeBSD Status Reports Q2/2007

URL: <http://tinderbox.marcuscom.com>

Contact: Mark Linimon <linimon@xxxxxxxxxxxx>

The ports count is over 17,300. The PR count has been stable at around 800; we have not quite cleared up the backlog that showed up during the freeze to import xorg7.2.

There have been 4 experimental runs on the build cluster, most notably resulting in some speedups for package registration. A further experimental run to genericize autotools handling is in progress.

One of the most sweeping ports commits to happen in years was the upgrade of xorg from 6.9 to 7.2. This involved a complete rework of the internals of the port, as X.org itself has effectively pushed the responsibility for packaging to the OSes that incorporate it. The idea was to be able for them to update individual code (such as video drivers) without having to reroll the entire distribution. This commit caused us to have the longest period of preparation work, and actual tree lockdown, that I am aware of. The commit continues to be controversial, partly due to the fact that none of our port upgrade tools was up to the task of doing the upgrade without manual intervention.

At the same time that xorg was upgraded, we moved the installation directory from the obsolete /usr/X11R6 to our default /usr/local. This further complicated the upgrade.

There have been new releases of the ports tinderbox code, the portmaster update utility, and portupgrade.

GNOME was updated to 2.18.2.

We have added 7 new committers since the last report. We appreciate all the new help. However, a few committers have turned in their commit bits for safekeeping, due to lack of time.

Unfortunately, Clement Laforet has also had to step down from portmgr due to lack of time. We thank him for his help so far.

Erwin, Kris and Mark met up at BSDCan and reviewed all the portmgr-owned PRs. A large number were closed, or suspended pending more work from the submitter. After closing the PRs that were committed after the -exp builds, the number of portmgr owned PRs came down to an all time low of 48 from around 70. We hope to make further progress during the rest of the year.

Open tasks:

1. gcc4.2 has been imported to the base for 7.0. Unfortunately, this breaks a large number of ports. We need committer and maintainer

help to get these in good shape for the release.

2. Most of the remaining ports PRs are "existing port/PR assigned to committer". Although the maintainer-timeout policy is helping to keep the backlog down, we are going to need to do more to get the ports in the shape they really need to be in.

3. Although we have added many maintainers, we still have many unmaintained ports. The packages on amd64 are lagging behind a bit; those on sparc64 require even more work.

Ports Collection infrastructure improvements

URL: <http://wiki.freebsd.org/G%C3%A1borSoC2007>

Contact: Gábor Kövesdán <gabor@xxxxxxxxxxx>

Contact: Andrew Pantyukhin <sat@xxxxxxxxxxx>

Gábor Kövesdán is working on some improvements for the Ports Collection infrastructure. This year, he aimed to work on long-standing issues, which are tracked in GNATS, but we have not had a volunteer for recently. With the mentorship of Andrew Pantyukhin, he is also reimplementing the DESTDIR support for Ports Collection in a more practical way. The complete description and status of this project is available on Gábor's SoC 2007 Wiki page.

Open tasks:

1. Please see the Wiki page for the current status.

Problem Report Database

URL: <http://www.freebsd.org/support.html#gnats>

URL: <http://people.freebsd.org/~bsd/prstats/>

Contact: Mark Linimon <bugmeister_at_FreeBSD_dot_org>

Gavin Atkinson has joined the bugbuster team via getting a GNATS account on the FreeBSD cluster. He is following in the footsteps of Matteo Riondato, who later graduated to a full src commit bit. So far, he has helped close nearly 150 PRs, including many that had become stale. Welcome!

Our short-term goal is to try to identify bugs that we might be easily able to fix before the 6.3/7.0 simultaneous release. So far, great progress has been made on ata- and usb-related PRs.

The goal for the rest of this year is to generate more developer interest in fixing bugs. To do this, we are, first, trying to do more work on triaging PRs as they come in, to help flag ones that seem to be valid problems (especially if they include patches.) Secondly, we have

started a new weekly periodic posting to the `freebsd-bugbusters@xxxxxxxxxxx` mailing list, which is a short list of PRs that we feel are ready for committer action. This posting is automatically generated from a text-file list that we maintain.

We are continuing to try to manage our community's expectations of what we can do with the incoming PRs. In particular, we are trying to discourage submissions of the form "I cannot get the XYZ function to work". In practice, these PRs are not worked on. Instead, we are now encouraging these postings to go to one of the mailing lists such as `freebsd-questions@`, `freebsd-x11@`, and so forth. The idea is to emphasize GNATS as a "Problem Report" method, rather than a "general FreeBSD support" method. I feel that, otherwise, we were creating a false expectation.

The overall PR count has dropped to below 5000, despite the extra PRs still not cleared up from the ports freeze for the `xorg7.2` import. Significant progress has been made on the `i386`, `kern`, and `bin` PRs, as well as PRs in the 'feedback' state. In addition, Warner Losh has made progress on closing many of the `usb` PRs.

Open tasks:

1. Please join us on the `freebsd-bugbusters@` mailing list, or on `#freebsd-bugbusters` on EFNet, to help us triage PRs as they come in and also help us to work through the backlog, and help us to try to create a bugbusting "community".

