WSUS cleanup aborting: Increase timeout for database and IIS

https://4sysops.com/archives/wsus-cleanup-aborting-increase-timeout-for-database-and-iis/

If many expired or superseded updates have accumulated in WSUS, then the server cleanup wizard will hang. The problem is often difficult to solve, but promising measures include a longer timeout for the IIS and the database, and more memory for the IIS AppPool.

Contents

The problem is well known to most WSUS admins: the animated progress bar in the server cleanup wizard is stuck in the same position for a long time, and at some point, the message appears that a database error has occurred. The console then offers to reset the server node.

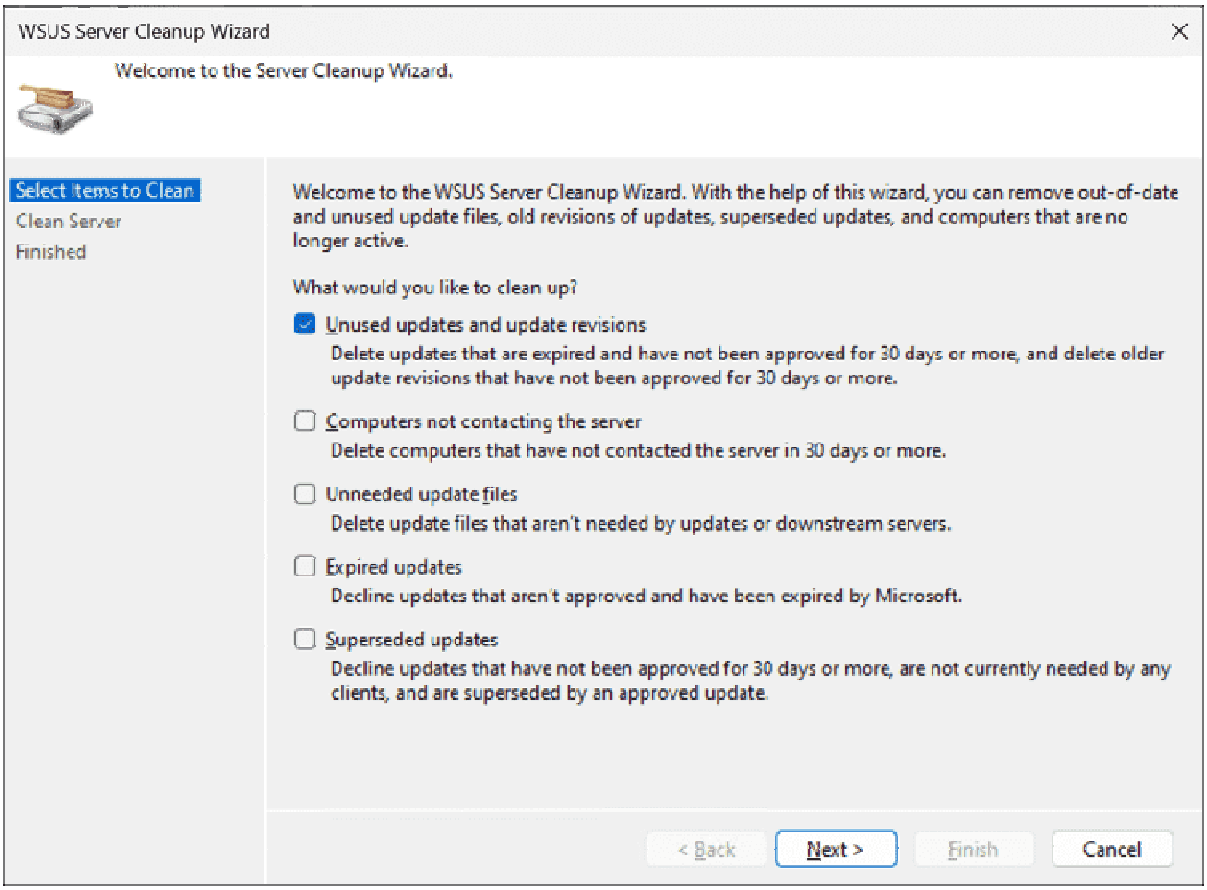
If you use PowerShell for the cleanup instead, the following message will appear in this situation:

Invoke-WsusServerCleanup: Execution Timeout Expired. The timeout period elapsed prior to completion of the operation or the server is not responding. The statement has been terminated.

