MySQL 8.4 Upgrade:

Proven Best Practices For Zero-Surprise Production Migration

Alok Pathak
Principal DBA
Percona

Abhinav Gupta
Senior Database Reliability Engineer
Percona

PERCONA

Alok Pathak

Principal DBA, ExpertOps – Percona

Professional Background

16+ years managing MySQL at Scale

Percona, Google, Icreon, India Today

Focus Area

Performance, HA, Observability, Troubleshooting

Delhi, India

LinkedIn

lnkd.in/in/alokp16/

PERCONA

Abhinav Gupta

Senior Database Reliability Engineer, Expert Support, Percona

Professional Background

15+ years as MySQL DBA

Percona, Pythian, Netmagic Solutions(NTT), ACG

Certifications

Kubernetes, MySQL, AWS

Raipur, India

LinkedIn

lnkd.in/in/abhinavpgupta/

PERCONA

Agenda

PERCONA

Why Upgrade to MySQL 8.4?

PERCONA

Why Upgrade to MySQL 8.4?

MySQL 8.0 EOL
MySQL 8.0 reached End-of-Life in April 2026.

LTS Release
Benefit from a stable Long-Term Support release.

Security & Fixes
Access improved security protocols and critical bug fixes.

Performance
Benefit from optimizer and performance improvements.

Lower operational risk with a modern, fully supported version

PERCONA

Supported Upgrade Paths

MySQL 8.4 Supported Upgrade Paths

In-Place Major Upgrade

MySQL 8.0 → 8.4

Recommended: upgrade to latest MySQL 8.0 release first.

Older Versions

5.6 → 5.7 → 8.0 → 8.4

Direct 5.6/5.7 → 8.4 is **NOT** supported.

Minor Version Upgrades

8.4.x → 8.4.y supported

Replication Compatibility

8.0 Primary → 8.4 Replica

Recommended rolling upgrade path.

Avoid newer source → older replica

PERCONA

Upgrade Readiness

PERCONA

Review Changes Before Upgrading

Official Documentation
Changes in MySQL 8.4

Features & Variables
Added, Removed, Deprecated [1]

System Tuning
Default Configuration Changes

Compatibility
Connector & Client Support

Percona Specific
Changes & Improvements

Plugins Migration
Upgrade from plugins to components

PERCONA®

Authentication Plugin Changes

- **mysql_native_password** - Deprecated and disabled by default in MySQL 8.4.
- **caching_sha2_password** - Default authentication plugin.

Upgrade existing users:

```
SELECT user, host, plugin FROM mysql.user WHERE
plugin='mysql_native_password';


ALTER USER 'username'@'hostname' IDENTIFIED WITH
caching_sha2_password BY 'password';
```

mysql_native_password support

```
[mysqld]
mysql_native_password=ON
```


PERCONA

Upgrade From Plugins to Components

MySQL 8.4 introduces a transition from legacy **plugins** to modern **components**.

⚠️ **Upgrades require planning to avoid downtime**

Examples:

`keyring_vault`

`audit_log`

`data_masking`

PERCONA®

Legacy Replication Syntax Removed

Removed Syntax	Replacement
CHANGE MASTER TO	CHANGE REPLICATION SOURCE TO
START SLAVE	START REPLICA
STOP SLAVE	STOP REPLICA
SHOW SLAVE STATUS	SHOW REPLICA STATUS
RESET SLAVE	RESET REPLICA
MASTER_*	SOURCE_*

PERCONA®

Important Default Changes

Variable	New Default	Operational Impact
innodb_adaptive_hash_index	OFF	May impact read-heavy workloads
innodb_change_buffering	none	Different write behavior
innodb_io_capacity	10000	Impacts flushing behavior
innodb_log_buffer_size	64MB	Better write throughput
innodb_numa_interleave	ON	NUMA-aware memory allocation
temptable_max_ram	Dynamic	Memory usage changes
innodb_doublewrite_pages	128	Improves doublewrite performance

Upgrade Compatibility Validation

PERCONA

Pre-Upgrade Compatibility Checklist

Configuration Checks

- Removed system variables
- New default values
- Invalid configuration options/values

Security & Authentication

- Deprecated authentication methods
- Removed privileges/plugins

Schema Checks

- CHECK TABLE … FOR UPGRADE
- Foreign key validation
- Invalid column definitions
- Orphaned objects

Operational Checks

- Replication compatibility
- Plugin compatibility
- Character set/collation review
- Reserved keywords conflicts
- Deprecated syntax usage

MySQL Shell Upgrade Checker

Purpose: Validates server readiness before upgrade.

