

What is a TUF File in SQL Server Log Shipping?

TUF stands for **Transaction Undo File** — a special file created **only when log shipping is configured in STANDBY mode**. In STANDBY mode, the secondary database remains **read-only**, allowing users to run queries between log restore cycles.

Purpose & Uses of the TUF File

1 Manages Unfinished (Active) Transactions

When a log backup is restored, there may be transactions that were active and not fully committed.

- The **TUF file stores information about these incomplete transactions**.
- SQL Server uses it to **roll back** these transactions so the secondary database stays consistent and readable.

2 Enables Standby/Read-Only Mode

The secondary can be opened for reporting or read-only queries.

- Without the TUF file, SQL Server **cannot** put the database in standby mode.
- It ensures the database is readable between log restores.

3 Supports Reapplying Rolled-Back Changes

During the next log restore:

- SQL Server checks the TUF file to **reapply (redo)** the previously rolled-back changes,
- then apply the next log backup in sequence.

4 Preserves Database Consistency Across Cycles

It ensures no partial transactions corrupt the secondary database during log shipping operations.

5 Used Only in Standby Mode (NOT in NoRecovery Mode)

- **Standby** = requires TUF file
- **NoRecovery** = TUF file is **NOT** created

Why the TUF File is Important

- ✓ **Prevents data corruption** during log restores
- ✓ **Maintains a consistent read-only secondary database**
- ✓ **Enables real-time reporting** without breaking log shipping
- ✓ **Helps SQL Server track undone and redo-required changes**
- ✓ **Ensures smooth and fast recovery during each restore cycle**
- ✓ **Critical for log shipping jobs to run successfully in standby mode**

Where is the TUF File Located?

The TUF file is created and stored **on the Secondary Server**, in the same directory where log backups are restored.

Typical location example:

D:\LogShipping\Standby\

The exact location depends on your log shipping configuration:

- The **Restore Settings → Standby File** path
- Folder permissions
- SQL Server Agent account access

Additional Points Related to Log Shipping & TUF Files

1. If the TUF File is Deleted or Corrupted

- The secondary database will **fail** subsequent log restores.
- You must **reinitialize** log shipping (full backup + restore).

2. TUF File Is Exclusive to Log Shipping

- Mirroring / AlwaysOn / Replication **do NOT use TUF files** — only log shipping with standby mode does.

3. TUF File Is Not Automatically Cleaned

Even after a restore job finishes, the TUF file remains until the next log restore requires it.

4. TUF File Extends with Large Transactions

If you have long-running transactions on the primary, the TUF file may grow significantly.

5. TUF File Name Pattern

SQL Server generates a name like:

dbname_undo_XXXXXX.tuf

In Simple Words

The **TUF file is SQL Server's "bookmark"** for incomplete transactions during log shipping.

It helps SQL Server undo uncommitted work, keep the secondary readable, and then redo the changes when the next log backup arrives. Without it, **standby mode log shipping cannot function**.

✅ Option 1: TUF File & Log Shipping Diagram

A clean diagram showing:

- Primary DB
- Log backup generation
- Copy + restore jobs
- Role of TUF file (Undo / Redo)
- Standby vs NoRecovery timeline

✅ Option 2: SQL Server Log Shipping – Interview Q&A (20–30 Questions)

Includes:

- TUF file questions
- Real-time log shipping scenarios
- Troubleshooting questions
- Architecture questions
- Difference between Log Shipping vs Mirroring vs AlwaysOn
- Performance & latency questions

✅ Option 3: Full Log Shipping Setup Notes (Step-by-Step)

Includes:

- Pre-setup checklist
- Required SQL Agent jobs explained
- Monitor server
- Folders, permissions, ports
- Standby vs NoRecovery mode behavior
- Common pitfalls

Option 4: Common Log Shipping Errors & Fixes

Including:

- “TUF file missing or corrupt”
- Error 14421
- Cannot restore because database is in use
- Log chain break
- Large TUF file growth
- Restore job failures

<https://www.sqldbachamps.com/>

Four options ready for study, documentation, or interview prep.

