

Log File Cleanup

10/21/2025 1:51 pm EDT

-- Check log and database sizes might be culprit.

-- Run log file cleanup script, ensure it is in the correct database and you are shrinking the correct log.

-- You may need to run multiple times to get size down. If something doesn't run in SQL, there is likely an issue on the SQL server itself such as resources, drive failing, permissions etc.

-- RENAME CHANGE ME to Database

-- Also ensure 'Security_log' to match db logical name right click properties > Files > Logical name xxx_log

```
BACKUP LOG [CHANGE_ME]
```

```
TO DISK = 'nul:' WITH STATS = 10
```

```
GO
```

```
dbcc shrinkfile ('Security_log', 0, TRUNCATEONLY)
```

```
GO
```

-- There are few tips involved to avoid from shrinking, which are,

-- 1) Ensure to run the log backup frequently and also manage the log file size as small.

-- 2) Try to run differential backup in simple recovery model, if the organization is ready to accept some data loss. This will manage the log file to maintain the size limit.

-- 3) Try to alter the database and move the log file by adding a new drive or transfer it to another drive.

-- Additional Context Transaction Log Truncation [https://learn.microsoft.com/en-us/previous-versions/sql/sql-server-2008-r2/ms189085\(v=sql.105\)](https://learn.microsoft.com/en-us/previous-versions/sql/sql-server-2008-r2/ms189085(v=sql.105))

-- log_reuse_wait and log_reuse_wait_desc

-- If log records were never deleted from the transaction log, it would eventually fill all the disk space that is available to the physical log files.

-- Log truncation automatically frees space in the logical log for reuse by the transaction log.

-- Except when delayed for some reason, log truncation occurs automatically as follows:

-- Under the simple recovery model, after a checkpoint.

-- Under the full recovery model or bulk-logged recovery model, after a log backup, if a checkpoint has occurred since the previous backup. For more information, see "Log truncation under the full and bulk-logged recovery models," later in this topic.

-- What is the Syntax to Truncate Log File in SQL Server?

-- Generally, the log file is truncated to keep the log file from filling up. Here, the transaction log file is divided into smaller chunks called as Virtual log files(VLF). Truncating the log file means deleting the actual log file. In case, the log file is not truncated on a regular basis, then all the space will be occupied that is been allocated to the physical log file. When the truncate command is implemented, it will overwrite the log file, instead it deletes the log file from the disk. Truncate process will change one or more Virtual log file status from active to inactive by which the storage space is marked for reuse. The status of VLF can be changed automatically from active to inactive state. However it will depend completely on the recovery model and backup type. The SQL server runs a checkpoint. If it occurs in simple, full or bulk recovery model, then a user is allowed to execute the truncate process. In which, the status of the VLF file is changed from active to inactive as now a user can also reuse it automatically. We can use Truncate_Only or With No_Log command to truncate the log file in SQL server.