Checks Include:

- Removed variables
- Deprecated syntax
- Invalid authentication methods
- Configuration incompatibilities
- Data dictionary issues and many more

Privileges: `RELOAD`, `PROCESS` and `SELECT`

Command Example:

```
mysqlsh -- util check-for-server-upgrade --target-version=8.4.8 --config-path=/etc/my.cnf
```

Recommendation: Resolve all errors before upgrade

PERCONA

MySQL Shell Upgrade Checker

Shell Utility: Sample Output

```
$ mysqlsh -- util check-for-server-upgrade --target-version=8.4.8 --config-path=/etc/my.cnf

1) System variable check for deprecation, removal, or invalid values (sysVars)

Error: The following variables will be removed. Please update before upgrade.

  • default_authentication_plugin:  removed at 8.4.0; currently mysql_native_password
  • expire_logs_days:  removed at 8.2.0; currently set to 7 (GLOBAL)

- innodb_adaptive_hash_index: default changed ON -> OFF
- innodb_change_buffering: default changed all -> none
- innodb_log_file_size: deprecated at 8.0.30; set to 5368709120 (GLOBAL)
```


PERCONA

MySQL Shell Upgrade Checker

Shell Utility: Sample Output

```
9) Checks for errors in column definitions (columnDefinition)
No issues found

10) Checks for user privileges that will be removed (invalidPrivileges)
Notice: The following users have the SET_USER_ID privilege which will be
removed as part of the upgrade process:
- 'user1'@'%': The user 'user1'@'%' has the following privileges

[ ... additional output truncated ... ]

Errors: 4
Warnings: 23
Notices: 12
```


PERCONA

Upgrade Testing

PERCONA

Are You Really Ready for Upgrade?

Questions to Answer

Functionality
Will all application queries still work?

Data Integrity
Will results remain identical?

Performance
Will latency increase?

Configuration
Are new defaults safe for production?

Connectivity
Are client drivers compatible?

PERCONA

Upgrade Testing Strategy

Goal: Safely validate MySQL 8.4 before production rollout.

Recommended Approach

1. **Clone** production environment
2. **Upgrade** test replica to 8.4
3. **Replay** production workload
4. **Compare** results and latency
5. **Fix** regressions before rollout

PERCONA

pt-upgrade Overview

What it Does

Replays production queries and compares results across versions.

Detects:

- Query regressions
- Result differences
- Errors/warnings

Environment

Recommended setup for testing:

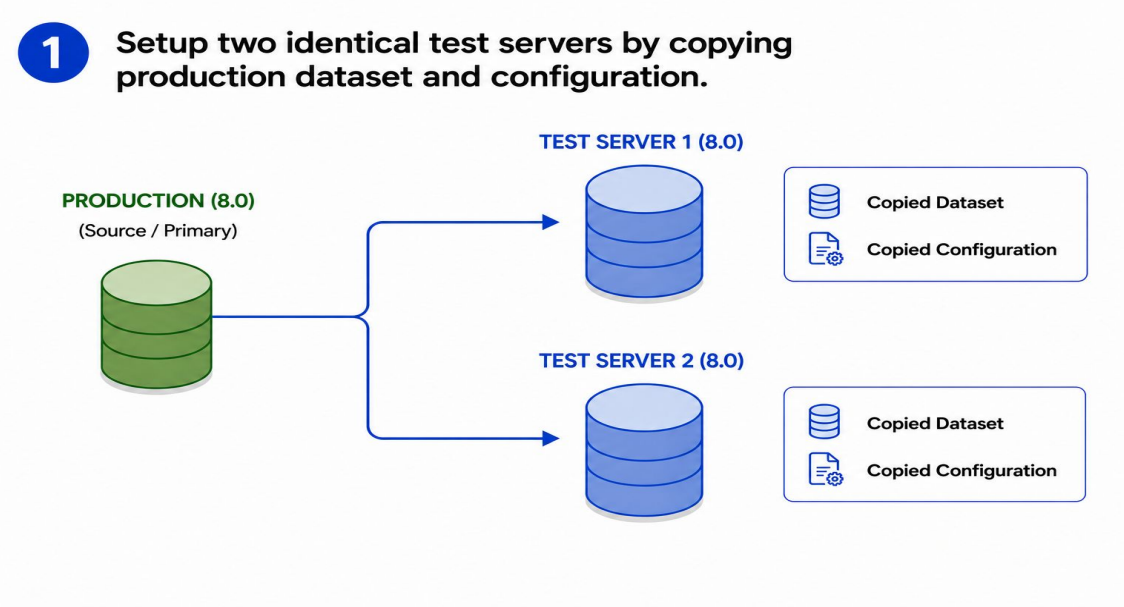
- Two identical non-production servers
- Same data and configuration

