

Simplify Process Troubleshooting with DebugDiag

Michael Morales

(Reprinted From WindowsItPro Magazine)

Executive Summary

The Microsoft Debug Diagnostic Tool (DebugDiag) tool makes it easy to troubleshoot Microsoft Windows process problems, such as crashes, Microsoft IIS hangs, or memory leaks. DebugDiag has a rich GUI and simplifies analysis of user dump files, so that you can more quickly resolve process problems.

When troubleshooting application-stability concerns and performance problems such as crashes, hangs, and high memory usage, sometimes you need to examine the process that was active when the problem occurred. To complicate troubleshooting, server applications such as Microsoft IIS, Exchange Server, SQL Server, COM+, and BizTalk Server often display no UI and restart automatically without indicating what caused them to fail. Having the right debugging tool to isolate a problem can make finding the solution much easier. For such problems, Debug Diagnostic Tool (DebugDiag) is often a better choice than other debugging tools such as ADPlus, Userdump, and WinDbg. I'll explain why and will walk you through using Debug-Diag to troubleshoot a process crash.

Why Use DebugDiag?

To understand why DebugDiag is often a good choice for Windows process troubleshooting, let's first look at why a process might crash. A process crash is an unexpected program termination when a process exits abnormally. Typically the crash is caused by an unhandled exception; however, it could also occur when the process detects a problem condition and exits without an exception (for instance, process recycling caused by excess memory utilization).

A commonly used workaround is to restart the process or service in hopes that whatever caused the crash will no longer occur. But to really determine what caused the problem and to fix it, you must analyze the process state at the time of failure. You could capture a process's state at any time by generating a user dump file. User dumps are generated by any Windows debugger and have the file extension .dmp, .hdmp, or .mdmp. The main Windows debuggers for processes are Windbg, Cdb, and ntsd, and their user dumps, when analyzed, can contain valuable clues about what caused a process crash. Accurately analyzing a process dump file can require some expertise. That's where DebugDiag comes in: It makes the analysis portion of the troubleshooting process much simpler.

DebugDiag combines many key features from each of the Windows Debugging Tools (ADPlus, Userdump, and WinDbg) and includes a rich UI, which helps make the tool easy to use.

DebugDiag is installed as a service, so configuration settings that you set in DebugDiag will survive system reboots. The tool's analysis feature is fast, easy to use, and portable, so you can send the data to a manufacturer or in-house developer for further review and troubleshooting. DebugDiag requires less than 19MB of disk space. It runs on Windows Vista/XP/2000/NT and Windows Server 2003 but hasn't been tested on Windows Server 2008.

DebugDiag in Action

Let's look at how the Microsoft Global Escalation Services team used DebugDiag to handle a recent customer issue. The customer's website kept going down, and we suspected that the Microsoft World Wide Web server process might be crashing. So we installed DebugDiag and configured it to monitor specifically for crashes in the World Wide Web Publishing Service.

Simplify Process Troubleshooting with DebugDiag

Michael Morales

(Reprinted From WindowsItPro Magazine)

After you install and start DebugDiag, you're immediately presented with the Select Rule Type wizard dialog box, which lets you choose the appropriate rule to use, depending on what you want to monitor. In this example, we'll concentrate on process crashes, so if you suspect or have confirmed that a process crash is occurring, you should select the Crash rule type in the Select Rule Type dialog box, then click Next.

Now you'll choose the type of process to monitor in the Select Target Type dialog box, such as a specific NT service, a specific process (e.g., an application process), or all IIS/COM+ related processes. For our customer support problem, we chose to monitor a specific service and selected the World Wide Web Publishing Service in the Select Target dialog box.

In the wizard's next dialog box, Advanced Configuration (Optional), you can configure optional advanced settings for crash monitoring. In our case, we simply chose the defaults and clicked Next. You'll then see a dialog box showing the name of the rule and the path in which the user dump data will be stored; click Next to keep the defaults or make changes, such as changing the default directory where dump files are stored.

You'll see the final dialog box, where you can either activate the rule now or manually activate it later. Then click Finish. Note that you might want to choose the activate later option if you aren't ready to monitor a process just then but want to complete the configuration steps ahead of time.

Now you'll see the main DebugDiag application window, which has three tabs. Click the Rules tab to see the configured rules on that system, the rule name, the rule's status (active or not), and Userdump Count. Userdump Count is the number of process crashes for the monitored process that DebugDiag captured and stored in the path listed under the Userdump Path column. The Processes Tab displays the currently running processes on the system.

Analyzing the Data

After you've configured DebugDiag to monitor for a specific process, you can reboot the system and log off without worrying about disturbing the monitoring process. When you suspect the monitored process has crashed, you can check the DebugDiag application window and view the Userdump Count column to verify that a user dump file has been created.

