

FreeBSD Debugging Tips And Tricks

Andriy Gapon <avg@FreeBSD.org>

KyivBSD, 2010

Outline

- 1 Build
 - Kernel Options
 - Build, Installation And Boot Options
- 2 Debuggers
 - DDB
 - Textdump
 - KGDB
 - GDB And Qemu
- 3 DTrace
 - Probes
 - Language
 - Preparing for DTrace
 - Examples

Introduction/Overview

- Why we need you
- Types of bugs and tools

Introduction/Overview

- Why we need you
- Types of bugs and tools

Kernel Configuration Options

- Debug symbols and optimization

```
makeoptions DEBUG="-g -O1"
```

- KDB options, see `sys/conf/NOTES`

```
options KDB  
options KDB_TRACE  
#options KDB_UNATTENDED
```

- Backends

```
options DDB  
#options GDB
```

Kernel Configuration Options

- Debug symbols and optimization

```
makeoptions DEBUG="-g -O1"
```

- KDB options, see sys/conf/NOTES

```
options KDB  
options KDB_TRACE  
#options KDB_UNATTENDED
```

- Backends

```
options DDB  
#options GDB
```

Kernel Configuration Options

- Debug symbols and optimization

```
makeoptions DEBUG="-g -O1"
```

- KDB options, see sys/conf/NOTES

```
options KDB  
options KDB_TRACE  
#options KDB_UNATTENDED
```

- Backends

```
options DDB  
#options GDB
```

Crash dump

- Don't forget dumpdev!
- Can be set only on a booted system, dumpon(8)
- Can be set via rc.conf

```
rc.conf
```

```
dumpdev="/dev/gpt/9VM0V8VZ.swap"  
dumpdev="AUTO"
```


Crash dump

- Don't forget dumpdev!
- Can be set only on a booted system, dumpon(8)
- Can be set via rc.conf

```
rc.conf
```

```
dumpdev="/dev/gpt/9VM0V8VZ.swap"  
dumpdev="AUTO"
```

Crash dump

- Don't forget dumpdev!
- Can be set only on a booted system, dumpon(8)
- Can be set via rc.conf

rc.conf

```
dumpdev="/dev/gpt/9VM0V8VZ.swap"  
dumpdev="AUTO"
```

Kernel Configuration Options

- Self-Diagnostics, see `sys/conf/NOTES`

```
options INVARIANTS  
options WITNESS
```

- Other debug options

```
options CAMDEBUG
```

- Miscellaneous options

```
options SC_HISTORY_SIZE=1000 #See sc(4)  
options MSGBUF_SIZE=131072  
options INCLUDE_CONFIG_FILE
```

Kernel Configuration Options

- Self-Diagnostics, see sys/conf/NOTES

```
options INVARIANTS  
options WITNESS
```

- Other debug options

```
options CAMDEBUG
```

- Miscellaneous options

```
options SC_HISTORY_SIZE=1000 #See sc(4)  
options MSGBUF_SIZE=131072  
options INCLUDE_CONFIG_FILE
```

Build, Installation And Boot Options

- Configuring multiple kernel variants

KERNELdbg

```
ident KERNELdbg
include KERNEL
nooption OPTNAME
nodevice devname
```

Build, Installation And Boot Options

- Building multiple kernel variants

```
make buildkernel KERNCONF=KERNELdbg
make buildkernel KERNFAST=KERNELdbg
make buildkernel SRCCONF=/path/to/src.conf
__MAKE_CONF=/path/to/make.conf
... NO_MODULES ... MODULES_OVERRIDE ...
```

- Cross-building kernel

```
make kernel-toolchain TARGET=i386
make buildkernel TARGET=i386
```

Build, Installation And Boot Options

- Installing multiple kernels

```
make installkernel KERNCONF=KERNELdbg  
INSTKERNNAME=kernel.dbg  
make installkernel DESTDIR=/alt/root
```

- Choosing a kernel to boot

```
nextboot -k kernel.dbg -o -v
```

nextboot doesn't properly work with ZFS

Build, Installation And Boot Options

- Installing multiple kernels

```
make installkernel KERNCONF=KERNELdbg  
INSTKERNNAME=kernel.dbg  
make installkernel DESTDIR=/alt/root
```

- Choosing a kernel to boot

```
nextboot -k kernel.dbg -o -v
```

nextboot doesn't properly work with ZFS

Loader

- Learn basic loader variables and commands before pressing “6”
- Loader online help: 'help' and '?'

Loader prompt

```
unload
show
lsdev
set currdev=disk1s1a
set kernel=kernel.dbg
set vfs.root.mount_from="ufs:/dev/ad4s1a"
boot
```

Loader

- Learn basic loader variables and commands before pressing “6”
- Loader online help: 'help' and '?'

Loader prompt

```
unload  
show  
lsdev  
set currdev=disk1s1a  
set kernel=kernel.dbg  
set vfs.root.mount_from="ufs:/dev/ad4s1a"  
boot
```

Loader

- Learn basic loader variables and commands before pressing “6”
- Loader online help: 'help' and '?'