There are many forums that discuss this issue. Like various blog posts, they ultimately boil down to a handful of tips that can help.

Perform cleanup tasks separately

First, it is worth trying not to run all cleanup tasks at once but to run them separately.



You can start the cleanup jobs one by one to avoid a timeout

In PowerShell, you call Invoke-WsusServerCleanup with just one parameter, which comprises one of the following:

- CleanupObsoleteComputers
- CleanupObsoleteUpdates
- CleanupUnneededContentFiles
- DeclineExpiredUpdates
- DeclineSupersededUpdates

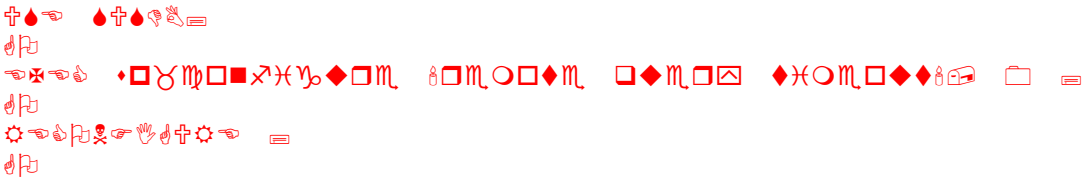
In most cases *CleanupObsoleteUpdates* will be the culprit if the operation fails.

Increase database timeout

Most WSUS installations use the internal Windows database (WID). In this case, you should run the maintenance script for susdb regularly anyway. If this has not happened for a while, it is recommended to do it now (see instructions).

Usually, not even this will help to successfully clean up WSUS. Therefore, the next step should be to increase the timeout for the database. The default value is 600 seconds. If you set this value to 0, you will remove the timeout completely.

To do so, execute the following T-SQL script with the help of sqlcmd.exe, which you can save in a file named *timeout.sql*:



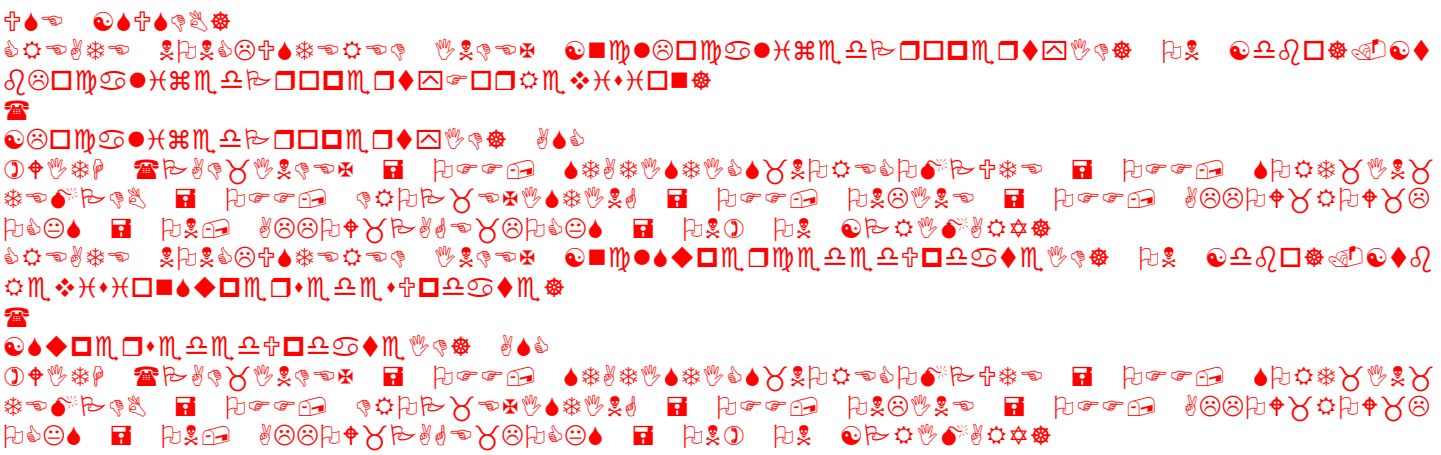
The command for it looks like this:

In this example eliminating the Windows database timeout resulted in a successful cleanup

The command line variant shown here has the advantage that it also runs under Server Core.

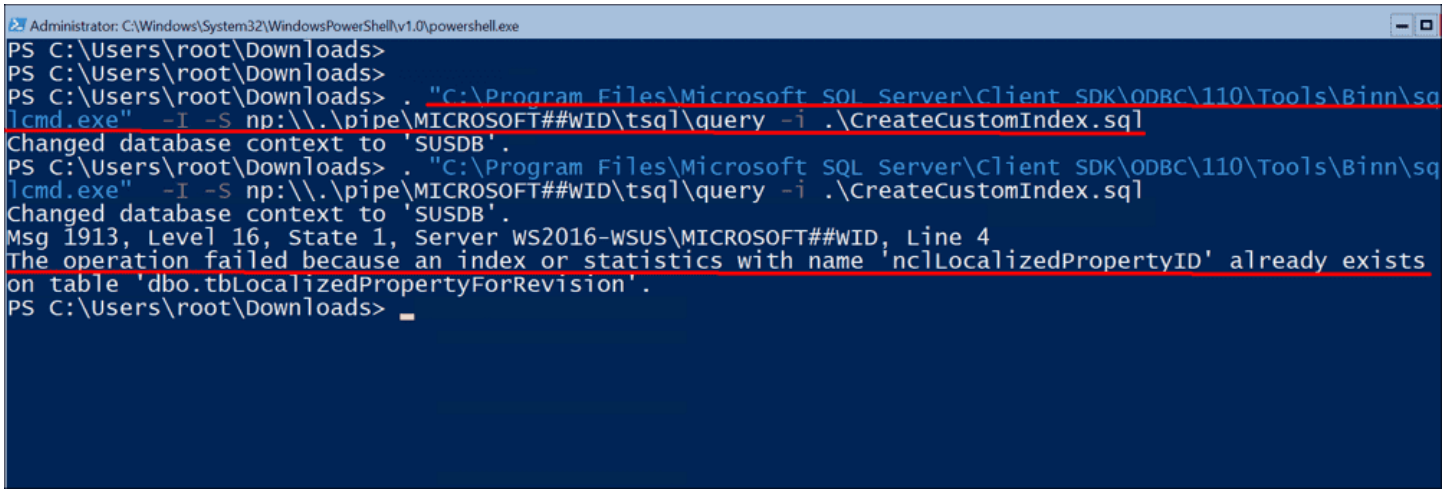
Create an index for susdb

Another measure recommended by Microsoft is to create a custom index to speed up database operations. For this purpose, save the following script in a file:



Execute this script in the same way as the one above to increase the timeout:

If the index already exists, the script will return an error.



A custom index speeds up database operations if it already exists the script will show an error message

Increase timeout for IIS-AppPool

Another measure is to adjust the IIS configuration. Here, it is also possible to increase the timeout or to eliminate it completely with a value of 0. This can be done via the graphical IIS Manager, but the procedure below using PowerShell also works under Server Core.