✅ **OPTION 1: LOG SHIPPING & TUF FILE DIAGRAM (VISUAL EXPLANATION)**

Below is a clear conceptual diagram (text-based, but structured like a flowchart):

PRIMARY SERVER (PRD)

Database: DB1

|
| 1. Log Backup Job (LSBackup_Job)
| Creates *.trn files
v

Log Backup Folder

Ex: \\BackupShare\DB1\LS\

|
| 2. Copy Job (LSCopy_Job)
v

SECONDARY SERVER (DR/REPORTING)

Restore Folder

Ex: D:\LogShipping\Restores\

|
| 3. Restore Job (LSRestore_Job)
| - Restores TRN files sequentially
| - If in STANDBY mode → creates TUF file
v

SECONDARY DATABASE (STANDBY MODE)	
- Read-Only	
- Users can run SELECT queries	
- Uses TUF file during every restore	

|
| TUF File = Undo File
|
| How it works:
| - Rolls back incomplete transactions
| so database becomes readable
| - On next restore:
| SQL Server redoes earlier undone
| transactions using the TUF file
v

TUF File Location (Example):

D:\LogShipping\Standby\DB1_undo_12345.tuf

✅ OPTION 2: INTERVIEW Q&A (30+ SQL LOG SHIPPING QUESTIONS)

Basic Questions

1. What is log shipping?
2. What are the components of log shipping?
3. What SQL Agent jobs are created automatically?
4. Difference between Standby mode vs NoRecovery mode?
5. What is a TUF file? Why is it created?
6. When is a TUF file NOT created?

Intermediate Questions

7. What happens if a transaction log backup is missing?
8. What happens if the secondary server is offline?
9. Can you take a full backup on the primary without affecting log shipping?
10. How do you reinitialize log shipping?
11. What is log chain and how is it maintained?
12. How often should log backups occur in log shipping?
13. Is log shipping synchronous or asynchronous?
14. What are common reasons for log shipping restoring delays?
15. What permissions are required for log shipping copy job?
16. How to check last log backup applied on secondary?

Advanced/Senior DBA Questions

17. What causes a TUF file to grow excessively?
18. How to resolve "TUF file is missing" error?
19. How to force restore when secondary DB is stuck in restoring state?
20. How to change log shipping frequency?
21. How to configure log shipping across domains?
22. Difference between log shipping vs mirroring vs AlwaysOn?
23. How to configure log shipping for very large databases (VLDBs)?
24. Can log shipping be automated with PowerShell?
25. How to handle orphaned users in a log-shipped secondary DB?
26. How does monitor server work in log shipping?
27. What is alert threshold in monitor server?

Scenario-Based Questions

28. Log shipping breaks due to a missing .trn file. What do you do?
29. Restore job fails because database is in use. How to fix?
30. Secondary is 2 hours behind — how do you troubleshoot?
31. You see "log sequence number mismatch" during restore — cause?

✅ OPTION 3: FULL LOG SHIPPING SETUP NOTES (STEP-BY-STEP)

Step 1: Pre-Setup Checklist

- Both servers must have SQL Server installed.
- SQL Agent must be running.
- Primary DB must be in **FULL recovery model**.
- Shared folder accessible by SQL Agent account.
- Sufficient disk space for logs.
- Proper network bandwidth for log shipping interval.

Step 2: Configure Log Shipping on Primary

1. Right-click **Primary DB** → **Properties** → **Transaction Log Shipping**
2. Enable log shipping
3. Configure backup job:
 - Set backup folder
 - Set frequency (e.g., every 2 minutes)

Step 3: Configure Secondary Server

1. Provide full backup + restore path
2. Choose restore mode:
 - **NoRecovery** → warm standby (cannot read)
 - **Standby** → read-only (creates TUF file!)
3. Configure copy job folder
4. Set restore delay or restore immediately

Step 4: Configure Monitor Server (Optional but Recommended)

- Tracks latency
- Alerts if log shipping falls behind
- Stores metadata in msdb tables