Important

Never run pt-upgrade directly on production.

PERCONA

pt-upgrade Workflow



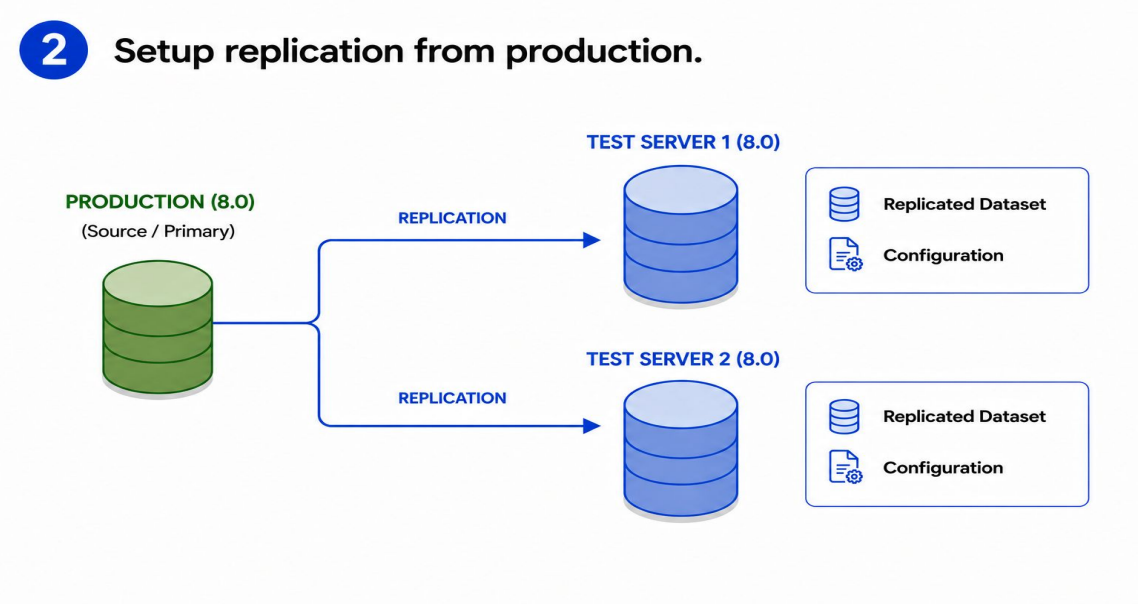
Test Servers Configuration

- Use identical hardware and software configurations.
- Test servers do not need to match production capacity or scale.
- Ensure consistency for:

  CPU & Memory
  Storage type
  Operating system
  MySQL configuration

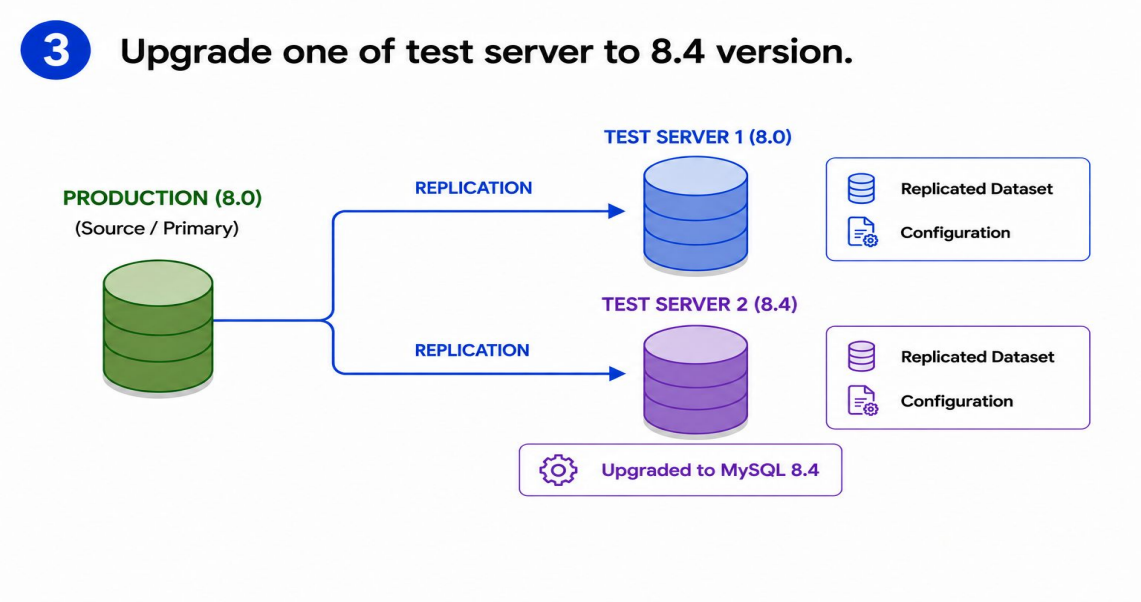
PERCONA

pt-upgrade Workflow



PERCONA

pt-upgrade Workflow



Upgrade Execution Steps

1. Run MySQL Upgrade Checker
Identify potential compatibility issues before starting.

2. Plugins to Components
Upgrade from Plugins to Components architecture.

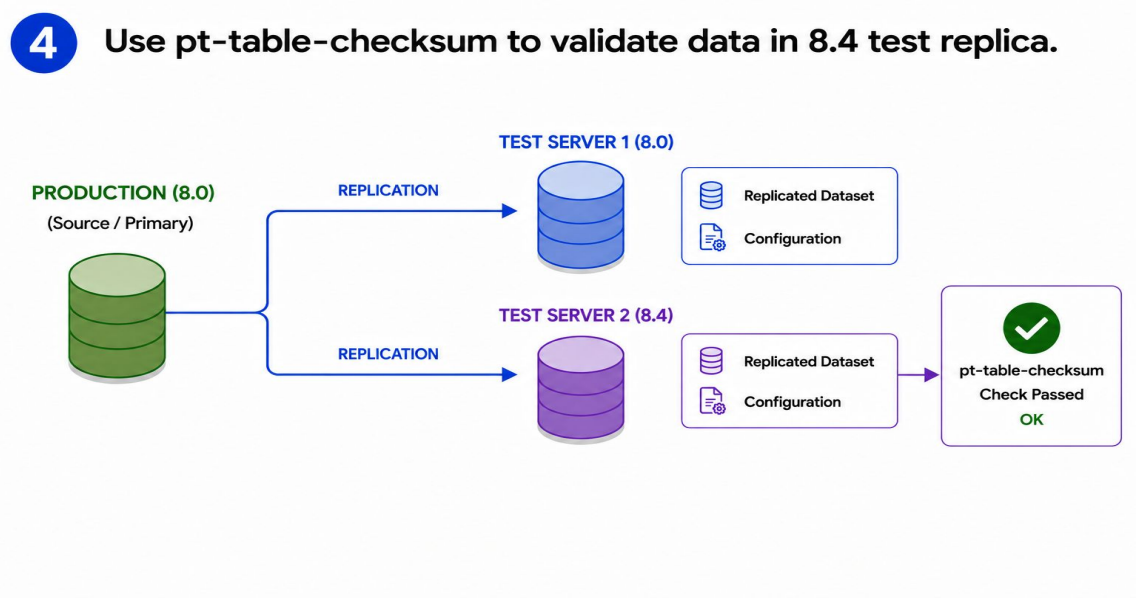
3. Resolve & Document
Fix all reported issues and keep a record of changes.

