

MySQL 8.0 Release Note

MySQL 8.0 릴리즈노트 번역 겸 요약

- [MySQL 8.0.34 릴리즈 노트](#)
- [MySQL 8.0.33 릴리즈 노트](#)
- [MySQL 8.0.32 릴리즈 노트](#)

MySQL 8.0.34 릴리즈 노트

MySQL 8.0.34 변동 사항 (2023-07-18 GA)

계정 관리 참고 사항

- 이제 새로운 비밀번호 유효성 검사 시스템 변수를 통해 사용자가 자신의 MySQL 계정 비밀번호를 바꾸려고 할 때 변경해야 하는 최소 문자 수를 구성하고 적용할 수 있습니다. 이 새로운 확인 설정은 현재 비밀번호의 총 문자 수에 대한 백분율입니다. 예를 들어, [validate_password.changed_characters_percentage](#)의 값이 50인 경우, 변경하려는 계정 비밀번호에 있는 문자의 절반 이상이 현재 비밀번호에 없어야 하며, 그렇지 않으면 비밀번호가 거부됩니다. 이 새로운 기능은 DBA가 비밀번호 관리를 더욱 완벽하게 제어할 수 있는 여러 기능 중 하나입니다. 자세한 내용 [원 비밀번호 관리](#)를 참고합니다. (WL #15751)

감사 로그 참고 사항

- MySQL 8.0.33에서는, audit_log 플러그인에 JSON 필터 테이블을 어떤 데이터베이스에 저장할 것인지 선택하는 기능이 추가되었습니다. 이제 플러그인 설치 스크립트를 실행할 때 기본 시스템 데이터베이스인 mysql 대신 다른 데이터베이스를 지정할 수 있습니다. 명령줄에서 [audit_log_database](#) 서버 시스템 변수(또는 -D database_name)를 대체 데이터베이스 이름과 함께 사용하면 됩니다. 예를 들면:

```
$> mysql -u root -D database_name -p < audit_log_filter_linux_install.sql
```

audit_log 플러그인 설치 스크립트 사용에 대한 자세한 내용은 [MySQL Enterprise Audit 설치 또는 제거](#)를 참고합니다. (버그 #35252268)

- 이제 MySQL Enterprise Audit에서 스케줄러 구성 요소를 사용하여 인메모리 캐시 플러시를 반복하는 작업을 구성하고 실행할 수 있습니다. 설정 지침은, [감사 로그 플러시 작업 활성화](#)를 참조하십시오. (WL #15567)

바이너리 로그

- 이제 libmysqlclient.so 공유 라이브러리에 개발자가 MySQL 서버 바이너리 로그에 액세스할 수 있는 몇 가지 함수가 추가되었습니다: [mysql_binlog_open\(\)](#), [mysql_binlog_fetch\(\)](#) 및 [mysql_binlog_close\(\)](#) 기여해 주신 유라 소로킨에게 감사드립니다. (버그 #110658, 버그 #35282154)

C API 참고 사항

- 호출 함수에서, net->vio가 null인 경우 len이 0으로 초기화되고 다시 변경되지 않는 문제가 있었습니다. 이번 수정을 통해 vio를 역참조하기 전 net을 먼저 검사하는 기능이 추가 되었습니다. 기여해 주신 메타에게 감사드립니다. (버그 #30809590)
- 특정 코드 경로에서 비동기 클라이언트의 변수가 초기화되지 않는 문제가 있었습니다. 항상 변수를 초기화함으로써 수정하였습니다. 기여해 주신 메타에게 감사드립니다. (버그 #30809590)

컴파일 참고 사항

- 마이크로소프트 윈도우: 윈도우의 경우, [MSVC_CPPCHECK](#) 지원이 개선되었습니다; 또한 'maintainer' 모드에서 처럼 MSVC 경고를 확인하십시오. 예를 들어, 모든 서드파티 설정이 완료된 후 확인하십시오. (버그 #35283401)
참고: 버그 #34828882.
- 마이크로소프트 윈도우: 윈도우 빌드의 경우, WIN_DEBUG_NO_INLINE=1 지원을 개선했습니다; 사용량이 라이브러리 제한인 65535개 오브젝트를 초과할 수 있습니다. (버그 #35259704)
- 번들로 제공되는 로빈 후드 해싱을 v3.8.1에서 v3.11.5로 업그레이드했습니다. (버그 #35448980)
- extra/libcbr/doc/source/requirements.txt가 GitHub의 가짜 풀 리퀘스트로부터 영향을 받으므로 extra/libcbr/doc/ 디렉토리를 제거했습니다. (버그 #35433370)
- icu-data-files 패키지의 번들 ICU 파일을 버전 69.1에서 버전 73으로 업데이트했습니다. (버그 #35353708)
- 소스 트리에 번들로 제공되는 ZSTD 소스가 1.5.0에서 1.5.5로 업그레이드되었습니다. (버그 #35353698).
- TRASH 매크로를 사용하여 내부 MEM_ROOT 클래스 메모리를 가비지로 초기화하여 MEM_ROOT에서 할당된 초기화 메모리를 읽음으로써 발생하는 버그를 쉽게 재현할 수 있도록 합니다. (버그 #35277644)
- 이제 스택 방향을 설정 시점이 아닌 런타임에 결정하게 되었습니다. (버그 #35181008)
- Sanitizer 빌드에 -O1 -fno-inline을 추가하는 [OPTIMIZE_SANITIZER_BUILDS](#) CMake 옵션을 추가했습니다. 기본값은 ON입니다. (버그 #35158758)

- Bison의 최소 버전 요구 사항을 v2.1에서 v3.0.4로 변경했습니다. macOS의 경우 홈브루와 같은 패키지 관리자를 통해 Bison을 설치해야 할 수 있습니다. (버그 #35154645, 버그 #35191333)
- MySQL은 비 ASCII 출력 문제를 피하기 위해 readelf를 실행할 때 환경변수 LANG = C를 설정합니다. 기여해 주신 켄토 타케우치에게 감사드립니다. (버그 #111190, 버그 #35442825)
- macOS에서 Homebrew를 통해 rapidjson을 설치한 경우 MySQL이 컴파일되지 않습니다. 해결 방법은 brew unlink rapidjson이었습니다. (버그 #110736, 버그 #35311140).
참고: 이 문제는 다음의 회귀입니다: 버그 #35006191.
- MySQL은 -DWITH_ZLIB=system에서 빌드되지 않습니다; 시스템 zlib 라이브러리를 찾았음에도 불구하고 찾지 못했다고 메시지를 뿌릴 것입니다. (버그 #110727, 버그 #110745, 버그 #35307674, 버그 #35312227)