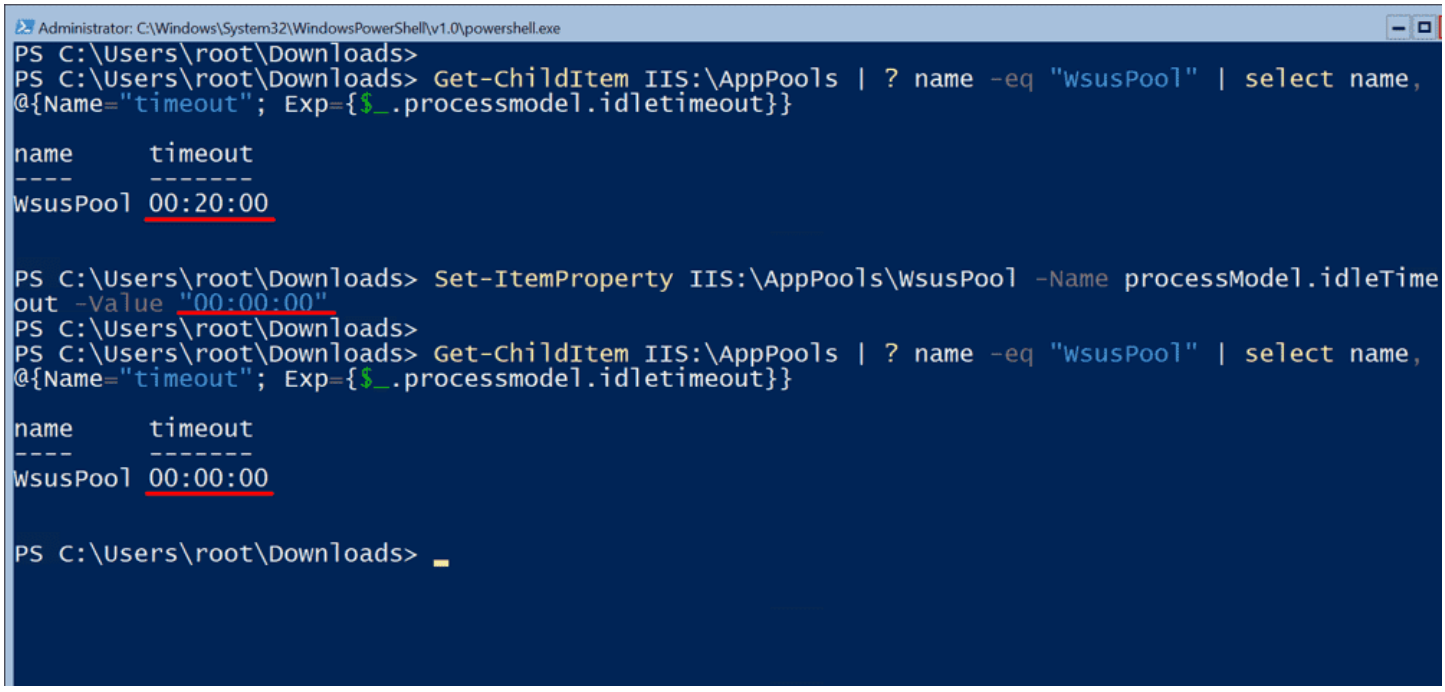
First, make sure that the PowerShell module for IIS is installed on the WSUS server. This can be done using the following:

If it is not present, then add it with

and import it using

Subsequently, PSDrive IIS:\ is available. This can be used to determine the current setting for the timeout of the WSUS AppPool:

Now, increase this from the default 00:20:00 to the desired value, or set it to 0 to override it:



Set the timeout for the WSUS AppPool to 00 00 00 to deactivate it

Remove the private memory limit

As a best practice, Microsoft recommends setting the private memory limit for the WSUS AppPool to 0 as well, thereby overriding it. You can get the current value like this:

If necessary, change the value to 0 using this command:

Summary

If you neglect regular WSUS cleanup and database maintenance, then at some point deleting unneeded updates will fail. This problem is widespread and often quite persistent.

Eliminating the timeout for the database and the IIS AppPool for WSUS and increasing the limit for its private memory are measures that often help to successfully clean up WSUS.