4. Validate Upgrade Path
Ensure the transition path is supported and stable.

5. Verify Compatibility
Confirm replication works between MySQL 8.0 and 8.4.

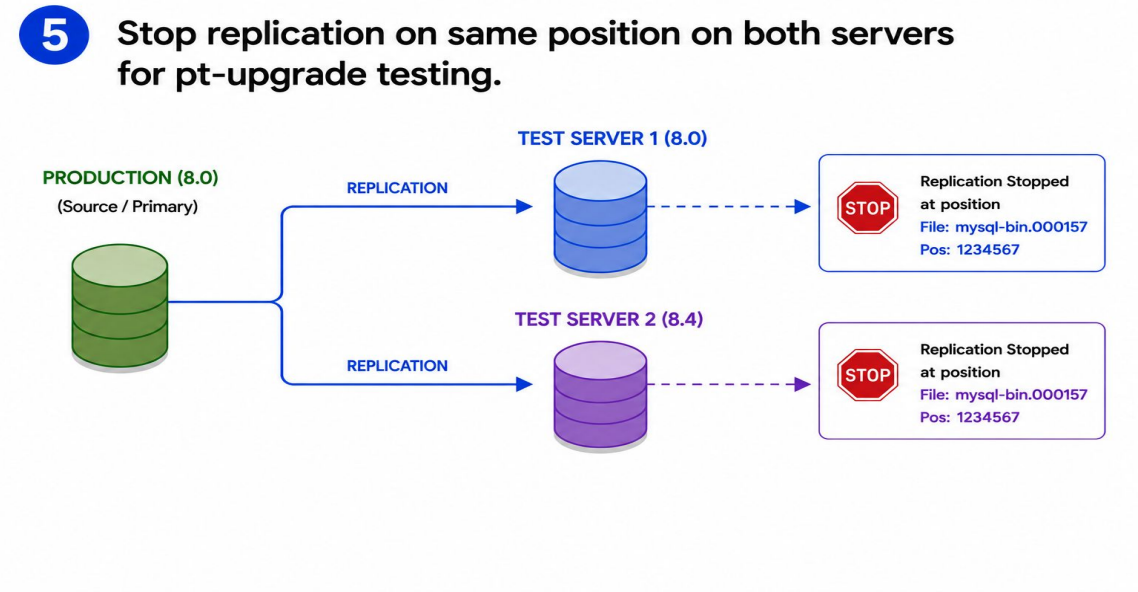
PERCONA

pt-upgrade Workflow



PERCONA

pt-upgrade Workflow



Test Environment Comparison

System Specs

- Same Hardware & OS
- Identical Dataset & Config

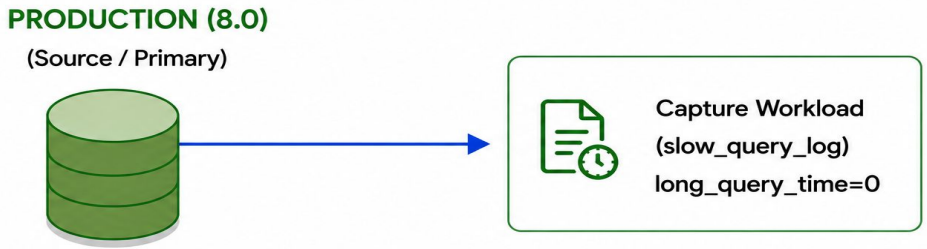
MySQL Versions

- **Test Server 1:** MySQL 8.0
- **Test Server 2:** MySQL 8.4

PERCONA

pt-upgrade Workflow

6 **Capture representative workload (slow_query_log) using long_query_time=0.**

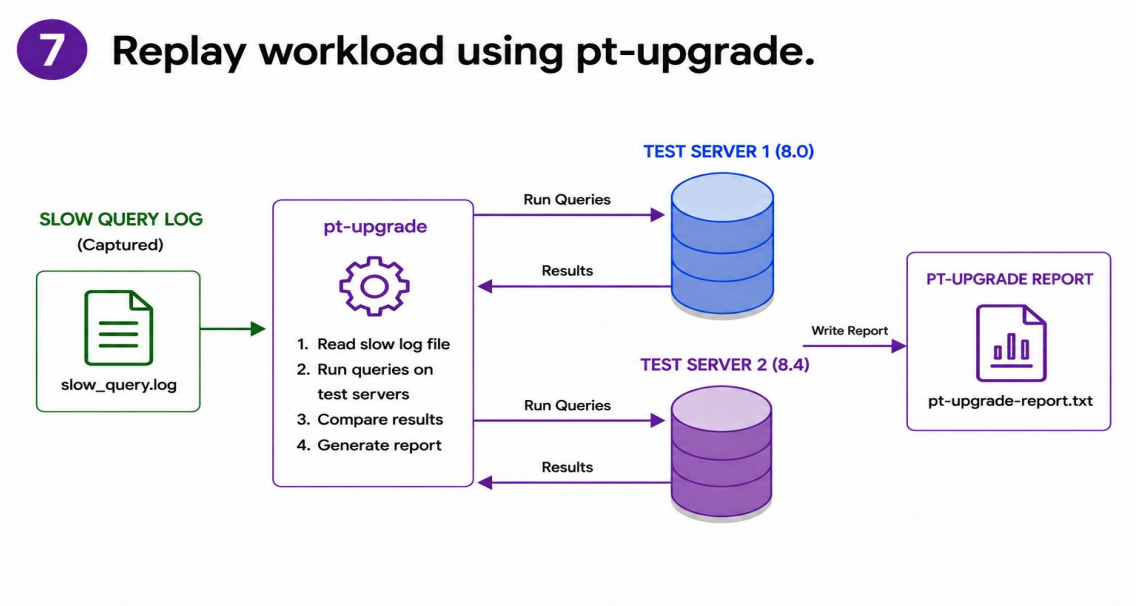



```
pt-query-digest \
--sample 50 \
--no-report \
--output slowlog
 slow.log >
slow_log_50_samples.log
```

- Capture real production workload