Release Engineering

URL: <http://www.FreeBSD.org/releng/>

URL: <http://www.FreeBSD.org/snapshots/>

Contact: Release Engineering Team <`re@xxxxxxxxxxx`>

Code freeze in preparation for FreeBSD 7.0 began on June 18th. There are several large projects still being finished up as well as some issues that resulted as "fallout" from the work done just before the code freeze started (e.g. things resulting from the GCC 4.2 import). A schedule for the 7.0 release has not been set yet but the hope is that the first BETA build will be done near the end of July with a "fairly normal" release cycle (a few BETA builds followed by two or three RCs, each separated by around two weeks).

We are planning to release FreeBSD 6.3 around the same time as FreeBSD 7.0 is released so the release schedule for that will be set at the same point we set the release cycle for 7.0, hopefully late in July.

SCHED_SMP and SCHED_ULE

URL: <http://jeffr-tech.livejournal.com/>

Contact: Jeff Roberson <jeff@xxxxxxxxxxx>

SCHED_SMP is a fork of the ULE scheduler which makes use of the new fine grain scheduler locking in 7.0-CURRENT to significantly improve SMP performance on some workloads. It has improved and stronger affinity, smarter CPU load balancing, structural improvements and many sysctl tunables. This can be considered ULE 3.0. Discussions are ongoing as to whether this will go into 7.0 as SCHED_SMP or as SCHED_ULE in 7.0 or 7.1.

SCHED_ULE has had many bugfixes and performance improvements over the 7.0 development cycle and should no longer be considered unstable or experimental. On most workloads it significantly outperforms SCHED_4BSD on SMP and even slightly outperforms it on UP. There are some pathological workloads which exhibit as much as a 5% performance penalty. Many thanks to Kris Kennaway and current users for bug reports and performance testing.

Security Officer and Security Team

URL: <http://www.freebsd.org/security/>

URL:

http://www.freebsd.org/doc/en_US.ISO8859-1/articles/contributors/staff-listing.html#STAFF-SECTEAM

URL: <http://vuxml.freebsd.org/>

Contact: Security Officer <security-officer@xxxxxxxxxxx>

Contact: Security Team <security-team@xxxxxxxxxxx>

In the time since the last status report, two security advisories have been issued concerning problems in the base system of FreeBSD; both of these problems were in "contributed" code maintained outside of FreeBSD. The FreeBSD Vulnerabilities and Exposures Markup Language (VuXML) document has continued to be updated; since the last status report, 35 new entries have been added, bringing the total up to 925.

In order to improve handling of security issues in the FreeBSD Ports Collection a new "ports-security" team has been created to include ports committers who periodically help with fixing ports security issues and documenting them in the FreeBSD VuXML document. Committers who wish to help with this effort can contact simon@ for details.

The following FreeBSD releases are supported by the FreeBSD Security Team: FreeBSD 5.5, FreeBSD 6.1, and FreeBSD 6.2. The respective End of Life dates of supported releases are listed on the web site; it is expected that of the upcoming releases, FreeBSD 6.3 will be supported for two years after release, while FreeBSD 7.0 will be supported for

one year after release.

Security Regression Test

URL:

<http://perforce.freebsd.org/depotTreeBrowser.cgi?FSPC=//depot/projects/soc2007/zhouzhouyi%5fmactest%5fsoc>

Contact: Zhouyi Zhou <zhouzhouyi@xxxxxxxxxxx>

Contact: Robert Watson <rwatson@xxxxxxxxxxx>

Security Regression Test is supported by the project of Google summer code 2007. The main objective of this stage is to test the correctness of FreeBSD Mandatory Access Control Framework including correctly passing the security label from userland to kernel and non-bypassibility of Mandatory Access Control Hooks.

Work performed in the last month:

1. Constructed a pair of pseudo ethernet drivers used for testing network related hooks. To avoid the packet go through the lo interface, the IP address in the packet is twisted in the driver.
2. Constructed a framework for logging Mandatory Access Control hooks which is got called during a period of time.
 - + In kernel, every non-null label is got externalized into human readable string and recorded in a tail queue together with the name of hook that got called and possible flags or modes (etc. VREAD/VWRITE for mac_check_vnode_open hook). There is a thread much like audit subsystem's audit_worker logging the queue into a userspace file. The userland program use open, ioctl and close the /dev/mactest node to trigger and stop the logging. The logging file is truncated to zero every time the logging mechanism is triggered.
 - + In userland, a bison based parsing tool is used to parse the logged file and reconstruct the record chain which will be compared with testsuite supplied configuration file to examine if expected hooks is got called and the label/flags/modes are correct. c) The testsuite mainly follows src/tools/regression/fstest, modified to adapt to test Mandatory Access Control Framework and include tests for signals

Open tasks:

1. The code is quick and dirty. For example, a call to vn_open without checking its return value which is not fault tolerance. The coding style also needs modifications.
2. Although a test framework is completely constructed, the detailed test cases still need to be written, the test cases beside fstest