

Isolating a rampant process in IIS

Asked 16 years ago Modified 15 years, 11 months ago Viewed 1k times



1



I have a webserver that is pegged and I've been able to isolate it to a particular website instance. I'd like to dig deeper and isolate the particular page/process that is causing the issue.. Any tips?

iis



Share

Improve this question

Follow

asked Dec 24, 2008 at 18:53



[madcolor](#)

8,174 ● 11 ● 53 ● 74

This turned out to be a SQL problem (sql 2005). The solution was found by using SQL activity monitor to identify a suspended process with a Async_network_io wait type. We then ran SQL profiler to narrow it down to two massive queries which were returning an abundance of results.

– [madcolor](#) Dec 31, 2008 at 17:29

5 Answers

Sorted by:

Highest score (default)



2

You can take a memory dump of the process and poke around with windbg.



There are [posts on this issue](#) from Tess Ferrandez blog. Just do as she say.



Share Improve this answer

answered Dec 24, 2008 at 19:29

Follow



[Igal Serban](#)

10.7k ● 3 ● 37 ● 40



0



Which version of IIS are you using? Some of the higher ones allow for a separation of which process gets used to handle requests such as a worker process that you could isolate a bit more that way. I'd also suggest reading through the IIS logs to see what requests were being handled, how long they took, etc.



There are many different quirks to each IIS version. The really low ones just had a start/stop functionality, but the newer ones have really given administrators much more control and power, IMO.

Share Improve this answer

answered Dec 24, 2008 at 18:57

Follow



[JB King](#)

11.9k ● 4 ● 40 ● 49



0



You should try using a profiler to identify what is using up the most resources. I've used [dotTrace Profiler](#), although that can be expensive if you're on a tight budget.



It allows you to see exactly what processes and method calls use of the most processing time of a request really



well so you can isolate the most resource intensive operations.

You should really be able to use any profiler to do this, not just dotTrace. I just happen to only have experience with this one in particular.

Share Improve this answer

answered Dec 24, 2008 at 19:08

Follow



Dan Herbert

103k ● 51 ● 192 ● 221



0

Change your web garden setting to 10 or greater. Then watch your CPU and memory utilization on the web server.



Continue to increase the web garden setting until either the app is completely responsive with less than 5% average utilization OR you have actually maxed your web server's memory.



UPDATE

It's not about diagnosing, it's about properly configuring the IIS server. Web Gardens are one of the top misunderstood features of IIS. By increasing the available threads to handle new requests you remove the appearance of contention at the web server level and place it squarely where it belongs. In this case at your database. Instead of masking a problem it actually highlights exactly where the problem is.

Share Improve this answer

edited Dec 31, 2008 at 15:33

Follow

answered Dec 31, 2008 at 1:38



[ChrisLively](#)

88k ● 27 ● 173 ● 247

Will this help diagnose, or just put a bandaid on the problem... btw, we think we've narrowed this down to a poorly called SP, but are still investigating. When we kill the SP process on SQL, the problem seems to vanish. – [madcolor](#)
Dec 31, 2008 at 15:20



0



This turned out to be a SQL problem (sql 2005). The solution was found by using SQL activity monitor to identify a suspended process with a Async_network_io wait type. We then ran SQL profiler to narrow it down to two massive queries which were returning an over abundance of results.



Share Improve this answer

Follow

answered Dec 31, 2008 at 17:54



[madcolor](#)

8,174 ● 11 ● 53 ● 74
