Adventures in building open source software

Sevan Janiyan - BSDCan 2015
sevan@NetBSD.org - pkgsrc
wireless hotspot

redirecting...
Building with clang

- check_options();
+ check_options(data, data, b);

- memset(ctx, 0, sizeof(ctx));
+ memset(ctx, 0, sizeof(*ctx));
SIOCSIFADDR  **struct ifreq** *  
Set the interface address for a protocol family.  
Following the address assignment, the “initialization” routine for the interface is called. 

This call has been deprecated and superseded by the SIOCAIFADDR call. 

```
struct ifreq  
vs  
struct ifaliasreq
```
char *safe_strncpy(char *dst, const char *src, size_t size) {
    if (!size) return dst;
    dst[--size] = '\0';
    return strncpy(dst, src, size);
}

VS

strlcpy(3) - #include <string.h> on modern operating systems
Total number of packages: 15114
Successfully built: 8501

Packages breaking the most other packages

<table>
<thead>
<tr>
<th>Package</th>
<th>Breaks</th>
<th>Maintainer</th>
</tr>
</thead>
<tbody>
<tr>
<td>graphics/MesaLib</td>
<td>2176</td>
<td>pkgsrc-users%NetBSD.org@localhost</td>
</tr>
<tr>
<td>devel/cmake</td>
<td>819</td>
<td>pkgsrc-users%NetBSD.org@localhost</td>
</tr>
<tr>
<td>lang/ruby200-base</td>
<td>568</td>
<td>taca%NetBSD.org@localhost</td>
</tr>
<tr>
<td>lang/gcc48-libs</td>
<td>511</td>
<td>sbd%NetBSD.org@localhost</td>
</tr>
<tr>
<td>lang/ruby193-base</td>
<td>505</td>
<td>taca%NetBSD.org@localhost</td>
</tr>
<tr>
<td>lang/ruby21-base</td>
<td>502</td>
<td>taca%NetBSD.org@localhost</td>
</tr>
<tr>
<td>devel/boost-jam</td>
<td>367</td>
<td>pkgsrc-users%NetBSD.org@localhost</td>
</tr>
<tr>
<td>x11/qt4-libs</td>
<td>359</td>
<td>pkgsrc-users%NetBSD.org@localhost</td>
</tr>
<tr>
<td>multimedia/x264-devel</td>
<td>131</td>
<td>joerg%NetBSD.org@localhost</td>
</tr>
<tr>
<td>lang/php55</td>
<td>117</td>
<td>pkgsrc-users%NetBSD.org@localhost</td>
</tr>
</tbody>
</table>

http://mail-index.netbsd.org/pkgsrc-bulk/2014/09/08/msg010950.html
# Berkeley DB's ndbm.h (since 1.85 at least) defines DBM_SUFFIX.
# Note that _DB_H_ is not defined on Mac OS X because
# it uses Berkeley DB 1 but ndbm.h doesn't include db.h.
pkgsrc-2015Q1

11224 binary packages built with gcc for Darwin 8.11.0/powerpc
10019 binary packages built with gcc for Darwin 10.8.0/i386
Autoconf

The configure tests for wctype.h & checks for the suitability of wctype_t & wctrans_t. It detects the presence of wctype.h but wctype_t & wctrans_t are unsuitable at which point it tries to use its own bundled copy of wctype.h. The build then fails as there's conflicting types for wctype_t & wctrans_t.

In file included from quotearg.c:43:
./wctype.h:724: error: conflicting types for 'wctype_t'
/opt/freeware/lib/gcc/powerpc-ibm-aix6.1.0.0/4.2.0/include/ctype.h:119: error: previous declaration of 'wctype_t' was here
./wctype.h:773: error: conflicting types for 'wctrans_t'
/usr/include/wctype.h:52: error: previous declaration of 'wctrans_t' was here

bundled wctype.h:724 typedef void * wctype_t;
bundled wctype.h:773 typedef void * wctrans_t;

system wctype.h:52  typedef wint_t (*wctrans_t)();
gcc ctype.h:119 typedef unsigned int   wctype_t;
config.guess

Attempt to guess a canonical system name.
<table>
<thead>
<tr>
<th>Package</th>
<th>Breaks</th>
<th>Maintainer</th>
</tr>
</thead>
<tbody>
<tr>
<td>lang/python27</td>
<td></td>
<td>[4984 <a href="mailto:pkgsrc-users@NetBSD.org">pkgsrc-users@NetBSD.org</a>]@localhost</td>
</tr>
</tbody>
</table>
@sevanjaniyan I thought you might like this — via Warner Losh on Facebook
<table>
<thead>
<tr>
<th>Physical</th>
<th>PageSpace</th>
<th>pages/sec</th>
<th>In</th>
<th>Out</th>
<th>FileSystemCache</th>
</tr>
</thead>
<tbody>
<tr>
<td>% Used</td>
<td>99.2%</td>
<td>0.3%</td>
<td>0.0</td>
<td>0.0</td>
<td>(numperm) 25.5%</td>
</tr>
<tr>
<td>% Free</td>
<td>0.8%</td>
<td>99.7%</td>
<td>0.0</td>
<td>0.5</td>
<td>Process 45.8%</td>
</tr>
<tr>
<td>MB Used</td>
<td>2030.8MB</td>
<td>7.6MB</td>
<td>0.0</td>
<td>0.0</td>
<td>System 27.9%</td>
</tr>
<tr>
<td>MB Free</td>
<td>17.2MB</td>
<td>2552.4MB</td>
<td>0.0</td>
<td>0.0</td>
<td>Free 0.8%</td>
</tr>
<tr>
<td>Total(MB)</td>
<td>2048.0MB</td>
<td>2560.0MB</td>
<td>0.0</td>
<td>0.0</td>
<td>Total 100.0%</td>
</tr>
</tbody>
</table>