- Include reporting and batch jobs
- For busy servers, use **log_slow_rate_limit** to capture a portion of workload
- Take a representative sample using **pt-query-digest**
- Goal: Replay realistic production traffic in testing.

PERCONA

pt-upgrade Workflow



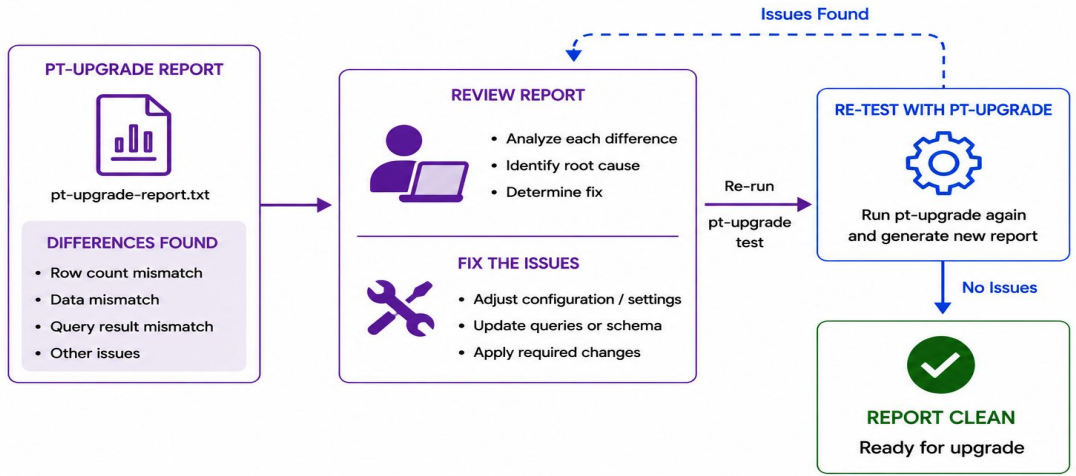

```
$ pt-upgrade \
  h=Test1 \
  h=Test2 \
  slow_log_50_samples.log \
  1> pt-upgrade-report.txt \
  2> pt-upgrade-report.err
```

- **pt-upgrade** only executes SELECT and SET statements
- Use --no-read-only for WRITE queries

PERCONA

pt-upgrade Workflow

8 Review pt-upgrade report and fix issues.



Investigate

- Slower queries
- Changed execution plans
- Different result sets
- Errors and warnings

Action

Document findings and fix before production upgrade.

