Steve Stedman
Freelance SQL Server Consultant
http://stevestedman.com

Silencing Backup Messages with Trace Flag 3226

As a DBA you regularly schedule backups you might have regular full backups, incremental backups and hopefully regular backups. Over time you might realize that your error log is not longer filled with errors, but instead a majority of your error log is filled with backup messages. "Backup Database successfully ... ", "Database backed up...", "Log was backed up". This makes it a bit difficult to find the actual errors in this flood of backup messages. Your error log is no longer an error log, but it looks more like a backup log.

Trace Flag 3226

Trace flags in SQL Server are switches or parameters that can be used to turn on or off different features. If you use trace flag 3226 this will turn off all successful backup messages being written to the error log. There are a couple of ways to use trace flag 3226, one is to use DBCC TRACEON to turn on trace flag 3226 for the currently running instance. Using this option will turn it on until the next time the database instance is restarted.

The second way is to add it as a start up parameter to the SQL Server process by using the SQL Server Configuration Manager.

Right click on the SQL Server instance that you want to configure and choose properties. Then click the advanced tab. There will be an option in the list called Startup Parameters. Add the ;-T3226 parameter (with the semicolon separating it from the previous parameter). Then click OK.

Changing the startup parameter won't take effect until the next time the instance is restarted. Also note that if you make a mistake here it may prevent the instance from starting. It is recommended to add the trace flag, then to restart the instance to confirm that it starts correctly.

Once this change has been made, successful backup messages will no longer be written to the error.

But what if I want a history of successful backups?

Every good DBA should want some type of history showing the successful backups so that in the event that a restore is required you can look at the history to find out what backup you need.

There is another history of backups stored in the msdb.dbo.backupset table, and you can access it with the following query.

Another way to get at the backup history is with the Database Health Monitor backup report that looks like this.

At this point you can see that the history of backups in the error log redundant to the backups stored in the msdb.dbo.backupset table, and that you can get a better understanding of your errors in the error log if you can see the errors rather than a flood of backup messages. How would it be if you missed the following error message indicating that you have database corruption.

Get those backup messages out of your error logs with trace flag 3226 and use your error logs for errors.

Related Links

- Database Health Monitor
- Database Corruption Repair