Memory and DLL Management in IIS 5

IIS 5 built on many of the principles in IIS 4 for the management of applications and the memory and extensions they used. The system was designed long before it became popular for sites to be largely—or often entirely—dynamically driven, and the limitations of the old system reflect that particular Web model.

Applications within your IIS installation are based on the DLL used to execute the application and the code or files that make up the page or application. For example, .asp pages are processed by the ASP ISAPI DLL.

Applications within IIS are configured on a site by site or directory by directory level from within the main directory Web site properties. When you define an application as part of the Web site, all files and directories are considered part of the application; when just a directory is configured, only the files and subdirectories within it are considered part of the application.

Applications are also given a *namespace* that defines a logical region in memory used to hold the application and any variables and information used by it during execution. These namespaces can be assigned to work with different protection settings.

The idea of the protection is to prevent the application from affecting the other applications and systems if there is some kind of problem with the application. For example, a bug in an application could cause it to use inordinate amounts of memory or the entire application to crash, which could easily wipe your Web site or at least make it unavailable for a significant period of time.

Three different protection levels were available within IIS, and each application could be assigned to use one of these levels. Different levels provided a rising scale of security and isolation from Low to High. The specifics of the three levels are

- **Low Protection**—Enables applications to run within the same process as the IIS system and share resources. This offers the highest performance, but the lowest possible security, making it possible for a faulty application to bring the entire IIS system down in the event of a failure.
- **Medium Protection**—Enables applications to run within a pooled memory area. Although this is separate from the main IIS process, applications running within the pooled area are all subject to the same risks. If one application within the pool causes the pool to fail, all the applications within the pool fail.
- **High Protection**—Allocates a separate memory process for each application. Should an application fail, the only process it will affect is its own—other applications operating in any of the other protection modes will be unaffected.

Both the medium and high protection modes also allow you to set scheduled application restarts and the ability to kill and re-create applications that are causing serious problems.

These protection modes also govern the memory availability to different applications and obviously as the number of pooled and individual applications increase, so does the overall memory requirement needed to serve the sites. Each application pool or isolated application requires the invocation of another instance of DLLHOST.EXE, which has a memory footprint of about 5MB. This can increase the overall memory requirements significantly from hosting the application with low protection within the IIS process.