사용 중단 및 제거된 기능 참고 사항

- **중요한 변경 사항:** MySQL은 **mysqldump** 및 [MySQL 쉘 유틸리티](#)를 포함하여 동일한 기능 또는 추가 기능으로 데이터베이스 덤프 및 백업을 수행하는 다른 방법을 제공하므로 **mysqldump** 클라이언트 유틸리티 프로그램이 더 이상 필요하지 않습니다. 이제 이 프로그램을 호출하면 경고가 표시됩니다. **mysqldump**는 MySQL의 향후 버전에서 제거될 수 있다는 점에 유의하고, 이에 의존하는 애플리케이션을 앞서 설명한 것과 같은 다른 솔루션으로 이동해야 합니다. (WL #15652)
- **복제:** [sync_relay_log_info](#) 서버 시스템 변수는 이 릴리즈에서 더 이상 사용되지 않으며 이 변수 또는 동등한 시작 옵션 [sync_relay-log-info](#)를 가져오거나 설정하면 경고가 발생합니다.
이 변수는 MySQL의 향후 버전에서 삭제될 예정입니다; 이것을 이용하는 애플리케이션은 제거되기 전에 이에 의존하지 않게 재작성 할 필요가 있습니다. (버그 #35367005, WL #13968)
- **복제:** [binlog_format](#) 서버 시스템 변수는 현재 더 이상 사용되지 않으며 MySQL의 향후 버전에서 삭제될 수 있습니다. 변수와 관련된 기능, 즉 바이너리 로그 형식을 변경하는 기능도 더 이상 사용되지 않습니다.
이 변경의 의미는 binlog_format이 삭제되면 이미 MySQL 8.0의 기본값 인 행-기반 바이너리 로깅만 MySQL 서버에서 지원될 것이라는 이야기입니다. 이러한 이유로, 신규 설치시에는 행 기반 바이너리 로깅만 사용해야 하며, 명령문 기반 또는 혼합 로그 형식을 사용하는 기존 설치본은 행 기반 형식으로 이동해야 합니다. 자세한 내용은 [복제 형식](#)을 참조하십시오.
명령문 기반 로깅 컨텍스트에서만 유용한 시스템 변수 [log_bin_trust_function_creators](#) 및 [log_statements_unsafe_for_binlog](#)도 현재는 더 이상 사용되지 않으며 MySQL의 향후 릴리즈에서 제거될 수 있습니다.
앞에서 언급한 변수의 값을 설정하거나 선택하면 경고가 표시됩니다. (WL #13966, WL #15669)
- **그룹 복제 :** [group_replication_recovery_complete_at](#) 서버 시스템 변수는 더 이상 사용되지 않으며 이것을 설정하면 경고가 생성됩니다. MySQL의 향후 릴리즈에서 삭제될 예정입니다. (WL #15460)
- **mysql_native_password** 인증 플러그인은 현재 더 이상 사용되지 않으며 MySQL의 향후 버전에서 삭제될 수 있습니다. [CREATE USER](#), [ALTER USER](#) 및 [SET PASSWORD](#) 작업에서 계정 인증 방법으로 mysql_native_password를 사용하여 인증을 시도하면 서버 에러 로그에 더 이상 사용되지 않는다는 경고문이 삽입됩니다. (버그 #35336317)
- 이전에는, 규칙 기반 필터링에 필요한 추가 감사 테이블과 함수가 없이 audit_log 플러그인이 설치된 경우, 플러그인이 레거시 필터링 모드에서 작동했었습니다. 이제, 레거시 필터링 모드는 더 이상 사용되지 않습니다. 레거시 감사 로그 필터링 시스템 변수에 대해 새로운 사용 중단 경고가 표시됩니다. 이 사용 중단된 변수는 읽기 전용 또는 동적 변수입니다.
(읽기 전용) [audit_log_policy](#)는 값이 ALL(기본값)이 아닌 경우 서버를 시작하는 동안 MySQL 서버의 에러 로그에 경고 메시지를 씁니다.
(동적) [audit_log_include_accounts](#), [audit_log_exclude_accounts](#), [audit_log_statement_policy](#) 및 [audit_log_connection_policy](#). 동적 변수는 사용 상황에 따라 경고 메시지를 출력합니다:
 - MySQL 서버를 시작하는 동안 [audit_log_include_accounts](#) 또는 [audit_log_exclude_accounts](#)에 널이 아닌 값을 전달하면 경고 메시지가 서버 에러 로그에 기록됩니다.
 - MySQL 서버를 시작하는 동안 [audit_log_statement_policy](#) 또는 [audit_log_connection_policy](#)에 기본값 이외의 값을 전달하면 경고 메시지가 서버 에러 로그에 기록됩니다. ALL은 두 변수의 기본값입니다.
 - MySQL 클라이언트 세션 중에 [SET](#) 구문을 사용하여 기존 값을 변경하면 경고 메시지가 클라이언트 로그에 기록됩니다.
 - MySQL 클라이언트 세션 중에 [SET PERSIST](#) 구문을 사용하여 변수를 영속화하면 경고 메시지가 클라이언트 로그에 기록됩니다. (WL #11248)
- MySQL에서는 시스템 변수와 클라이언트 옵션을 사용하여 서버 측과 클라이언트 측에서 FIPS 모드를 제어할 수 있습니다. 애플리케이션 프로그램은, [mysql_options\(\)](#)의 MYSQL_OPT_SSL_FIPS_MODE 옵션을 사용해, 클라이언트에서 FIPS 모드를 활성화 할 수가 있습니다. 또는 현재 서버 측 시스템 변수와 클라이언트 측 옵션을 사용하는 대신 OpenSSL 구성 파일을 통해 직접 FIPS 모드를 처리할 수 있습니다. MySQL이 OpenSSL 3.0을 사용하여 컴파일되고 OpenSSL 라이브러리와 FIPS 객체 모듈을 런타임에 사용할 수 있는 경우, 서버는 OpenSSL 구성 파일을 읽고 FIPS 공급자가 설정된 경우 해당 기본 설정을 따릅니다. OpenSSL 3.0은 FIPS에서 사용하도록 인증되었습니다.
OpenSSL 대체를 선호하기 위해 [ssl_fips_mode](#) 서버 시스템 변수, [--ssl-fips-mode](#) 클라이언트 옵션 및 MYSQL_OPT_SSL_FIPS_MODE 옵션은 더 이상 사용되지 않으며 MySQL의 향후 버전에서 제거될 수 있습니다. 응용프로그램이 MYSQL_OPT_SSL_FIPS_MODE 옵션을 사용하거나 클라이언트 사용자가 명령행, 옵션 파일 또는 둘 다에서 [--ssl-fips-mode](#) 옵션을 지정하는 경우 더 이상 사용되지 않는다는 경고가 표준 오류 출력으로 출력됩니다.
더 이상 사용되지 않기 전에 [ssl_fips_mode](#) 서버측 시스템 변수는 동적으로 구성 가능했습니다. 이제 이 변수는 읽기 전용 변수입니다 (SET PERSIST_ONLY는 허용되지만 SET PERSIST 또는 SET GLOBAL은 허용되지 않습니다). 명령 행 또는 [mysqld-auto.cnf](#) 옵션 파일 (SET PERSIST_ONLY 사용)에 지정하면 더 이상 사용되지 않는다는 경고가 서버 에러 로그에 출력됩니다. (WL #15631)
- **mysql_ssl_rsa_setup** 프로그램은 초기에 OpenSSL이 시스템에 설치된 경우 커뮤니티 사용자가 인증서를 수동으로 생성할 수 있는 간단한 방법을 제공했습니다. MySQL 커뮤니티 에디션은 더 이상 SSL 라이브러리로 yaSSL을 사용하는 것을 지원하지 않으며 소스 배포판에는 yaSSL이 더 이상 포함되지 않으므로 **mysql_ssl_rsa_setup**은 더 이상 사용되지 않습니다. 대신 MySQL 서버를 사용하여 누락된 SSL 및 RSA 파일을 DB 기동시 자동으로 생성합니다. ([자동 SSL 및 RSA 파일 생성](#) 참조) (WL #15668)
- **keyring_file** 플러그인과 **keyring_encrypted_file** 플러그인은 더 이상 사용되지 않습니다. 이러한 키링 플러그인은 component_keyring_file 및 component_keyring_encrypted_file 컴포넌트로 대체됩니다. [키링 컴포넌트와 키링 플러그인 비교](#)를 참조하십시오. (WL #15659)
- 이전에는 MySQL 서버가 포함된 MySQL 버전 번호 뒤에 공백이 오는지 여부에 관계없이 버전별 주석을 처리했습니다. 예를 들어, 코멘트 `/*!80034KEY_BLOCK_SIZE=1024*/`와 `/*!80034 KEY_BLOCK_SIZE=1024*/`는 같은 방식으로 처리되었습니다. 이

릴리즈부터는 이러한 주석의 버전 번호 다음에 오는 문자가 공백 또는 주석 끝이 아니면 서버는 경고를 발행합니다. 버전 번호 바로 뒤에 버전 주석을 시작하지 않는 것이 좋습니다. 이며 향후 릴리즈에서 동작이 변경될 수 있습니다. 버전 번호 뒤에 공백 문자를 삽입합니다.

MySQL의 향후 버전에서는 버전별 주석에 대한 공백 요구 사항이 엄격하게 적용된다고 가정해야 합니다.

(WL #15686)

- MySQL 클라이언트 라이브러리는 현재, 접속이 다운되고 있는 것으로 판명되어 어플리케이션이 실행된 스테이트먼트를 서버에 송신하려고 했을 경우에, 서버에의 자동 재접속의 실행을 서포트하고 있습니다. 현재 이 기능은 더 이상 사용되지 않으며 MySQL의 향후 릴리즈에서 제거될 수 있습니다.

연관된 MYSQL_OPT_RECONNECT 옵션은 계속 사용할 수 있지만 이것도 더 이상 사용되지 않습니다C API 함수 `mysql_get_option()` 및 `mysql_options()` 는, 어플리케이션이 MYSQL_OPT_RECONNECT 를 지정했을 경우에, 비추천의 경고를 표준 에러 출력에 기입하게 되었습니다. (WL #15766)

- 이전에는 MySQL 서버가 포함된 MySQL 버전 번호 뒤에 공백이 오는지 여부에 관계없이 버전별 주석을 처리했습니다예를 들어, 코멘트 `/*!80034KEY_BLOCK_SIZE=1024*/`와 `/*!80034 KEY_BLOCK_SIZE=1024*/`는 같은 방식으로 처리되었습니다. 이 릴리즈부터는 이러한 주석의 버전 번호 다음에 오는 문자가 공백 또는 주석 끝이 아니면 서버는 경고를 발행합니다:

Immediately starting the version comment after the version number is deprecated and may change behavior in a future release. Please insert a whitespace character after the version number.

MySQL의 향후 버전에서는 버전별 주석에 대한 공백 요구 사항이 엄격하게 적용된다고 가정해야 합니다.

자세한 정보는 [주석](#)을 참조하십시오. (WL #15686)

- MySQL 클라이언트 라이브러리는 현재, 접속이 끊어 졌고 어플리케이션이 명령문을 서버에 송신하려 한다 생각되면, 서버에 대한 자동 재접속 수행을 지원하고 있습니다. 현재 이 기능은 더 이상 사용되지 않으며 MySQL의 향후 릴리즈에서 제거될 수 있습니다. 연관된 MYSQL_OPT_RECONNECT 옵션은 계속 사용할 수 있지만 이것도 더 이상 사용되지 않습니다C API 함수 `mysql_get_option()` 및 `mysql_options()`는 어플리케이션이 MYSQL_OPT_RECONNECT를 지정했을 경우에, 더이상 사용되지 않는다는 경고를 표준 에러 출력에 기입하게 되었습니다. (WL #15766)

Performance Schema 참고 사항

- Performance Schema의 [clone_status](#) 테이블의 `gtid_executed` 컬럼에 사용되던 데이터 타입이 [VARCHAR \(4096\)](#)에서 [LONGTEXT](#)로 변경되었습니다. (버그 #109171, 버그 #34828542)

SQL 문법 참고 사항

- 이제 [CREATE TABLE](#) 및 [ALTER TABLE ... ADD COLUMN](#) 문에서 [VARCHAR](#)와 [TEXT](#)컬럼의 default 값으로 [CURRENT_USER\(\)](#)를 사용할 수 있습니다.

