NAME

dtrace_kinst - a DTrace provider for tracing arbitrary instructions in a given kernel function

SYNOPSIS

kinst::<function>:<instruction>

DESCRIPTION

The DTrace **kinst** provider allows the user to trace any instruction in a given kernel function. <function> corresponds to the function to be traced, and <instruction> is the offset to the specific instruction, and can be obtained from the function's disassembly using kgdb from the gdb package.

kinst creates probes on-demand, meaning it searches for and parses the function's instructions each time dtrace(1) is run, and not at module load time. This is in contrast to FBT's load-time parsing, since **kinst** can potentially create thousands of probes for just a single function, instead of up to two (entry and return) in the case of FBT. A result of this is that **dtrace -l -P kinst** will not match any probes.

IMPLEMENTATION NOTES

The provider is currently implemented only for amd64.

EXAMPLES

Find the offset corresponding to the third instruction in **vm_fault**() and trace it, printing the contents of the RSI register:

```
# kgdb
(kgdb) disas /r vm_fault
Dump of assembler code for function vm_fault:
   0xffffffff80876df0 <+0>:     55      push   %rbp
   0xffffffff80876df1 <+1>:     48 89 e5        mov    %rsp,%rbp
   0xffffffff80876df4 <+4>:     41 57   push   %r15

# dtrace -n 'kinst::vm_fault:4 {printf("%#x", regs[R_RSI]);}'
  2  81500                 vm_fault:4 0x827c56000
  2  81500                 vm_fault:4 0x827878000
  2  81500                 vm_fault:4 0x1fab9bef0000
  2  81500                 vm_fault:4 0xe16cf749000
  0  81500                 vm_fault:4 0x13587c366000
  ...
```

Trace all instructions in **amd64_syscall**():

```
# dtrace -n 'kinst::amd64_syscall:'
```

SEE ALSO

dtrace(1)

HISTORY

The **kinst** provider first appeared in FreeBSD 14.0.

AUTHORS

This manual page was written by Christos Margiolis <*christos@FreeBSD.org*>.