Step 5: Validate Log Shipping

- Check SQL Agent jobs on both servers
- Verify .trn files being generated/copy/restored
- Ensure standby DB is accessible
- Review monitor server status

Key SQL Agent Jobs

Server	Job Name	Purpose
Primary	LSBackup_Job	Generates .trn backups
Secondary	LSCopy_Job	Copies .trn files
Secondary	LSRestore_Job	Restores .trn files
Monitor Server	LSAlert_Job	Tracks health/status

Best Practices

- Keep log shipping interval at **1–5 minutes**.
- Use separate drive for restore folder.
- Enable compression for large DBs.
- Monitor long-running transactions (affects TUF size).
- Use “disconnect users when restoring” on standby mode.

✅ OPTION 4: COMMON LOG SHIPPING ERRORS & FIXES

1. TUF File Missing or Corrupted

Error:

Could not find file <dbname_undo_XXXXX.tuf>

Fix:

- Restore latest FULL + DIFF + all logs
- Or reinitialize log shipping completely

2. "Database is in use" during restore

Cause:

Users querying standby DB

Fix:

Enable:

Disconnect users in the database when restoring backups

3. Log Chain Broken

Common causes:

- Someone took manual log backup
- Log backup deleted
- Full backup WITH INIT

Fix:

Reinitialize log shipping.

4. Log Shipping Latency

Causes:

- Slow network
- Long-running transaction on primary
- TUF file oversized
- Heavy server load

Fix:

- Increase backup frequency
- Compress backups
- Fix network bottlenecks

5. TUF File Growing Too Large

Cause:

- Long open transactions on primary

Fix:

- Identify open/long transactions using:
- DBCC OPENTRAN

6. Log Sequence Number (LSN) Mismatch

Cause:

- Logs applied out of order

- Missing log file
Fix:
- Reapply correct backup chain
- Reinitialize if sequence cannot be corrected

7. Restore Job Failing Due to Permissions

Fix:

- Ensure SQL Agent account has read/write share permissions
- Validate UNC path accessibility