같은 경우에서 [SESSION_USER\(\)](#), [USER\(\)](#) 및 [SYSTEM_USER\(\)](#) 함수도 지원됩니다. 예를 들어, 다음 구문들의 경우 이제 여기에 표시된 것과 유사하게 작동하며, 정확한 출력은 사용자 환경에 따라 달라집니다:

```
mysql> SELECT CURRENT_USER();
+-----+
| CURRENT_USER() |
+-----+
| sakila@localhost |
+-----+
1 row in set (0.00 sec)

mysql> CREATE TABLE t (
  > c1 VARCHAR(288) DEFAULT (USER()),
  > c2 VARCHAR(288) DEFAULT (CURRENT_USER()),
  > c3 VARCHAR(288) DEFAULT (SESSION_USER()),
  > c4 VARCHAR(288) DEFAULT (SYSTEM_USER())
  > );
Query OK, 0 rows affected (0.04 sec)

mysql> INSERT INTO t VALUES ROW();
Query OK, 1 row affected (0.01 sec)

mysql> TABLE t;
+-----+-----+-----+-----+
| c1      | c2      | c3      | c4      |
+-----+-----+-----+-----+
| sakila@localhost | sakila@localhost | sakila@localhost | sakila@localhost |
+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

이러한 방식으로 사용하는 경우, 이 함수들이 [SHOW CREATE TABLE](#) 및 [SHOW COLUMNS](#)의 출력에도 포함되며, Information Schema의 [COLUMNS](#) 테이블의 `COLUMN_DEFAULT` 열에서도 표기됩니다.

이러한 컬럼에 가능한 최대 길이의 값이 확실히 저장될 수 있는지 확인하려면, 열에 최소 288자를 수용할 수 있는지 확인해야 합니다.

다. (사용자 이름으로 255자, 호스트 이름으로 32자, 구분 기호 @로 1글자 추가되어 총 288자) 이러한 이유로, 위 함수 중 하나를 **CHAR** 열의 기본값으로 사용할 수도 있지만 예러가 발생하거나 값이 잘릴 위험이 있으므로 권장하지는 않습니다. (버그 #17809, 버그 #11745618)

추가 또는 변경된 기능

- 중요 변경 사항:** OpenSSL 라이브러리가 번들로 제공되는 플랫폼의 경우, 연결된 MySQL Server용 OpenSSL 라이브러리가 OpenSSL 1.1.1에서 OpenSSL 3.0으로 업데이트 되었습니다. 정확한 버전은 이제 3.0.9입니다. 1.1.1에서 3.0으로의 변경 사항에 대한 자세한 내용은 https://www.openssl.org/docs/man3.0/man7/migration_guide.html 에서 확인할 수 있습니다. (버그 #35475140, WL #15614)
- 패키징:** 이번 릴리즈부터는, 기존에 **-el7-x86_64.tar** 및 **-el7-x86_64.tar.gz** 라는 이름을 사용한 바이너리 아카이브 패키지의 이름이 각각 **-linux-glibc2.17-x86_64.tar** 및 **-linux-glibc2.17-x86_64.tar.gz**로 변경됩니다. 예를 들어, MySQL 8.0.33을 설치 또는 업그레이드할 때 **mysql-8.0.33-el7-x86_64.tar**를 사용했다면, MySQL 8.0.34를 설치 또는 업그레이드할 때는 **mysql-8.0.34-linux-glibc2.17-x86_64.tar**를 사용해야 합니다. 이름이 변경된 것은 이 바이너리들이 glibc2.17 이상을 사용하는 모든 Linux 시스템과 호환된다는 사실을 반영한 것입니다; 여기에는 Enterprise Linux 7, EL8 및 EL9가 포함됩니다.
- 시스템의 curl 라이브러리에 연결하는 것이 아닌, 자체적으로 curl을 포함하는 바이너리 패키지의 경우 curl 8.1.2를 사용하도록 업그레이드되었습니다. (버그 #35329529)

버그 수정

- Important Change:** The default value of the `connection_memory_chunk_size` server system variable, when introduced in MySQL 8.0.28, was mistakenly set at 8912. This fix changes the default to 8192, which is the value originally intended. (Bug #35218020)
- NDB Cluster:** The fix for a previous issue introduced a slight possibility of unequal string values comparing as equal, if any Unicode 9.0 collations were in use, and the collation hash methods calculated identical hash keys for two unequal strings. (Bug #35168702)
References: See also: Bug #27522732. This issue is a regression of: Bug #30884622.
- InnoDB:** Possible congestion due to purging a large number of system threads has been fixed. (Bug #35289390, Bug #110685)
- InnoDB:** `ddl::Aligned_buffer` now uses the standard memory allocator and not kernel memory management. (Bug #35194732, Bug #110411)
- InnoDB:** An upgrade from MySQL 5.7 to MySQL 8.0.32 might fail due to deprecated configuration parameters `innodb_log_file_size` or `innodb_log_files_in_group`. The workaround is to start MySQL 8.0.32 with `--innodb-redo-log-capacity=206158430208`. (Bug #35155280)
- InnoDB:** The rules for aggregating entries in the redo log have been fixed. (Bug #34752625, Bug #108944)
- InnoDB:** Several errors due to tablespace deletion and the buffer pool have been fixed. (Bug #34330475, Bug #107689)
- Packaging; Group Replication:** The group replication plugin from the Generic Linux packages did not load on some platforms that lacked a compatible version of `librt`. (Bug #35323208)
- Replication:** Changes in `session_track_gtids` were not always propagated correctly. (Bug #35401212)
- Replication:** By design, all DDL operations (including binary log operations such as purging the binary log) acquire a shared lock on the `BACKUP_LOCK` object, which helps to prevent simultaneous backup and DDL operations. For binary log operations, we checked whether any locks existed on `BACKUP_LOCK` but did not check the types of any such locks. This caused problems due to the fact that binary log operations should be prevented only when an exclusive lock is held on the `BACKUP_LOCK` object, that is, only when a backup is actually in progress, and backups should be prevented when purging the binary log. Now in such cases, instead of checking for locks held on the `BACKUP_LOCK` object, we acquire a shared lock on `BACKUP_LOCK` while purging the binary log. (Bug #35342521)
- Replication:** In all cases except one, when `mysqlbinlog` encountered an error while reading an event, it wrote an error message and returned a nonzero exit code, the exception being for the active binary log file (or any binary log where the `format_description_log_event` had the `LOG_EVENT_BINLOG_IN_USE_F` flag set), in which case it did not write a message, and returned exit code 0, thus hiding the error. Now `mysqlbinlog` suppresses only those errors which are related to truncated events, and when doing so, prints a comment rather than an error message. This fix also improves the help text for the `--force-if-open` option. (Bug #35083373)
- Replication:** Compressed binary log event handling was improved. (Bug #33666652)
- Replication:** A transaction consisting of events each smaller than 1 GiB, but whose total size was larger than 1 GiB, and where compression did not make it smaller than 1 GiB, was still written to the binary log as one event bigger than 1 GiB. This made the binary log unusable; in effect, it was corrupted since neither the server nor other tools such as `mysqlbinlog` could read it. Now, when the compressed data grows larger than 1 GiB, we fall back to processing the transaction without any compression. (Bug #33588473)
- Group Replication:** In a group replication setup, when there was a source of transactions other than the applier channel, the following sequence of events was possible:
 - Several transactions being applied locally were already certified, and so were associated with a ticket, which we refer to as Ticket 2, but had not yet been committed. These could be local or nonlocal transactions.
 - A view is created with Ticket 3, and must wait on transactions from Ticket 2.
 - The view change (VC1) entered the GR applier channel applier and waited for the ticket to change to 3.
 - Another group change, and another view change (VC2), occurred while the transactions from Ticket 2 were still completing.

This gave rise to the following issue: There was a window wherein the last transaction from Ticket 2 had

already marked itself as being executed but had not yet popped the ticket; VC2 popped the ticket instead but never notified any of the participants. This meant that VC1 continued to wait indefinitely for the ticket to change, and with the additional effect that the worker could not be killed.

We fix this by checking for the need to break each second so that this loop is responsive to changes in the loop condition; we also register a new stage, so that the loop is more responsive to kill signals. (Bug #35392640)

References: See also: Bug #35206392, Bug #35374425.

- **Group Replication:** Removed a memory leak discovered in `Network_provider_manager::open_xcom_connection()`. (Bug #34991101)
- **Group Replication:** When a group action was sent to the group and the connection was killed on the coordinator, group members were in different states, with members which received the coordinated action waiting for the member that executed it, and the member which started execution having nothing to process, which caused problems with coordination of the group. Now in such cases, we prevent this issue from occurring by causing group actions to wait until all members have completed the action. (Bug #34815537)
- **Group Replication:** Cleanup of resources used by OpenSSL connections created indirectly by group replication was not carried out as expected at all times. We fix this by adding cleanup functionality that can be called at any time such connections are created by group replication. (Bug #34727136)
- **JSON:** When the result of `JSON_VALUE()` was an empty string and was assigned to a user variable, the user variable could in some cases be set to `NULL` instead, as shown here:

```
mysql> SELECT JSON_VALUE('{"fname": "Joe", "lname": ""}', '$.lname') INTO @myvar;
Query OK, 1 row affected (0.01 sec)
```

```
mysql> SELECT @myvar = "", @myvar IS NULL;
```

```
+-----+-----+
| @myvar = " | @myvar IS NULL |
+-----+-----+
|      NULL |           1 |
+-----+-----+
1 row in set (0.00 sec)
```

With this fix, the query just shown now returns `(1, 0)`, as expected. (Bug #35206138)

- **JSON:** Some JSON schemas were not always processed correctly by `JSON_SCHEMA_VALID()`. (Bug #109296, Bug #34867398)
- In rare cases, MySQL server could exit rather than emit an error message as expected. (Bug #35442407)
- The internal resource-group enhancement added in MySQL 8.0.31 and refactored in MySQL 8.0.32 is now reverted. (Bug #35434219)
- References: Reverted patches: Bug #34702833.
- An in-place upgrade from MySQL 5.7 to MySQL 8.0, without a server restart, could result in unexpected errors when executing queries on tables. This fix eliminates the need to restart the server between the upgrade and queries. (Bug #35410528)
- A fix in MySQL 8.0.33 made a change for `ORDER BY` items already resolved so as not to resolve them again (as is usually the case when a derived table is merged), but this did not handle the case in which an `ORDER BY` item was itself a reference. (Bug #35410465)
- References: This issue is a regression of: Bug #34890862.
- Changes in `session_track_qtids` were not always handled correctly. (Bug #35401212)
- Some pointers were not always released following statement execution. (Bug #35395965)
- Some instances of subqueries within stored routines were not always handled correctly. (Bug #35377192)
- Fortified parsing of the network packet data sent by the server to the client. (Bug #35374491)
- Encryption enhancements now strengthen compliance and remove the use of deprecated APIs. (Bug #35339886)
- When a column reference given by table name and column name was looked up in the function `find_item_in_list()`, we ignored that the item searched for might not have a table name, as it was not yet resolved. We fix this by making an explicit check for a null table name in the sought-after item. (Bug #35338776)
- Deprecated the `lz4_decompress` and `zlib_decompress` command-line utilities that exist to support the deprecated `mysqlpump` command-line utility. (Bug #35328235)
- Queries using `LIKE '%...%'` ran more poorly than in previous versions of MySQL. (Bug #35296563)
- In `Bounded_queue::push()`, when `Key_generator::make_sortkey()` returns `UINT_MAX` (error), then no key has been produced; now when this occurs, we no longer update the internal queue. As part of this fix, `push()` now returns true on error. (Bug #35237721)
- The `authentication_oci` plugin is fixed to allow federated and provisioned users to connect to a DB System as a mapped Proxy User using an ephemeral key-pair generated through the OCI CLI. (Bug #35232697)
- Some queries using common table expressions were not always processed correctly. (Bug #35231475)
- The internal function `compare_pair_for_nulls()` did not always set an explicit return value. (Bug #35217471)
- Removed the clang-tidy checks that clash with the MySQL coding style. (Bug #35208735)
- Some subqueries using `EXISTS` in both the inner and outer parts of the query were not handled correctly. (Bug #35201901)
- Rotated audit log files now always reset the ID value of the bookmark to zero, rather than continuing the value from the previous file. (Bug #35200070)
- Errors were not always propagated correctly when evaluating items to be sorted by filesort. (Bug #35195181)
- References: See also: Bug #35145246.
- The fix for a previous issue with `ROLLUP` led to a premature server exit in debug builds. (Bug #35168639)
- References: This issue is a regression of: Bug #33830659.
- Simplified the implementation of `Item_func_make_set::val_str()` to make sure that we never try to reuse any of the input arguments, always using the local string buffer instead. (Bug #35154335, Bug #35158340)
- When transforming subqueries to a join with derived tables, with the containing query being grouped, we created an extra derived table in which to do the grouping. This process moved the initial select list items from the containing query into the extra derived table, replacing all of the original select list items (other than

subqueries, which get their own derived tables) with columns from the extra derived table.

This logic did not handle `DEFAULT` correctly due to the manner in which default values were modelled internally. This fix adds support for `DEFAULT(expression)` in queries undergoing the transform previously mentioned. This fix also solves an issue with item names in metadata whereby two occurrences of the same column in the select list were given the same item name as a result of this same transform. (Bug #35150085, Bug #35101169)

- A query of the form `SELECT * FROM t1 WHERE (SELECT a FROM t2 WHERE t2.a=t1.a + ABS(t2.b)) > 0` should be rejected with Subquery returns more than 1 row, but when the `subquery_to_derived` optimization was enabled, the transform was erroneously applied and the query returned an incorrect result. (Bug #35101630)
- Handling of certain potentially conflicting `GRANT` statements has been improved. (Bug #35089304)
- A query using both `MEMBER OF()` and `ORDER BY DESC` returned only a partial result set following the creation of a multi-valued index on a JSON column. This is similar to an issue fixed in MySQL 8.0.30, but with the addition of the `ORDER BY DESC` clause to the problematic query. (Bug #35012146)

References: See also: Bug #106621, Bug #33917625.

- The debug server asserted on certain operations involving `DECIMAL` values. (Bug #34973932)
- All instances of adding and replacing expressions in the select list when transforming subqueries to use derived tables and joins have been changed so that their reference counts are maintained properly. (Bug #34927110)
- Index Merge (see [Index Merge Optimization](#)) should favor ROR-union plans (that is, using RowID Ordered Retrieval) over sort-union plans if they have similar costs, since sort-union requires additionally sorting of the rows by row ID whereas ROR-union does not.

For each part of a `WHERE` clause containing an `OR` condition, the range optimizer gets the best range scan possible and uses all these range scans to build an index merge scan (that is, a sort-union scan). If it finds that all the best range scans are also ROR-scans, the range optimizer always proposes a ROR-union scan because it is always cheaper than a sort-union scan. Problems arose when the best range scan for any one part of an `OR` condition is not a ROR-scan, in which case, the range optimizer always chose sort-union. This was true even in cases, where it might be advantageous to choose a ROR-scan (even though it might not be the best range scan to handle one part of the OR condition), since this would eliminate any need to sort the rows by row ID.

Now, in such cases, when determining the best range scan, the range optimizer also detects whether there is any possible ROR-scan, and uses this information to see whether each part of the `OR` condition has at least one possible ROR-scan. If so, we rerun the range optimizer to obtain the best ROR-scan for handling each part of the `OR` condition, and to make a ROR-union path. We then compare this cost with the cost of a sort-union when proposing the final plan. (Bug #34826692, Bug #35302794)

- Selecting from a view sometimes raised the error Illegal mix of collations ... for operation '=' when the collation used in the table or tables from which the view definition selected did not match the current session value of `collation_connection`. (Bug #34801210)
- Valid MySQL commands (`use` and `status`) and C API functions (`mysql_refresh`, `mysql_stat`, `mysql_dump_debug_info`, `mysql_ping`, `mysql_set_server_option`, `mysql_list_processes`, and `mysql_reset_connection`) could write an error message to the audit log, even though running the command or calling the function emitted no such error. (Bug #33966181)