The Advanced Analysis tab, which Figure 1 shows, is where you select which script you want to run to analyze the user dump data for a monitored process. We chose the Crash/Hang Analyzers script since we want to analyze a process crash. Next, you'll need to add a user dump file to analyze, by clicking the Add Data Files button and navigating to the stored location of the captured user dumps. Highlight the appropriate .dmp file and click the Open button. You'll see that the dump file has been added; you're now ready to start the analysis.

Simplify Process Troubleshooting with DebugDiag

Michael Morales

(Reprinted From WindowsItPro Magazine)

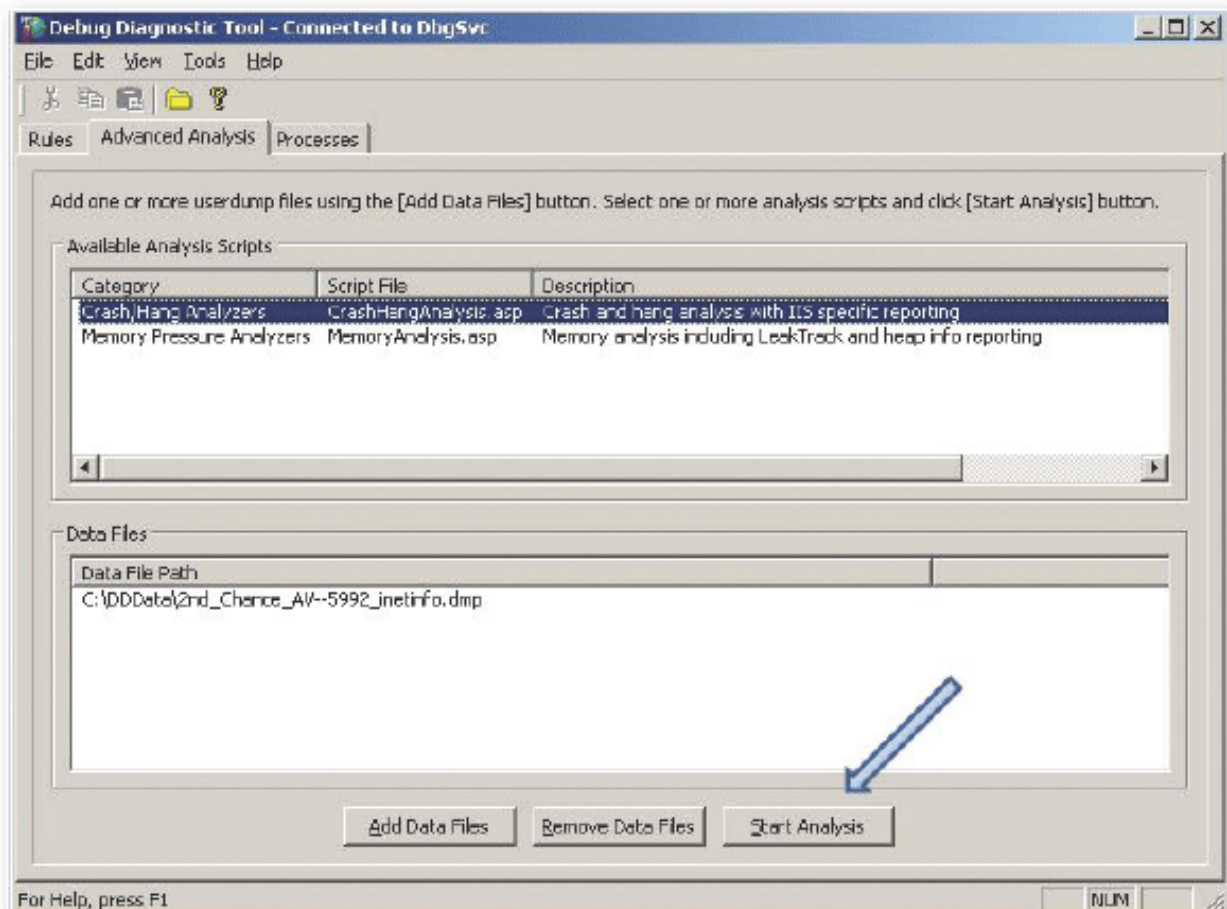


Figure 1: Selecting a script to analyze user dump data

Click the Start Analysis button to execute the script you selected. DebugDiag will show the analysis progress. When the analysis is finished, DebugDiag automatically saves the analysis report in the DebugDiag\Reports folder and opens it in Internet Explorer. An analysis report has three main sections:

- Analysis summary—an Event Viewer– type of message that records errors, warnings, and information relevant to the user dump analysis along with descriptions and recommendations for solving the problem shown by the error and warning information.
- Analysis details—starts with a table of contents listing all the analyzed memory dumps. For each memory dump, there's a listing of report titles indicating the type of analysis performed.
- Script summary—reports the status of the script that was run to analyze the user dump. If any errors occurred while the script ran, this section will list the error code, source, description, and lines that caused the errors.