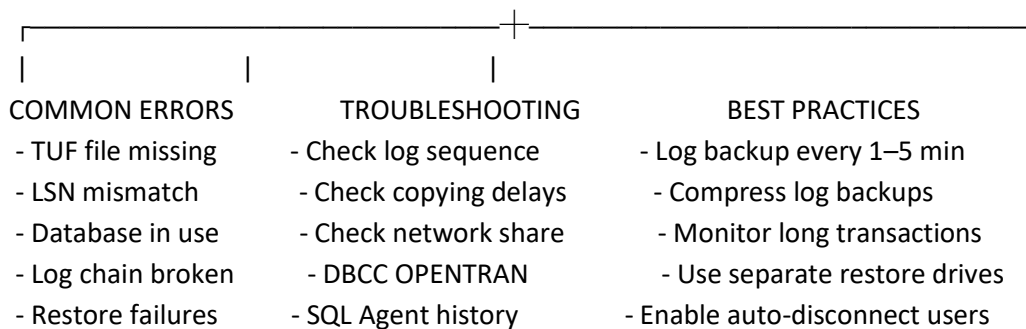
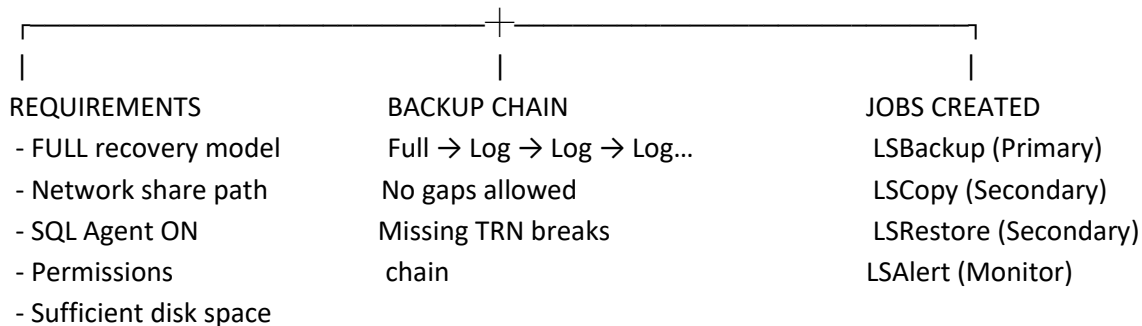
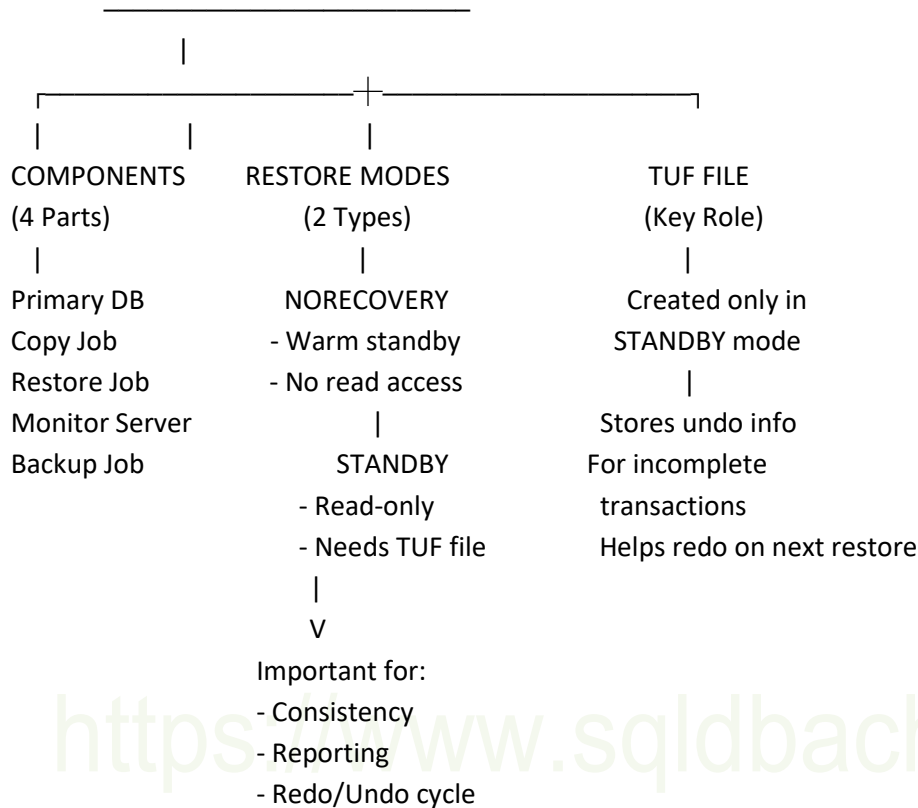
<https://www.sqldbachamps.com/>

Mind Map for Interview Revision and a Full Hands-on Lab Exercise for SQL Server Log Shipping.

1. SQL Server Log Shipping — Mind Map (Interview Revision)

(A text-based mind map you can convert to a diagram or PPT as needed.)

SQL SERVER LOG SHIPPING



COMPARISONS	MONITOR SERVER	SCENARIO QUESTIONS
Log Shipping vs	- Stores metadata	- What if a log file missing?
Mirroring vs AG	- Alerts & health	- How to reduce delay?
Key differences	- Latency tracking	- How to reinitialize LS?

This mind map covers everything interviewers typically ask.

2. Full Hands-On Lab Exercise for SQL Server Log Shipping

(End-to-end practical guide you can execute in your lab environment.)