- Increased the maximum fixed array size to 8192 instead of 512. This fixes an issue with `mysqldadmin` extended status requests, which can exceed 512 entries. Our thanks to Meta for the contribution. (Bug #30810617)
- The `mysqldump --column-statistics` option attempted to select from `information_schema.column_statistics` against MySQL versions before 8.0.2, but this now generates the warning column statistics not supported by the server and sets the option to false. Our thanks to Meta for the contribution. (Bug #28782417)
- The function used by MySQL to get the length of a directory name was enhanced. (Bug #28047376)
- Executing a query with an implicit aggregation should return exactly one row, unless the query has a `HAVING` clause that filters out the row, but a query with a `HAVING` clause which evaluated to `FALSE` sometimes ignored this, and returned a row regardless. (Bug #14272020)
- For a query with a derived condition pushdown where a column in the condition needs to be replaced, a matching item could not found, even when known to be present, when the replacement item was wrapped in a `ROLLUP` while the matching item was not. (Bug #111665, Bug #35570065)
- References: This issue is a regression of: Bug #33349994.
- The presence of an unused window function in a query, along with an `ORDER BY` that could have been eliminated, led to an unplanned server exit. (Bug #111585, Bug #35168639, Bug #35204224, Bug #35545377)
- References: This issue is a regression of: Bug #35118579.
- `ORDER BY RANDOM_BYTES()` had no effect on query output. (Bug #111252, Bug #35148945, Bug #35457136)
- Fixed an issue which could occur when loading user-defined functions. (Bug #110576, Bug #35242734)
- Concurrent execution of `FLUSH STATUS`, `COM_CHANGE_USER`, and `SELECT FROM I_S.PROCESSLIST` could result in a deadlock. A similar issue was observed for concurrent execution of `COM_STATISTICS`, `COM_CHANGE_USER`, and `SHOW PROCESSLIST`. Our thanks to Dmitry Lenev for the contribution. (Bug #110494, Bug #35218030)
- The `mysqldump` utility could generate invalid INSERT statements for generated columns. (Bug #110462, Bug #35208605)
- During optimization, range-select tree creation uses logic which differs based on the left-hand side of the `IN()` predicate. For a field item, each value on the right-hand side is added to an OR tree to create the necessary expression. In the case of a row item comparison (example: `WHERE (a,b) IN ((n1,m1), (n2, m2), ...)`), an expression in disjunctive normal form (DNF) is needed. A DNF expression is created by adding an `AND` tree with column values to an `OR` tree for each set of RHS values, but instead the `OR` tree was added to the `AND` tree causing the tree merge to require exponential time due to $O(n^2)$ runtime complexity. (Bug #108963, Bug #34758905)
- When using `SELECT` to create a table and the statement has an expression of type `GEOMETRY`, MySQL could generate an empty string as the column value by default. To resolve this issue, MySQL no longer generates default values for columns of type `GEOMETRY` under these circumstances. Our thanks to Tencent for the contribution. (Bug #107996, Bug #34426943)
- For index skip scans, the first range read set an end-of-range value to indicate the end of the first range, but the next range read did not clear the stale end-of-range value and applies this stale value to the current

range. Since the indicated end-of-range boundary had already been crossed in the previous range read, this caused the reads to stop, causing multiple rows to be missed in the result. We fix this by making sure in such cases that the old end-of-range value is cleared. (Bug #107460, Bug #34235624, Bug #34982949)

MySQL 8.0.33 릴리즈 노트

MySQL 8.0.33 변동 사항 (2023-04-18 GA)

감사 로그 참고 사항

- 이전에 MySQL Enterprise Audit는 필터 및 사용자 계정 데이터의 영구 저장을 위해 mysql 시스템 데이터베이스의 테이블을 사용했습니다. 유연성을 높이기 위해, 새로운 [audit_log_database](#) 서버 시스템 변수를 사용하면 서버를 시작할 때 글로벌 스키마 네임스페이스의 다른 데이터베이스를 지정할 수 있습니다. 테이블 스토리지의 기본값은 mysql 시스템 데이터베이스입니다. (WL#15500)

컴파일 참고 사항

- 마이크로소프트 윈도우: Visual Studio 2017 이상의 MSVC 코드 분석 지원이 추가되었습니다. 이렇게 하면 현재 디렉터리와 해당 하위 디렉터리에서 이 분석을 활성화/비활성화하도록 설정하는 MSVC_CPPCHECK(기본값은 OFF) CMake 옵션이 추가됩니다. (버그 #34828882)
- MySQL이 GNU 컴파일러 또는 clang으로 빌드되며 curl 버전이 7.86보다 큰 경우, curl 대체 예정 경고가 -Wno-error로 다운그레이드되었습니다. (버그 #35111625)
- macOS에서 필요에 따라 공유 시스템 라이브러리와 링크하기 위해 curl 인터페이스를 연결할 때 -framework CoreFoundation 및 -framework SystemConfiguration을 추가했습니다. (버그 #35104962)
- MY_INCLUDE_SYSTEM_DIRECTORIES 매크로를 라이브러리 인터페이스로 대체했습니다. (버그 #35018072, 버그 #35028089, 버그 #35072295)
- 대체 링커를 지원하도록 CMake 코드를 개선했습니다. (버그 #34963568)
- 더이상 사용되지 않는 Docs/mysql.info 파일을 빌드 시스템에서 제거했습니다. (버그 #34960126)
- 최상위 .clang-tidy 파일과 관련된 .clang.tidy 파일을 strings/ 디렉터리 및 mysys/ 디렉터리에 추가했습니다. 또한 compdb 지원을 활성화하여 해당 파일에서 clang-tidy 사용을 활성화했습니다. (버그 #34917075)
- uca-dump나 uctypedump와 같은 기능들에 유지 보수되지 않거나 사용되지 않은 C++ 소스 파일을 일부 삭제했습니다. (버그 #34898978)
- 커맨드 라인에서 컴파일 할 때 GCC 및 Clang의 컬러 컴파일러 출력을 활성화하는 CMake 빌드 옵션이 추가되었습니다. 활성화하려면 -DFORCE_COLORED_OUTPUT=1을 CMake에 전달합니다. (버그 #34897192)
- 윈도우의 경우, 써드 라이브러리에서 관련된 .dll 파일이 발견되면 해당 .pdb 파일도 설치합니다. (버그 #34863555)
- 엔터프라이즈 리눅스 8 빌드 및 엔터프라이즈 리눅스 9 빌드는 이제 GCC 11 대신 GCC 12를 사용합니다. (버그 #34829151)
- static-libgcc -static-libstdc++를 사용하여 빌드하면, 필요에 따라, 정적 라이브러리를 사용하여 번들된 프로토콜 버퍼도 같이 빌드합니다. (버그 #110216, 버그 #35123848)

컴포넌트 참고 사항

- 이제 [INSTALL COMPONENT](#)는 하나 이상의 컴포넌트를 설치하는 동안 컴포넌트 시스템 변수의 값을 설정하는 SET 절이 추가됩니다. 이 새로운 절은 변수 값을 할당하는 다른 방법과 관련된 불편과 제약점을 줄여줍니다. 사용법에 대한 자세한 내용은, [INSTALL COMPONENT 문](#)을 참조하세요. (WL #10916)

사용 중단 및 제거된 기능 참고 사항

- 사용자 정의 콜레이션([문자 집합에 콜레이션 추가하기](#) 참고)은 더 이상 사용되지 않습니다. 다음 중 하나를 사용하면 경고가 로그에 기록됩니다:
 - SQL 문에서 COLLATE 다음에 사용자 정의 콜레이션명을 넣는 경우
 - [collation_server](#), [collation_database](#) 또는 [collation_connection](#)의 값으로 사용자 정의 콜레이션명을 사용하는 경우

사용자 정의 콜레이션에 대한 지원은 향후 MySQL 버전에서 제거될 예정입니다. (WL #14277)

MySQL Enterprise 참고 사항

- MySQL Enterprise Edition은 이제 MySQL 8.0.13에서 도입된 플러그인 라이브러리를 기반으로 하지 않고 컴포넌트 기반 데이

터 마스크 및 익명화 기능을 제공합니다. 컴포넌트 구현은 사건을 관리하기 위한 전용 권한을 제공하고 다음을 포함하는 특정 유형의 목록을 확장합니다.

- Canada Social Insurance Number(캐나다 사회 보험 번호)
- United Kingdom National Insurance Number(영국 국민 보험 번호)
- International Bank Account Number(국제 은행 계좌 번호)
- Universally Unique Identifier (UUID 범용 고유 식별자)

향상된 테이블 기반 사전 레지스트리는 플러그인에서 사용하는 파일 기반 사전을 대체합니다. 컴포넌트와 플러그인 구현 간의 차이점에 대한 요약은 [데이터 마스크 컴포넌트와 데이터 마스크 플러그인 비교](#)를 참조하십시오. 기존 플러그인 사용자는 새로운 MySQL Enterprise 데이터 마스크 및 익명화 컴포넌트를 설치하기 전에 서버측 플러그인을 제거하고 로드 가능한 기능을 제거해야 합니다. (버그 #33851601, WL #12641)

Performance Schema 참고 사항

- 이번 릴리즈에는 Performance Schema Server Telemetry Traces 서비스가 추가되었습니다. SQL문의 수명과 관련된 알림을 검색할 수 있는 방법을 플러그 인과 컴포넌트에 제공하는 인터페이스입니다. 아래 항목들이 추가되었습니다:

- 상태 변수 [Telemetry traces supported](#). 서버 원격 측정 추적이 지원되는지 여부. (Boolean 값)
- TELEMETRY_ACTIVE 열이 [threads](#) 테이블에 추가되었습니다. 스레드에 활성 원격 측정 세션이 연결되어 있는지 여부를 나타냅니다.

(WL #15059)

추가 또는 변경된 기능

- **중요한 변경 사항:** OpenSSL 라이브러리가 번들로 제공되는 플랫폼의 경우, 링크된 MySQL 서버용 OpenSSL 라이브러리가 버전 1.1.1t로 업데이트되었습니다. OpenSSL 버전 1.1.1t에서 수정된 문제는 <https://www.openssl.org/news/cl111.txt>에 설명되어 있습니다. (버그 #35092429)
- **복제:** MySQL 제품에서 사용되는 오래된 용어들을 변경하기 위해 진행중인 작업의 일환으로 "master", "slave", "MTS"라는 용어를, MySQL 복제 관련된 오류 메시지에서, "source", "replica", "MTA"로 대체하였습니다. 여기에는 messages_to_clients.txt 및 messages_to_error_log.txt에 나열된 복제와 관련된 모든 오류 메시지가 포함됩니다; 현재 작업에서는 (에러 메시지가 아닌 / 복제 관련이 아닌) 다른 컨텍스트에서 사용되는 메시지에 대해서는 변경을 수행하지 않았습니다. 자세한 내용은 [MySQL 8.0 오류 메시지 레퍼런스](#)를 참조하세요. (버그 #108422, 버그 #34594819, WL #14191)
- **복제:** [mysqlbinlog --start-position](#)은 이제 최대 18446744073709551615까지의 값을 허용합니다. 그러나 [--read-from-remote-server](#) 또는 [--read-from-remote-source](#) 옵션이 같이 사용되는 경우 최대값은 4294967295입니다. (버그 #77818, 버그 #21498994)
- 시스템의 curl 라이브러리에 링크하는 대신, curl을 포함하고 있는 바이너리 패키지의 경우 curl 버전 7.88.1을 사용하도록 업그레이드되었습니다. (버그 #34828111)
- 디폴트 값으로 [DEFAULT\(col_name\)](#)이란 형태로 생성하여 기본값으로 컬럼명을 지정 하는 것은 이제 허용되지 않으며 오류 메시지가 출력됩니다. (버그 #34463652, 버그 #34369580)
- 이제새로운 [TELEMETRY_LOG_ADMIN](#) 권한으로 원격 분석 로그 구성을 활성화할 수 있습니다. 이 권한은 MySQL HeatWave on AWS를 통해 독점적으로 배포되는 telemetry_log 플러그인에 의해 정의됩니다. (Bug #111395, Bug #35494180)