Simplify Process Troubleshooting with DebugDiag

Michael Morales

(Reprinted From WindowsItPro Magazine)

For the World Wide Web Publishing Service crash, we found the resolution in the analysis summary's Recommendation section, which provided a link to a Microsoft article that contained the fix for the problem, as Figure 2 shows.

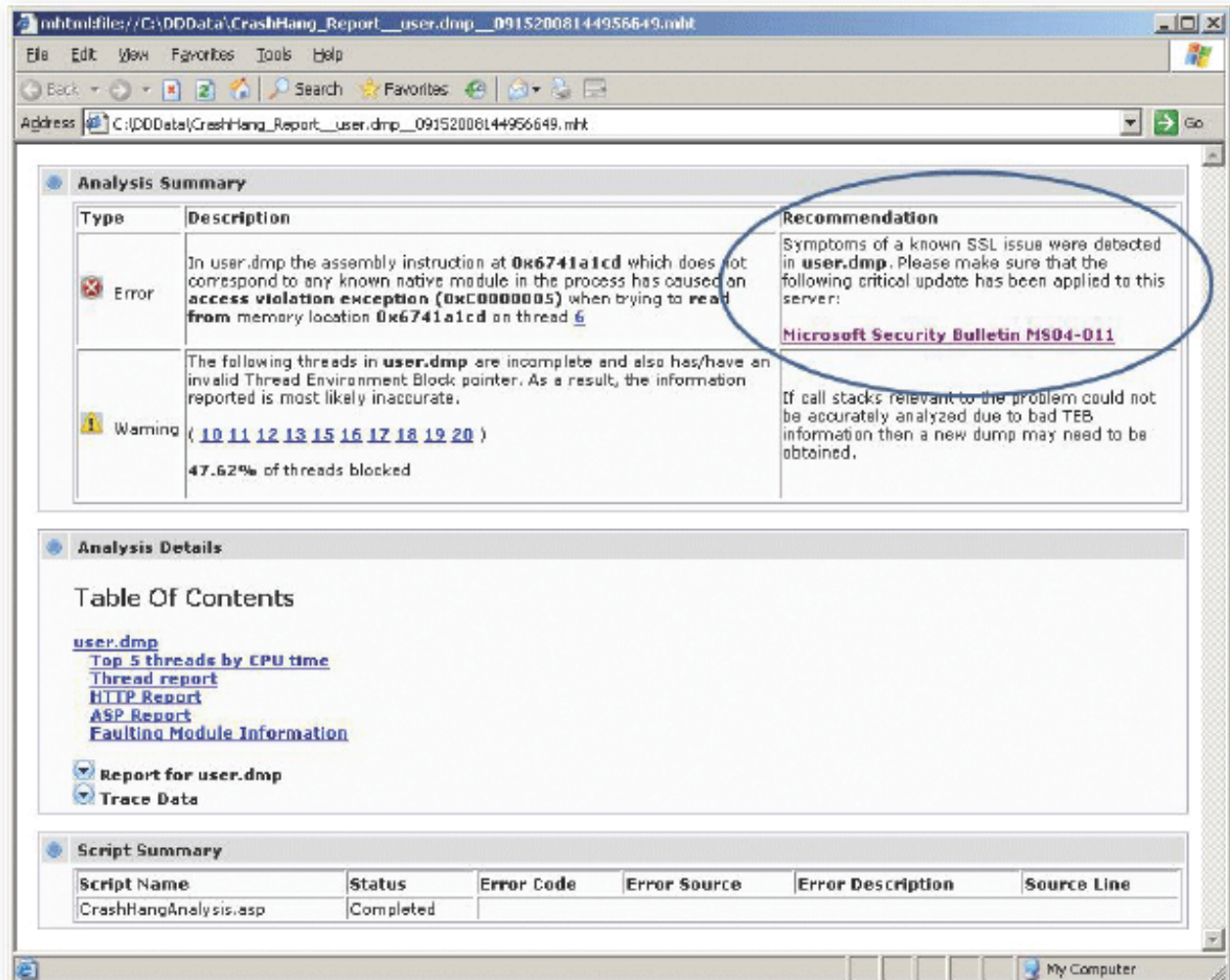


Figure 2: Analysis report recommendations

Closing in on a Solution

Although DebugDiag probably won't resolve every Windows process problem, it will usually provide data to move you closer to a solution. Sometimes you might get only the .dll name and manufacturer that caused the problem, but with such data you can search online for a solution or help your tech support person more quickly resolve the problem.