Loader prompt

```
unload  
show  
lsdev  
set currdev=disk1s1a  
set kernel=kernel.dbg  
set vfs.root.mount_from="ufs:/dev/ad4s1a"  
boot
```

DDB

- Available online
- Lots of useful built-in commands
- No access to debug symbols

DDB

- Available online
- Lots of useful built-in commands
- No access to debug symbols

DDB

- Available online
- Lots of useful built-in commands
- No access to debug symbols

Textdump

- `textdump(4)` is enabled by `KDB` and `DDB` options
- `ddb(8)` - configure `ddb(4)` scripting

```
ddb script kdb.enter.panic="textdump set; capture on; bt; call  
doadump; reset"
```

- Can be configured in `rc.conf`

```
rc.conf
```

```
ddb__enable="YES"  
ddb__config="/etc/ddb.conf"
```

Textdump

- textdump(4) is enabled by KDB and DDB options
- ddb(8) - configure ddb(4) scripting

```
ddb script kdb.enter.panic="textdump set; capture on; bt; call  
doadump; reset"
```

- Can be configured in rc.conf

```
rc.conf  
ddb_enable="YES"  
ddb_config="/etc/ddb.conf"
```


Textdump

- textdump(4) is enabled by KDB and DDB options
- ddb(8) - configure ddb(4) scripting

```
ddb script kdb.enter.panic="textdump set; capture on; bt; call  
doadump; reset"
```

- Can be configured in rc.conf

```
rc.conf
```

```
ddb__enable="YES"  
ddb__config="/etc/ddb.conf"
```

KGDB

- Live kernel examination

```
kgdb
```

```
info line *0xffffffff80539b60
```

- Post-mortem analysis
- Serial/Firewire console
- GDB scripting, Scripts from John Baldwin

```
http://people.freebsd.org/~jhb/gdb/
```

KGDB

- Live kernel examination

```
kgdb
```

```
info line *0xffffffff80539b60
```

- Post-mortem analysis
- Serial/Firewire console
- GDB scripting, Scripts from John Baldwin

```
http://people.freebsd.org/~jhb/gdb/
```

KGDB

- Live kernel examination

```
kgdb
```

```
info line *0xffffffff80539b60
```

- Post-mortem analysis
- Serial/Firewire console
- GDB scripting, Scripts from John Baldwin

```
http://people.freebsd.org/~jhb/gdb/
```

KGDB

- Live kernel examination

```
kgdb
```

```
info line *0xffffffff80539b60
```

- Post-mortem analysis
- Serial/Firewire console
- GDB scripting, Scripts from John Baldwin

```
http://people.freebsd.org/~jhb/gdb/
```

GDB And Qemu

- Qemu

```
qemu ... -S -s
```

- Remote GDB

```
GDB session
```

```
(gdb) target remote localhost:1234
```

Probes

- fbt
- systrace
- sdt
- profile

Probes

- fbt
- systrace
- sdt
- profile

Probes

- fbt
- systrace
- sdt
- profile

Probes

- fbt
- systrace
- sdt
- profile

Language

- Variables, functions

```
self->ts = timestamp;
```

- Predicates

```
/self->trace && stringof(args[0]->uk_name) == "zio_buf_512"/
```

- Aggregations

```
@times[stack()] = quantize(timestamp - self->ts);
```

Language

- Variables, functions

```
self->ts = timestamp;
```

- Predicates

```
/self->trace && stringof(args[0]->uk_name) == "zio_buf_512"/
```

- Aggregations

```
@times[stack()] = quantize(timestamp - self->ts);
```

Language

- Variables, functions

```
self->ts = timestamp;
```

- Predicates

```
/self->trace && stringof(args[0]->uk_name) == "zio_buf_512"/
```

- Aggregations

```
@times[stack()] = quantize(timestamp - self->ts);
```

Preparing for DTrace

<http://wiki.freebsd.org/DTrace>

- CTF, WITH_CTF
- Kernel options

```
options KDTRACE_HOOKS
options DDB_CTF
options KDTRACE_FRAME # amd64-only
makeoptions WITH_CTF=1
```

- DTrace modules

```
kldload fbt
kldload dtraceall
```

Preparing for DTrace

<http://wiki.freebsd.org/DTrace>

- CTF, WITH_CTF
- Kernel options

```
options KDTRACE_HOOKS
options DDB_CTF
options KDTRACE_FRAME # amd64-only
makeoptions WITH_CTF=1
```

- DTrace modules

```
kldload fbt
kldload dtraceall
```

Preparing for DTrace

<http://wiki.freebsd.org/DTrace>

- CTF, WITH_CTF
- Kernel options

```
options KDTRACE_HOOKS
options DDB_CTF
options KDTRACE_FRAME # amd64-only
makeoptions WITH_CTF=1
```

- DTrace modules

```
kldload fbt
kldload dtraceall
```


Examples

```
fbt:kernel:wakeup:entry
/(void*)arg0 == (void*)&kernel'cnt.v_free_count/
{
    @stacks[stack()] = count();
    @tids[pid, tid] = count();
}
```

Examples

```
fbt::vm_page_wire:entry
/args[0]->wire_count == 0/
{
    @new[stack(10)] = count();
}
```

Examples

```
profile:::profile-1001
{
    @stacks[curthread->td_oncpu, pid,
            tid, execname, stack()] = count();
}
END {
    printf("\n\n");
    printf("***** Top Processes:\n");
    trunc(@stacks, 20);
    printa(@stacks);
}
```

Summary

- Debugging in a wide sense has many faces: diagnostics, debuggers, checks
- Debugging is not as hard as it may seem
- Outlook
 - ktr(4) for scheduler debugging
 - hwpmc(4) for system and userland profiling
 - SW_WATCHDOG and MP_WATCHDOG for deadlock debugging
 - DTrace in userland

For Further Reading I

man(1)
make(1)
config(5)
ddb(4)
textdump(4)
make.conf(5)
src.conf(5)
build(7)
ddb(8)
loader(8)
nextboot(8)

NOTES files
GDB manuals
<http://sourceware.org/gdb/>
FreeBSD Handbook
FreeBSD Developers' Handbook
FreeBSD Wiki
Google

References



Kernel Debugging
[FreeBSD Developers' Handbook.](#)



[John H. Baldwin.](#)
Introduction to Debugging the FreeBSD Kernel.
[BSDCan 2008.](#)