- 이제 성공적으로 실행된 명령문에 대하여 Statement ID를 클라이언트에게 반환할 수 있습니다. 세션별로 이 기능을 사용하려면 [session_track_system_variables](#)의 값으로 statement_id를 주거나, [session_track_system_variables](#)의 값으로 특별 값인 * (아스테리스크)을 지정합니다. (WL #15418)

버그 수정

- **NDB Cluster:** Occasional temporary errors which could occur when opening a table from the [NDB](#) dictionary while repeatedly performing concurrent schema operations were not retried. (Bug #34843889)
- **NDB Cluster:** During iteration, ordered index scans retain a cursor position within each concurrently scanned ordered index fragment. Ordered index fragments are modified and balanced as a result of committing DML transactions, which can require scan cursors to be moved within the tree. When running with query threads configured ([AutomaticThreadConfig](#) set to 1), multiple threads can access the same index fragment tree structure, and the scans of multiple threads can have their cursors present in the same structure. The current issue arose due to an assumption in the logic for moving scan cursors when committing DML operations that all scan cursors belonged to the LDM thread owning the index fragment, which did not allow for the possibility that such fragments might belong to query threads. (Bug #33379702)
References: See also: Bug #32257063.
- **InnoDB:** Dead code removal. (Bug #35036850, Bug #109873)

- **InnoDB:** Error messages related to [innodb_doublewrite](#) moved to the error log. (Bug #34883045, Bug #109330)
- **InnoDB:** Prevent online DDL operations from accessing out-of-bounds memory. (Bug #34750489, Bug #108925)
- **InnoDB:** [ALTER TABLE ... AUTO INCREMENT](#) could be set to less than MAX + 1 and not forced to MAX + 1. (Bug #33419246, Bug #105092)
- **InnoDB:** [Innodb_data_pending_fsyncs](#) could show extremely high inaccurate values because of a variable overflow. (Bug #30133150)
- **Partitioning:** Some `IN()` queries on partitioned tables were not always handled correctly. (Bug #34801284)
References: This issue is a regression of: Bug #32311183.
- **Partitioning:** Queries using the [INDEX MERGE](#) optimizer hint was not handled correctly in all cases. (Bug #34797257)
- **Replication:** XA transactions whose XIDs contained null bytes could not be recovered. (Bug #34918985)
- **Replication:** When [binlog_order_commits](#) was set equal to `1`, for any two transactions and for any sub-step of the commit phase, the transaction that was written to the binary log first did not always execute the sub-step first, as expected. (Bug #34703698)
- **Replication:** Some binary log events were not always handled correctly. (Bug #34617506)
- **Replication:** The binary log recovery process did not report all possible error states. (Bug #33658850)
- **Replication:** Following [CHANGE REPLICATION SOURCE TO SOURCE CONNECTION AUTO FAILOVER=1](#), failover generated a number of misleading warnings in the log that implied there were problems when in fact conditions were those expected for such a failover. These log messages have been updated accordingly. (Bug #32135376)
- **Replication:** When a transaction failed, as a side effect, extraneous error messages relating the replication data repositories were written to the log. Now in such cases, we suppress such error messages, which are not directly related to the issue of the failed transaction or its cause. (Bug #19820134)
- **Replication:** Setting [binlog_order_commits](#) to `OFF` could lead to a missed GTID in the next binary log file's `Previous_gtid` event.
Our thanks to Yewei Xu and the Tencent team for the contribution. (Bug #109485, Bug #34930969)
- **Replication:** Corrected the SQL statements suggested in the error message text for [ER_RPL_REPLICA_ERROR_RUNNING_QUERY](#).
Our thanks to Dan McCombs for the contribution. (Bug #109154, Bug #34822612)
- **Replication:** A hash scan builds a hash of changes, scans the target table or index, and applies any matching change for the current entry. In the build phase, it uses only the before image, and skips any after image. Problems arose in some cases because generated columns were computed for the (skipped) after image, leading to replication errors. This is fixed by not computing generated columns any longer for seek-only calls such as hash scans.
Our thanks to dc huang for the contribution. (Bug #107366, Bug #34198907)
- **Replication:** In certain rare cases, it was possible to set [gtid_mode=OFF](#) for one session while another session, after [WAIT_FOR_EXECUTED_GTID_SET\(\)](#) was issued by a user in this second session, was still waiting for the next GTID set from the first session. This could result in the second session waiting indefinitely for the function to return. (Bug #99921, Bug #31505993)
- **Group Replication:** Accessing the Performance Schema [replication_group_communication_information](#) and [replication_group_member_stats](#) tables in parallel sometimes caused subsequent group replication operations to hang. (Bug #34870181)
- **Group Replication:** In certain cases, the group replication secondary node unexpectedly shut down while purging the relay log. (Bug #34397106)
- **Group Replication:** When shutting down the Group Replication plugin, the order in which the associated events were reported the error log sometimes led to confusion. To remove any doubts, we now make sure that `Plugin group_replication reported: 'Plugin 'group_replication' has been stopped.'` is in fact the last log message relating to the shutdown, written only when all other events associated with shutting down the plugin have been logged. (Bug #109345, Bug #34887491)
- **Microsoft Windows:** The [authentication_fido_client](#) plugin stopped responding during the authentication process if it was unable to find a FIDO device on the Windows client host. (Bug #34918044)
- In certain cases, [CONVERT\(utf8mb3_column USING UTF16\)](#) was rejected with the error Cannot convert string '\x--...' from binary to utf16. (Bug #35129361)
- When joining two tables on a string column, and the column from one of the tables has an additional predicate comparing it with a temporal literal, constant propagation in some cases incorrectly caused the join condition to be modified such that it used temporal rather than string semantics when comparing the strings. This caused incorrect results to be returned from the join. (Bug #35115909)
- Error messages returned after calling the [mysql_reset_connection\(\)](#) C API function in a prepared statement did not identify the function name properly. (Bug #35107280)
- Fixed a regression in a previous fix for an issue with windowing functions.
Our thanks to Dmitry Lenev for the contribution. (Bug #35061924)
References: This issue is a regression of: Bug #34572136.
- When replacing subqueries in transforms, the internal flag showing whether a given query block contains any subqueries (`PROP_SUBQUERY`) was not updated afterwards. (Bug #35060385)
- A client setting the character set to an impermissible client character set `ucs2`, `utf16`, `utf16le`, or `utf32` could cause unexpected behavior when the client used an authentication plugin. (Bug #35054579)
- [EXPLAIN ANALYZE](#) displayed 0 when the average number of rows was less than 1. To fix this, we now format numbers in the output of [EXPLAIN ANALYZE](#) and [EXPLAIN FORMAT=TREE](#) such that numbers in the range 0.001-999999.5 are printed as decimal numbers, and numbers outside this range are printed using engineering notation (for example: `1.23e+9`, `934e-6`). In addition, trailing zeroes are no longer printed, and numbers less than `1e-12` are printed as 0.
This helps ensure consistent precision regardless of the number's value and improve readability, while producing minimal rounding errors. (Bug #34990948)
- The [NTILE\(\)](#) function did not work correctly in all cases. (Bug #34986665)
- Some joins on views did not perform correctly. (Bug #34985359)
- Transforming a correlated scalar subquery to a derived table led to a wrong result for [InnoDB](#) tables when the subquery included duplicate predicates. An example of a query which could be affected by this issue is shown here:

```
SELECT * FROM t1
WHERE (
```

```
SELECT t2.a FROM t2
WHERE t2.a = t1.a AND t2.a = t1.a
) > 0;
```

(Bug #34973220)

- Fixed an assert in `sql/item_strfunc.cc` that could potentially lead to issues with the `SPACE0` function. (Bug #34962821)
- Using `ROW_COUNT0` as the `length` argument to `LPAD0` or `RPAD0` did not perform as expected. (Bug #34961236)
- A query with a window function having an expression with a `CASE` function in its `ORDER BY` clause could lead to a server exit. (Bug #34933045)
- The fix for a previous issue introduced an assertion in debug builds when optimizing a `HAVING` clause. (Bug #34923792)
- References: This issue is a regression of: Bug #33725415.
- When using `mysql_multi`, the system that obscures "--password" usage as "--password=*****" would also match "--password-history" and "--password-require-current" definitions as "--password", but now explicitly checks for "--password=" instead. (Bug #34918740)
- In some cases, calling the `mysql_bind_param0` C API function could cause the server to become unresponsive. (Bug #34869076)
- The `authentication_oci_client` plugin was unable to open a valid configuration file if any of its entries contained an equals sign character separated by spaces (for example, `key_file = /home/user/.oci/oci_api_key.pem`). Now, both `'key = value'` and `'key = value'` entry formats are supported. (Bug #34864078)
- Incorrect results were returned when the result of an `INTERSECT` or `EXCEPT` operation was joined with another table. This issue affected these operations in such cases when used with either `DISTINCT` or `ALL`. (Bug #34843764)
- When preparing a view query, the operation used the system character set (instead of the character set stored in data dictionary) and then reported an invalid character-string error. (Bug #34800905)
- Prepared statements that operate on derived tables, including views, could stop unexpectedly due to problems with the code for reopening tables after an error. (Bug #34798403)
- Removed an assertion raised in certain cases by the `RANDOM_BYTES0` function in debug builds. (Bug #34781507)
- There was an issue in how persisted variables were set on startup, causing certain variables not to get properly set to their persisted value. (Bug #34751419)
- The `MAKETIME0` function did not perform correctly in all cases. (Bug #34745241)
- Some functions with multiple arguments did not produce the expected results. (Bug #34741801)
- A table reference in an `ORDER BY` outside the parenthesized query block in which the table was used, and which query block had no `LIMIT` or `ORDER BY` of its own, raised an error. (Bug #34704011)
- References: This issue is a regression of: Bug #103954, Bug #32980249.
- A left join with an impossible condition as part of an `ON` clause was not optimized as in MySQL 5.7, so that in MySQL 8.0, the query executed more quickly without the impossible condition than with it. An example of such a query, impossible condition included, is `SELECT * FROM t1 JOIN t2 ON t1.c1=t2.c1 AND 1=2`. (Bug #34668756)
- When a user defined function was part of a derived table that was merged into the outer query block, or was part of a subquery converted to a semi-join, knowledge of whether this UDF was deterministic (or not) was lost during processing. (Bug #34666531)
- With JSON logging enabled and an event subclass specified in the audit log filter definition, an empty item `{} :` was appended to the end of the logged event. (Bug #34659904)
- Some subqueries did not execute properly. (Bug #34616553)
- After the `asymmetric_encrypt0` component function in a `SELECT` query encountered a NULL field to decrypt, it could return NULL values for other non-NULL encrypted fields. (Bug #34598912)
- The server did not always shut down cleanly after uninstalling the audit log plugin. (Bug #34594035)
- Certain antijoins were not handled correctly by the server. (Bug #34370673)
- References: This issue is a regression of: Bug #30573446.
- When the MySQL 5.7 Optimizer has 2 choices for an index to filter rows, one primary and one secondary, it picks a range scan on the secondary index because the range scan uses more key parts. MySQL 8.0 did not use this logic, instead choosing the primary index to filter rows with `WHERE` clause filtering. Primary key use is not suitable in such cases due to the presence of `LIMIT`, and due to the nature of data distribution. The secondary index was not considered while resolving order by due to constant elimination. This resulted in much different query plans in MySQL 5.7 and MySQL 8.0 for the same query. We solve this issue in MySQL 8.0 by skipping the constant key parts of the index during order-by evaluation only if the query is constant-optimized, which can be done at this time, but not during `LIMIT` analysis. (Bug #34291261)
- The MySQL data dictionary caches failed lookups of `se_private_id` values (IDs which are not found), which speeds up execution of code specific to `InnoDB`, relying on the fact that `InnoDB` does not reuse these IDs. This assumption does not necessarily hold for other storage engines, most notably `NDB`, where this problem was resolved previously by not using this cache. We extend the previous fix made for `NDB` so that the cache lookup is now employed only when the table uses the `InnoDB` storage engine. (Bug #34145006)
- References: See also: Bug #33824058.
- Unexpected results were seen in some queries using `DENSE_RANK0`, possibly with the addition of `WITH ROLLUP`. (Bug #34099408)
- Fixed an assert raised in `sql/sql_tmp_table.cc` following work done previously to reimplement `ROLLUP` processing. (Bug #33830659)
- References: This issue is a regression of: Bug #30969045.
- Some CTEs that did not use any tables were not always handled correctly. (Bug #33725542)
- References: This issue is a regression of: Bug #27062031.
- Accessing rows from a window frame of a window function call present only in the query's `ORDER BY` list raised an error. (Bug #33069747)
- `PERCENT_RANK0` used with `ORDER BY column` did not return the correct result. (Bug #33064174)
- References: This issue is a regression of: Bug #27484133.
- The `--exclude-tables` and `--include-tables` `mysqlpump` options did not handle views. (Bug #21303549)

- Changed the MySQL systemd service unit configuration from `After=network-online.target` to `Wants=network-online.target` to ensure that all configured network devices are available and have an IP address assigned before the service is started. (Bug #109996, Bug #35068274)
- `AVG(...)` OVER (ROWS BETWEEN 1 FOLLOWING AND UNBOUNDED FOLLOWING) did not return the correct result. (Bug #109725, Bug #35013880)
References: This issue is a regression of: Bug #108008, Bug #34431996.
- A query of the form `SELECT 1 FROM t1 WHERE NOT EXISTS (VALUES ROW(1), ROW(2))` caused an assert in debug builds when the `subquery_to_derived` optimizer switch was enabled. (Bug #109723, Bug #35014318)
References: See also: Bug #108910, Bug #34746261.
- `mysqlexport` did not escape reserved word table names when used with the `--delete` option. (Bug #109711, Bug #34999015)
- When cloning a condition to push down to a derived table, characters in strings representing conditions were converted to `utfmb4` correctly only for values less than 128 (the ASCII subset), and code points outside the ASCII subset were converted to invalid characters, causing the resulting character strings to become invalid. For derived tables without `UNION`, this led to problems when a column name from the derived table used characters outside the ASCII subset, and was used in the `WHERE` condition. For derived tables with `UNION`, it created problems when a character outside the ASCII subset was present in a `WHERE` condition. We fix these issues by initializing the string used for representing the condition in such cases to the connection character set. (Bug #109699, Bug #34996488)
- Using `--single-transaction` with `mysqldump` version 8.0.32 required either the `RELOAD` or `FLUSH TABLES` privilege. This requirement now applies only when both `gtid_mode=ON` (default `OFF`) and with `--set-gtid-purged=ON/AUTO` (default `AUTO`). (Bug #109685, Bug #109701, Bug #34993824, Bug #34998910, Bug #35020512)
References: This issue is a regression of: Bug #105761, Bug #33630199.
- Fixed a number of issues present in the internal documentation for the scramble generator algorithm in `sha256_scramble_generator.cc` and `sha2_password_common.cc`.
Our thanks to Niklas Keller for the contribution. (Bug #109576, Bug #34967141)
- `CREATE USER IF NOT EXISTS` added a password history entry even when the user already existed and the password was not updated. This caused a subsequent `ALTER USER` statement to be rejected. (Bug #109415, Bug #34906592)
- Many joins using `eq_ref` access did not perform as well as in previous versions. This issue was first reported in MySQL 8.0.29. (Bug #109361, Bug #34891365)
- A hash outer join sometimes incorrectly matched `NULL` with a decimal zero or an empty string that used a non-padding collation, leading to erroneous results. (Bug #109211, Bug #34837464)
References: This issue is a regression of: Bug #33794977.
- An object used internally by `ALTER INSTANCE RELOAD TLS` was not freed until the number of readers reached 0, under the assumption is that the number of readers should reach 0 fairly frequently. The read lock held during an SSL handshake is generally an expensive operation, with network calls, so when roundtrips between the client and the server took excessively long, the lock was held for a relatively long amount of time. This meant that, when changing the value of this object and there were a sufficient number of incoming SSL connections being made, the number of readers might not reach 0 in a reasonable length of time, leaving the thread holding the lock using 100% of the CPU until the lock was released.
We fix this by adding a wait after setting the pointer to this object to a new value, but before releasing the old object.
Our thanks to Sinisa Milivojevic for the contribution. (Bug #107567, Bug #34284186)
- If `mysqldump` or `mysqldump` could not convert a field's default value to UTF-8 (for instance, if the field was of type `BINARY` and the default value did not coincide with valid UTF-8), the operation produced results that were not valid to import. Further, using the `--hex-blob` option did not resolve the issue. We now convert the default value to the system character set. If this fails, the server sends the value as hexadecimal instead to make it more human-readable. (Bug #104840, Bug #33322551)
- A connection using the C API (`libmysqlclient`) client library could fail with the `FUTURE` crypto policy. (Bug #104778, Bug #33316709)
- While cloning a temporary table for a common table expression which used shared materialization, the cloned temp table was not marked as using hash deduplication, leading to wrong results. We now set the hash field for the cloned temporary table correctly, and update the hidden field count to take this into account. (Bug #102251, Bug #32383712)
References: See also: Bug #103052, Bug #32659629.
- `CREATE EVENT` and `ALTER EVENT` assumed that all values passed to them (other than in a `DO` clause) resolved as scalars without actually checking the values. This led to assertions when any such values actually rows. We now perform an explicit check for the number of columns when resolving such items, and report an error when one produces a row and not a scalar value. (Bug #57552, Bug #11764690)
- A view reference whose underlying field is a constant is not marked as constant when the reference is part of an inner table of an outer join. It was found that, when pushing a condition down to a derived table, the reference was stripped off and only the underlying field was cloned, which made it a constant, and led to wrong results.
To fix this problem, we ensure that we do not push such a condition down to the derived table by adding a check to see first whether the table used by the condition matches the derived table or is a constant expression; only when it is one or the other of these do we actually push the condition down. (Bug #34661, Bug #11747971)

MySQL 8.0.32 릴리즈 노트

MySQL 8.0.32 변동 사항 (2023-01-17 GA)

계정 관리 참고 사항

- The server could return `LDAP_OPERATIONS_ERROR` for LDAP authentication failures, rather than only for actual LDAP server errors such as when an AD domain is not accessible. Now, the server returns `LDAP_AUTHENTICATION_ERROR`, a MySQL-specific error code, to indicate authentication errors. (Bug #100333, Bug #31680279)

C API 참고 사항

- The `sha256_password_auth_client_nonblocking()` function always returned an error, even when the public key was available. (See the MySQL Server Doxygen documentation, available at <https://dev.mysql.com/doc/index-other.html>.) Our thanks to Facebook for the several fixes in this patch. (Bug #34556764)

컴파일 참고 사항

- Microsoft Windows:** The `authentication_idap_sasl` server plugin is no longer built for Windows as only the client is supported for SASL-based LDAP authentication. (Bug #34448155)
- On Windows, compiling MySQL server using VS 2022 would emit an error about two projects named "parser-t" if tests and the NDB storage engine were enabled. The tests were renamed to avoid conflict on case-insensitive operating systems. (Bug #34790413)
- On MacOS, silenced deprecation warnings generated by Xcode 14; this includes suggestions to use `snprintf(3)` instead of `sprintf(3)`, and warnings about possible loss of precision from 64 to 32 bit integers. (Bug #34776172)
- Removed the boost library usage from the plugins. (Bug #34694419)
- Removed all 3rd party files named `'Makefile'` as they were not used. (Bug #34648199)
- Added clang 15 support. (Bug #34638573)
- Located and removed unused code; located it using `fastcov`. (Bug #34583577)
- Improved code related to building the `ndbcluster` plugin by fixing warnings generated with 'gcc 11.2.0 RelWithDebInfo on Ubuntu 22.04' and 'gcc 8.3.1 on el6'. (Bug #34384889)
- Now use full file paths for the Bison and Flex source files to help simplify debugging and gcov reports. (Bug #109022, Bug #34776151)
- Building MySQL would fail if the building user lacked access to the `mysql` temporary directory. Now `--no-defaults` is used when creating the `INFO_BIN` file. (Bug #108947, Bug #34756282)

사용 중단 및 제거된 기능 참고 사항

- Use of the dollar sign (\$) as the first character of an unquoted identifier is now deprecated, raises a warning ([ER_WARN_DEPRECATED_SYNTAX_NO_REPLACEMENT](#)), and is subject to removal in a future release. This affects statements using such an identifier for the name of any database, table, view, column, stored program, or alias. Identifiers beginning with a dollar sign are still permitted only when they are quoted—that is, delimited by single or double quote marks (') or ("), or by backtick characters (`), depending on the server SQL mode. *Example:*