| Min/Maxperm | 50MB(3%) | 1511MB(90%) <---% of RAM | numclient 0.7% |
| Min/Maxfree | 960 | 1088 | Total Virtual 4.5GB | maxclient 90.0% |
| Min/Maxpgahead | 2 | 8 | Accessed Virtual 1.4GB 32.2% | User 55.0% |
|             |          |             |         | **Pinned** 55.4% | lruable pages 429664.0 |


bootstrap with CC=/usr/bin/gcc
No error

Change shell to pdksh:
bmake: don't know how to make pbulk-index. Stop
pbulk-scan: realloc failed:

Free RAM, raise resource limits to 256MB:
/usr/pkgsrc/pbulk/libexec/pbulk/scan[54]: 11272416 Segmentation fault(coredump).

Coredump is available but dbx is not installed.

Unable to install dbx because dependencies missing.

bootstrap again without $CC set:
/bin/sh: There is no process to read data written to a pipe..

bootstrap again without $CC set & script modified to use pdksh by default
Memory fault (core dumped).
ld: 0711-596 SEVERE ERROR: Object xxx.o
An RLD for section 2 (.data) refers to symbol 111, but the storage class of the symbol is not C_EXT or C_HIDEXT. The source file contains implicitly initialized global symbols

FreeBSD/amd64
Total number of packages: 16319
Successfully built: 13615
Solaris 11 SPARC
Total number of packages: 15264
Successfully built: 7190

Solaris 11 x86
Total number of packages: 15264
Successfully built: 5911

Solaris 10 SPARC
Total number of packages: 15264
Successfully built: 8094

OmniOS
Total number of packages: 16312
Successfully built: 8056
OpenBSD/sparc64
Total number of packages: 15246
Successfully built: 5899

OpenBSD/amd64
Total number of packages: 16274
Successfully built: 8770
Bitrig/amd64

Total number of packages: 16319
Successfully built: 5183
Total number of packages: 16312
Successfully built: 11863
Linux/ppc64le
Total number of packages: 15036
Successfully built: 11313
Mandoc support for Solaris improved
http://permalink.gmane.org/gmane.comp.tools.mdocml.devel/802

FreeBSD - pax(1) gets -o flag
https://bugs.freebsd.org/bugzilla/show_bug.cgi?id=198481

DragonFlyBSD - Deadlock issues fixed

BSD Make - Found use of memcpy between overlapping buffers
https://bugs.freebsd.org/bugzilla/show_bug.cgi?id=199486

OpenSSH-portable - updating config.guess added to checklist
https://bugzilla.mindrot.org/show_bug.cgi?id=2409#c3
I'm surprised we didn't catch it until now. your pkgsrc work gave us a dividend :-) . It's a serious bug but only having a dozens of processes on a big multi-core machine like monster can trigger it for review: http://apollo.backplane.com/DFlyMisc/fflag01.patch that's what I'm compiling on monster now. will commit later this evening was triggered by Sevan's pkgsrc build. apparently calling fcntl F_SETFL or flock() on a log file that is being appended to by lots of processes all at once was the trigger. caused them to get stuck do to an improper update of the fp->f_flag field. hopefully that is what the cause was. we will know by tomorrow
you hit a procfs bug it looks like. I'm looking at it hmm.
something else is going on. there's a token live lock. how very odd
I believe I've found the issue.
It's actually a bug in vm_fault in an optimized copyout_nofault path
there's an optimization that is supposed to allow trivial vm_faults but in this case the
vm_fault had to do a copy-on-write after a fork, which is not trivial, and then deadlocked
on a lock.
basically there is an optimization for uiomove() (moving data to userspace) and copyin
and copyout that allows those functions to be called with VM objects held without
deadlocking, causing them to fail if they would otherwise deadlock the caller then undoes
his fast path and reexecutes on the slow path with normal uiomove()'s and such its used to
allow (mainly) tmpfs to issue uiomove() ops deep in its VM object / VM page handling
code to handle reads and writes optimally there was a case not being handled by vm_fault
related to when the vm_map_lookup() has to shadow a VM page due to a copy on write.
we've run lots of dports bulk builds without hitting this issue. I think there is just something
the pkgsrc bulk build is doing which is revealing the issues hopefully there won't be very
many more, the whole idea is for it to be a stable platform for bulk builds :-)