PERCONA

Application Testing

Compatibility Check

Run the full application test suite against MySQL 8.4 in the test environment to ensure compatibility.

User Acceptance (UAT)

Conduct UAT to confirm that the application behaves as expected under the new database version.

Client Library Update

Upgrade client libraries to ensure support for modern MySQL features and protocol changes.

Performance Validation

Perform real workload tests. Critical for production stability.

PERCONA

Success Criteria

- Stable replication
- Application test suite passes
- Consistent results
- Acceptable latency under load
- No critical query regressions

Final Outcome

Proceed to production only after all blocking issues are resolved.

PERCONA

In-Place Upgrade

PERCONA

In-Place Upgrade

Before Upgrade

⚠!! Take a Backup !!

Upgrade Steps

- Upgrade Replica(s) first.
- Set InnoDB to do a complete/slow shutdown

```
SET GLOBAL innodb_fast_shutdown = 0;
```

- Shutdown MySQL 8.0 gracefully
- Remove MySQL 8.0 Binaries
- Install MySQL 8.4 Binaries
- Review and update my.cnf for MySQL 8.4
- Start MySQL on the existing data directory
- MySQL 8.4 startup will upgrade the data directory to 8.4.x version.
- Review error logs carefully

PERCONA

Post Upgrade Validation

Important Log Messages

Verify these events in the upgrade log

- InnoDB initialization successful
- Data dictionary upgrade started/completed
- Server upgrade started/completed
- Ready for connections

⚠ Watch for

- Startup warnings/errors
- Plugin failures
- Replication issues
- Authentication errors

Upgrade Logs: 8.0.43 → 8.4.8

```
[System] [MY-010116] [Server] /opt/mysql/8.4.8/bin/mysqld (8.4.8) starting...

[System] [MY-013576] [InnoDB] InnoDB initialization has started.

[System] [MY-013577] [InnoDB] InnoDB initialization has ended.

[System] [MY-011090] [Server] Data dictionary upgrading: '80023' to '80300'.

[System] [MY-013413] [Server] Data dictionary upgrade completed.

[System] [MY-013381] [Server] Server upgrade from '80043' to '80408' started.

[System] [MY-013381] [Server] Server upgrade from '80043' to '80408' completed.

[System] [MY-010931] [Server] mysqld: ready for connections.
Version: '8.4.8'
```


PERCONA

MySQL 8.4 Upgrade Recap

PERCONA

MySQL 8.4 Upgrade Recap

Before Upgrade

- ✅ Review MySQL 8.4 changes
- ✅ Take backup and validate
- ✅ Run Upgrade Checker
- ✅ Plugins Migration
- ✅ Complete compatibility testing
- ✅ Application and workload testing

During Upgrade

- ✅ Upgrade replicas first
- ✅ Shutdown with `innodb_fast_shutdown=0`
- ✅ Remove 8.0 Binaries
- ✅ Install 8.4 Binaries
- ✅ Update my.cnf and start 8.4
- ✅ Review upgrade logs

After Upgrade

- ✅ Validate Applications
- ✅ Watch Error Logs
- ✅ Monitor Performance
- ✅ Percona Monitoring and Management (PMM)

PERCONA

How Percona Helps?

Self Managed Upgrade

Percona Documentation

https://docs.percona.com/percona-server/8.4/upgrade.html

Percona Tools to Support 8.4 Upgrade

- Percona XtraBackup
- pt-table-checksum
- pt-upgrade
- Percona Monitoring and Management (PMM)

Percona Community Forum

Connect with experts and peers for upgrade advice and troubleshooting.

https://forums.percona.com/

PERCONA

Upgrade Assistance from Percona Experts

End-to-End Handling

Full execution and management of the upgrade process from start to finish.

Any Upgrade Phase

Flexible support available at any point in your upgrade journey.

Broad Distribution Support

Support for Percona and non-Percona distributions including MySQL Community, MariaDB, RDS, Aurora, Cloud SQL and Azure.

Cutover and Post-Migration Support

Assistance during the final switch and ongoing support for stability and optimization.

PERCONA®

PERCONA®

PERCONA®


PERCONA®