Ruby master - Feature #14813

[PATCH] gc.c: make gc_enter+gc_exit pairs dtrace probes, too

06/02/2018 02:05 PM - normalperson (Eric Wong)

Status:	Open
Priority:	Normal
Assignee:	ko1 (Koichi Sasada)
Target version:	

Description

gc.c: make gc_enter+gc_exit pairs dtrace probes, too

I would like to use these with systemtap to gather
min/max/avg/variance data for gc_*_continue functions.

I prefer to use systemtap/dtrace and not modify or load extra
Ruby code to use built-in tracing. Systemtap also has aggregate
functionality built in for @min/@max/@avg and @hist_log for
generating histograms:

https://80x24.org/spew/20180602135820.6686-1-e@80x24.org/raw
(work-in-progress)

I will add documentation to probes.d and doc/dtrace_probes.rdoc
if accepted.

History

#1 - 06/05/2018 11:23 PM - normalperson (Eric Wong)

> https://bugs.ruby-lang.org/issues/14813
>
> I would like to use these with systemtap to gather
> min/max/avg/variance data for gc_*_continue functions.

Fwiw, I tried using function tracing for this, but they get
inlined, timing becomes impossible :<

#2 - 06/06/2018 08:42 AM - normalperson (Eric Wong)

> https://bugs.ruby-lang.org/issues/14813

Fwiw, I'm still learning systemtap myself; and I made r63581
the dtrace tests to work with systemtap.

I played around with systemtap ~5 years ago for other projects
and forgot much of it :x

For Debian/Ubuntu users:

```
# as root:
apt-get install systemtap systemtap-sdt-dev
stap-prep # should install necessary kernel headers + debug symbol(*)
adduser $YOUR_USER stapusr
adduser $YOUR_USER stapdev
```

Red Hat-based systems should be similar; and most of the
systemtap team works for Red Hat. There's also dyninst support
which won't require special privileges (or kernel context
switch), but current Debian stable doesn't enable it, yet.

```
# (re-)login as regular user:
```

```
# rebuild Ruby with --enable-dtrace
# (make sure probes.h has SDT stuff and isn't a dummy file)

# trace a new command
$ stap -v script.stp -c "$RUBY_CMD"

# trace running command
$ stap -v script.stp -x $PID_OF_RUBY
```

My original example with:

```
probe process("/path/to/ruby").mark("foo")
```

was a bit strict since it requires the path of executable.
Now I favor "-x $PID" or "-c $CMD" so I can use:

```
probe process.mark("foo")
```

There is also "systemtap-doc" package
I look at docs + *.stp examples in their git tree:

```
git://sourceware.org/git/systemtap.git
```

(*) I never tested `stap-prep' myself since I build my own kernels
from source. For people like me who run the latest kernels;
be prepared to backport patches from systemtap.git or
build+install from that yourself. Users of distro-provided
kernels get things working out-of-the-box in my experience.

#3 - 06/23/2018 07:07 PM - vo.x (Vit Ondruch)

Just FTR, Ruby in Fedora/RHEL has built in support for SystemTap.

Not sure what r63581 really does, but you might find useful these two .stp files:

https://src.fedoraproject.org/rpms/ruby/blob/master/f/libruby.stp
https://src.fedoraproject.org/rpms/ruby/blob/master/f/ruby-exercise.stp

#4 - 12/10/2018 06:56 AM - ko1 (Koichi Sasada)

sorry to left this issue.

I have no strong idea. If it will help for systemtap user. I don't care.
(I feel changinge "gc_enter" to "GC_ENTER" is a bit difficult to read, but I understand why...).

#5 - 12/16/2018 12:22 PM - normalperson (Eric Wong)

> https://bugs.ruby-lang.org/issues/14813#change-75529
>
> I have no strong idea. If it will help for systemtap user. I don't care.
> (I feel changinge "gc_enter" to "GC_ENTER" is a bit difficult to read, but I understand why...).

I think the caps change is a minor issue. It's more important to
use ALL_CAPS to help identify macros (because of side-effects).

Anyways, updated patch here:
https://80x24.org/spew/20181216120630.17237-2-e@80x24.org/raw
(tested on FreeBSD 11.2 and Debian 9.6)

Still waiting on tenderlovemaking (Aaron Patterson) for [Bug #15399] on
duparray/duphash hook tracing comments

Sidenote:

Since I always build my own kernels and usually run the latest
kernel, systemtap remains a pain because it relies heavily on
kernel internals. Given that, systemtap should probably be like
perf(1) and be maintained/distributed with the kernel (and the
default prefix should be $HOME :P)

#6 - 12/17/2018 07:42 AM - ko1 (Koichi Sasada)

On 2018/12/16 21:20, Eric Wong wrote:

> I think the caps change is a minor issue. It's more important to
> use ALL_CAPS to help identify macros (because of side-effects).

I mean, to reach "gc_enter" function is a bit difficult because we need
to read (understand) GC_ENTER() macro. Additional code introduce new
difficulty (yes, it is trivial).

--
// SASADA Koichi at atdot dot net

Files

0001-gc.c-make-gc_enter-gc_exit-pairs-dtrace-probes-too.patch	4.94 KB	06/02/2018	normalperson (Eric Wong)