ruby/ruby on rails memory leak detection

Asked 16 years, 2 months ago   Modified 5 years, 5 months ago

Viewed 38k times

▲

47

▼

I wrote a small web app using ruby on rails, its main purpose is to upload, store, and display results from xml(files can be up to several MB) files. After running for about 2 months I noticed that the mongrel process was using about 4GB of memory. I did some research on debugging ruby memory leaks and could not find much. So I have two questions.

- Are there any good tools that can be used to find memory leaks in Ruby/rails?

- What type of coding patterns cause memory leaks in ruby?

`ruby-on-rails`   `ruby`   `memory`   `memory-leaks`   `coding-style`

Share

Improve this question

Follow

edited Apr 6, 2010 at 23:10

BalusC
1

asked Oct 2, 2008 at 8:15

Josh Moore
13.5k ● 15 ● 61 ● 74

7 Answers

Some tips to find memory leaks in Rails:

- use the [Bleak House](#) plugin

- implement [Scout monitoring](#) specifically the memory usage profiler

- try another [simple memory usage logger](#)

40

The first is a graphical exploration of memory usage by objects in the ObjectSpace.

The last two will help you identify specific usage patterns that are inflating memory usage, and you can work from there.

As for specific coding-patterns, from experience you have to watch anything that's dealing with file io, image processing, working with massive strings and the like.

I would check whether you are using the most appropriate XML library - ReXML is known to be slow and believed to be leaky (I have no proof of that!). Also check whether you can [memoize](#) expensive operations.

Share   Improve this answer

Follow

edited Jul 9, 2019 at 20:02

Dave Powers
2,353 ● 2 ● 35 ● 37

answered Oct 2, 2008 at 11:53

9   The post for Bleak House is from 2007, is that still a valid option today? – Pablo Fernandez May 8, 2012 at 12:27

▲

23

▼

🔖

🕘

A super simple method to log memory usage after or before each request (only for Linux).

```
#Put this in applictation_controller.rb
before_filter :log_ram # or use after_filter
def log_ram
  logger.warn 'RAM USAGE: ' + `pmap #{Process.pid} | t
end
```

You might want to load up script/console and try the statement out first to make sure it works on your box.

```
puts 'RAM USAGE: ' + `pmap #{Process.pid} | tail -1`[1
```

Then just monitor top, when a request makes your memory usage jump, go check the logs. This, of course, will only help if you have a memory leak that occurs in large jumps, not tiny increments.

Share   Improve this answer

Follow

answered Oct 2, 2008 at 19:46

---

where is the log_ram method from? – Joshua Partogi Aug 23, 2010 at 7:07

7   on osx you can use vmmap #{Process.pid}
     – rmontgomery429 Sep 26, 2011 at 19:18

6

Memory leak is a problem in the current ruby implementation a good place to start about this is ~~http://whytheluckystiff.net/articles/theFullyUpturnedBin.html~~ Whytheluckystiff website doesn't exist anymore but you can find the original article here: https://viewsourcecode.org/why/hacking/theFullyUpturnedBin.html

for a more specific answer on problems with long running ruby processes see https://just.do/2007/07/18/heap-fragmentation-in-a-long-running-ruby-process/

maybe you could give passenger (mod_rails) a try https://web.archive.org/web/20130901072209/http://nubyonrails.com/articles/ask-your-doctor-about-mod_rails

Share Improve this answer

Follow

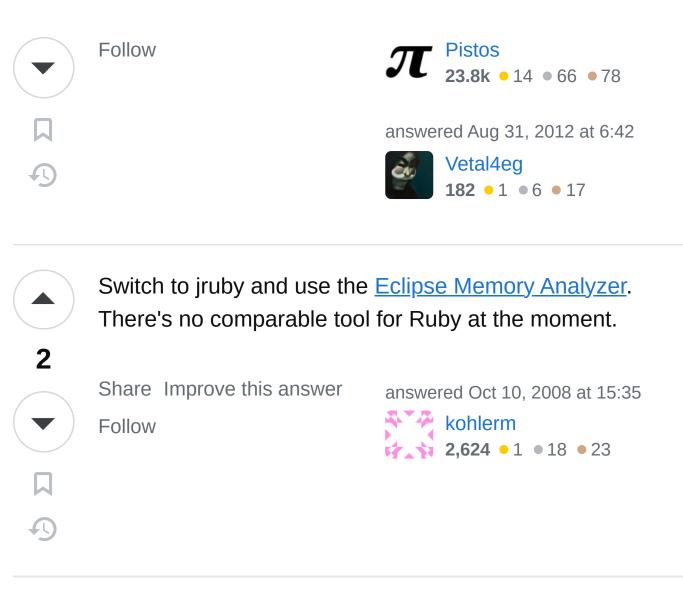edited Jul 10, 2019 at 4:44

Dave Powers
2,353 ● 2 ● 35 ● 37

answered Oct 2, 2008 at 10:35

Jean
21.6k ● 5 ● 48 ● 64

5

You should give a look to ruby-prof.

Share Improve this answer   edited Nov 10, 2014 at 14:47

answered Aug 31, 2012 at 6:42

2

Switch to jruby and use the [Eclipse Memory Analyzer](#). There's no comparable tool for Ruby at the moment.

Share Improve this answer

answered Oct 10, 2008 at 15:35

2

Now, you can run the following to get the memory in a format that R can read. I am assuming that your log line looks like:

```
1234567890 RAM USAGE: 27456K
```

Run this (or modify to suite):

```
$ grep 'RAM USAGE' fubar.log | awk '{print s " " $1 "
's/K//g' > mem.log
```

Then you can run this:

```
#!/bin/sh
rm -f mem.png
R --vanilla --no-save --slave <<RSCRIPT
    lst <- read.table("mem.log")
    attach(lst)
    m = memory / 1024.0
    summary(m)
    png(filename="mem.png", width=1024)
    plot(date, m, type='l', main="Memory usage", xlab=
RSCRIPT
```

and get a nice graph.

Share Improve this answer

Follow

answered Mar 4, 2010 at 8:32

Sardathrion - against SE abuse

18.2k ● 29 ● 107 ● 162

These gems worked for me:

0

MemoryLogic

> Adds in proccess id and memory usage in your rails logs, great for tracking down memory leaks

Oink

> Log parser to identify actions which significantly

> increase VM heap size

Share Improve this answer

Follow