LAB SETUP OVERVIEW

You will configure and test log shipping between:

- **Primary Server:** SQL01
- **Secondary Server:** SQL02
- **Database:** SalesDB
- **Share Path:** \\SQL01\LSBackup
- **Restore Mode:** STANDBY (Read-only with TUF file)

STEP 1: Pre-Checks (Mandatory)

Run these on the **PRIMARY** server:

1. Check database recovery model

```
SELECT name, recovery_model_desc
FROM sys.databases
WHERE name = 'SalesDB';
```

If not **FULL**:

```
ALTER DATABASE SalesDB SET RECOVERY FULL;
```

2. Take a Full Backup

```
BACKUP DATABASE SalesDB
TO DISK = '\\SQL01\LSBackup\SalesDB_FULL.bak'
WITH INIT, COMPRESSION;
```

STEP 2: Create & Configure the Share Folder

On **SQL01** (Primary):

1. Create folder: C:\LSBackup
2. Share as: \\SQL01\LSBackup
3. Grant permissions to:
 - SQL Server Agent account on SQL01
 - SQL Server Agent account on SQL02

STEP 3: Configure Log Shipping on Primary Server

1. Right-click SalesDB → **Properties**
2. Click **Transaction Log Shipping**
3. Check: **Enable this as primary**
4. Configure Backup Settings:
 - Path: \\SQL01\LSBackup
 - Backup retention: 72 hours

- Frequency: Every **2 minutes**
 - Compression: Enabled
5. Click **Add Secondary**
 6. Enter secondary server: **SQL02**

STEP 4: Configure Secondary Server (SQL02)

INITIALIZE SECONDARY

Choose **“Restore the full backup and apply all log backups”**

RESTORE MODE

Select **STANDBY** mode:

- Path for undo file (TUF file):
- D:\LogShipping\Standby\SalesDB_undo.tuf

COPY SETTINGS

- Destination folder:
- D:\LogShipping\IncomingLogs\

RESTORE SETTINGS

- Restore frequency: 1 minute
- Delay: 0
- Alert if restore behind by: 30 minutes

STEP 5: Validate Log Shipping Jobs

On SQL01:

- LSBackup_SalesDB

On SQL02:

- LSCopy_SQL01_SalesDB
- LSRestore_SQL01_SalesDB

Run Jobs Manually

Run in order:

1. Backup job
2. Copy job
3. Restore job

Confirm:

SalesDB on SQL02 is in **STANDBY/READ-ONLY** state.

STEP 6: Verify Log Shipping Functionality

Insert data on PRIMARY

```
INSERT INTO SalesDB.dbo.Orders (OrderDate, Amount)
VALUES (GETDATE(), 2500);
```

After restore job runs, check SECONDARY

```
SELECT TOP 10 *
FROM SalesDB.dbo.Orders
ORDER BY OrderID DESC;
```

You should see the newly inserted row.

STEP 7: Test TUF File Behavior (Important)

On Secondary, check TUF file exists:

Path:

D:\LogShipping\Standby\SalesDB_undo.tuf

Test long-running transaction:

On PRIMARY:

BEGIN TRAN

UPDATE Orders SET Amount = Amount + 10 WHERE OrderID = 1;

-- Do NOT commit

Now run backup/copy/restore jobs.

On SECONDARY:

- The DB will be readable.
- The TUF file will **grow** as SQL rolls back the open transaction.

Commit the transaction:

COMMIT TRAN;

Next restore cycle → SQL will **redo** changes using the TUF file.

STEP 8: Break Log Shipping and Repair It

1. Delete a .trn file from share folder
2. Run restore job → It fails
3. Note the error
4. Fix: Reinitialize log shipping
 - Take fresh FULL backup
 - Restore with NORECOVERY
 - Reconfigure

This is typically asked in interviews.

BONUS TASKS (Optional but Valuable)

Task A: Configure Monitor Server

- Use SQL03 as monitor
- Review info in msdb.dbo.log_shipping_monitor_history_detail

Task B: Automate Log Shipping Status Report

Create a stored procedure:

EXEC master.dbo.sp_help_log_shipping_monitor;

Task C: Test Failover (Manual)

1. Stop log shipping jobs
2. Bring secondary online (RESTORE WITH RECOVERY)
3. Redirect application connection

RESULT OF LAB

After completing this lab, you will understand:

- ✓ Full setup
- ✓ How TUF file works
- ✓ Restore cycles
- ✓ Log chain
- ✓ Monitoring & alerts
- ✓ Troubleshooting
- ✓ Real-world scenarios