Sick Server Troubleshooting Part 1 - Gathering Information

Posted At : August 11, 2008 1:21 PM | Posted By : Mark Kruger
Related Categories: Coldfusion Troubleshooting

I get a lot of calls from folks with dedicated Coldfusion servers who have suddenly developed problems. Usually the server has suddenly stopped responding in a seemingly random fashion. Often the caller indicates that Jrun is "pegged" at 99 percent. If you search through this blog (and many others) you will find a great number of tips and hints on how to attack this problem. The next three posts summarize *my own* process and give you a quick resource to the posts regarding troubleshooting a sick server.

Keep in mind that there is *no substitute for experience*. If you are novice at this you will need to get comfortable with the idea that you will spend several hours (sometimes days) working through possibilities. If the server is mission critical and there is money at stake you should consider calling in the cavalry. A troubleshooter is also a unique animal - a "technologist" of sorts. The best troubleshooters have gathered a good deal of knowledge and experience in several areas (app server specifics, database, hardware, networking etc). The best one s are also a special breed that think in a certain way. They make lists, figure out test patterns, know when to make a leap and when to keep digging etc. So with that in mind, let's talk about how to start.

Step 1 is to gather information.

Take Inventory

First, take stock of the whole system. Log into the desktop and the CF Admin and gather the following information.

- **Hardware Resources** - How much RAM, What type of disks, How many processors and processor cores.
- **Software Resources** - Other than the Web server and Coldfusion *what else* is the server tasked with doing? What processes are running? What version of Apache or IIS? What version and updater of CF? Are .NET sites also running on the server? How about PHP?
- **JVM Settings** - Go directly to the file system and open the file in /runtime/bin and copy the arguments to your notes.
- **Coldfusion Settings** - What are the settings for client variables? What are the defaults for session and in memory variables? How many scheduled tasks are running? What is the simultaneous request setting? Make sure you are familiar with the whole setup.
- **Database Resources** - What databases are you using and where are they located relative to the web server. If you are able, log into the DB server desktop and take the same inventory. Take note of the *way ColdFusion connects* to the DB (driver and IP or FLQN?). Are there any Access Databases? NOTE: This is important. About 70 percent of the time the database is involved in performance problems. It's important to remember that the web server and database server work together as a single system.
- **Network Resources** - What is the speed and duplex of the links? Can you determine the settings of the downstream port at the switch?

- **Traffic and Requests** - Is there a measurable change in traffic? If so, you could have simply reached a threshold and need upgrading to a more scalable solution.

This is *not* the time to draw conclusions. Gather information and resist the temptation to think you have found a smoking gun. Even folks with great experience find themselves reinstalling a server when all they really needed was to tweak a network or JVM setting or reindex a database. Make sure you have all the information at your disposal before you start making changes.

Gather Log Information

Next, gather information from the log files. These include the following logs:

- **ColdFusion Logs** - These are found in the <coldfusion files>/logs. There should be an application.log an exception.log, a server.log and a few others. Take a look at these logs and in particluar try to find the spot in the log where the problem began. This is not always possible.
- **JVM Logs** - Don't forget these! They are loced in the /runtime/bin directory and include an *-err.log, a *-out.log and some other logs that can be useful. Generally these logs include more stack information and they are more verbose.
- **Web Server Logs** - Take a look at the raw logs and check out things like user agents, query strings and IP addresses. You can sometimes stumble onto an attack or a malformed request of malicious intent (sort of a Rodent of Unusual Size).
- **System Logs** - In windows this would be the event viewer. Look for disk errors or IO errors of any kind. Look for anything else that seems suspicious. Google things that catch your attention.
- **JRUN Metrics** - You can enable metrics and get some excellent counters that can help you figure out the when and what of an outage. See this post for a good how to.

Now that you have taken the pulse of your system you are ready to shake it down a little bit. In step 2, we will discuss what to try.