```
mysql> TABLE $t; # Unquoted, produces warning
+-----+
| a |
+-----+
| 1 |
| 2 |
+-----+
2 rows in set, 1 warning (0.00 sec)

mysql> SHOW WARNINGS\G
***** 1. row *****
Level: Warning
Code: 1681
```

```
Message: '$ as the first character of an unquoted identifier' is deprecated and
will be removed in a future release.
1 row in set (0.00 sec)
```

```
mysql> TABLE `t`; # Quoted, no warning
+-----+
| a |
+-----+
| 1 |
| 2 |
+-----+
2 rows in set (0.00 sec)
```

User variables are not affected by this change. For example, the statement `SELECT 1 INTO @$x` does not produce a warning.

See [Schema Object Names](#), for more information. (Bug #34785775, WL #15422)

References: See also: Bug #34684193.

- The `CLIENT_NO_SCHEMA` flag is deprecated. Client programs that specify `CLIENT_NO_SCHEMA` as the `client_flag` argument to `mysql_real_connect()` can now omit the flag and the `db` argument to have the connection set the database value to the current (or default) database. The `libmysqlclient` library now prints a warning on standard error when `mysql_real_connect()` is called with `CLIENT_NO_SCHEMA`. In addition, the server adds a deprecation warning for every non-prepared query executed, if the connection has `CLIENT_NO_SCHEMA`. (WL #13128)
- Previously, legacy compression-control parameters were deprecated and replaced with new configuration parameters for greater control over the use of compression in connections to the server. The new and deprecated parameters are:
 - The `--compression-algorithms` client option enables deprecation of the legacy `--compress` client option and the `Compression` status variable.
 - The `MYSQL_OPT_COMPRESSION_ALGORITHMS` C API option enables deprecation of the legacy `MYSQL_OPT_COMPRESS` C API option.
 - The `MASTER_COMPRESSION_ALGORITHMS` option for the `CHANGE MASTER TO` statement enables deprecation of the legacy `slave_compressed_protocol` system variable.

The deprecated parameters will be removed in a future MySQL version.

Now, the following client programs print a deprecation warning to standard error when a client user invokes one of the programs with `--compress` (or `-C`, if

applicable): **mysqlpump**, **mysqlcheck**, **mysql**, **mysqladmin**, **mysqlbinlog**, **mysqldump**, **mysqlimport**, **mysql** and **mysqltest**.

The **mysqlbackup** `--compress` option has different capabilities and is not deprecated. (WL #13292)

Generated Invisible Primary Keys (GIPK)

- **Replication:** It is now possible to cause a replica to add a generated invisible primary key to any `InnoDB` table that otherwise, as replicated, has no primary key. This is implemented by adding `GENERATE` as a possible value for the `CHANGE REPLICATION SOURCE TO` statement's `REQUIRE_TABLE_PRIMARY_KEY_CHECK` option. `REQUIRE_TABLE_PRIMARY_KEY_CHECK = GENERATE` can be used on a per-channel basis. Suppose the replica is configured with two replication channels, named `ch1` and `ch2`, and you execute the statements shown here:

```
STOP REPLICATION;

SET @@sql_require_primary_key = ON;

CHANGE REPLICATION SOURCE TO
  REQUIRE_TABLE_PRIMARY_KEY_CHECK = GENERATE
  FOR CHANNEL ch1;

START REPLICATION;
```

The effect of these statements is that the replica now adds an invisible primary key for tables whose creation is replicated by channel `ch1`, but does not do so for any keyless table that is created in the context of `ch2`. A replica ignores any setting for `sql_generate_invisible_primary_key` that has been made on the source, and this variable is not replicated. (This behavior is unchanged from previous releases.)

`REQUIRE_TABLE_PRIMARY_KEY_CHECK = GENERATE` is not compatible with MySQL Group Replication, where `ON`, `OFF`, and `STREAM` are the only supported values for this option.

See [CHANGE REPLICATION SOURCE TO Statement](#), and [Generated Invisible Primary Keys](#), for further information. (WL #15419)

키링 참고 사항

- Host names of endpoints specified in `component_keyring_oci` configuration files, and obtained from the Oracle Cloud Infrastructure Console or by querying the Oracle Cloud Infrastructure API, now can retain the `https://` prefix that previously had to be removed when generating a MySQL configuration for the Oracle Cloud Infrastructure Vault keyring component. (Bug #34636297)

플러그인 인증

- On Windows, the client-side Kerberos authentication plugin now supports GSSAPI through the MIT Kerberos library. It is possible to choose between SSPI and GSSAPI at runtime using a new plugin option supported by the `authentication_kerberos_client` authentication plugin on Windows. Client users invoke **mysql** or **mysqldump** with the `--plugin-authentication-kerberos-client-mode` command-line option to set the mode to GSSAPI. The default mode of the `authentication_kerberos_client` plugin is SSPI, previously the only authentication method on Windows. For more information, see Connection Commands for Windows Clients in GSSAPI Mode. (WL #15336)

공간 데이터 지원

- The MySQL `ST_Transform()` function now supports all Cartesian projections, with the exceptions of EPSG 1042 (Krovak Modified), EPSG 1043 (Krovak Modified (North Orientated)), EPSG 9816 (Tunisia Mining Grid), and EPSG 9826 (Lambert Conic Conformal (West Orientated)). (Bug #27272733, Bug #34495023, WL #15164)

SQL 문법 참고 사항

- Important Change:** Previously, MySQL supported the use of “full” as the name of a table, column, view, stored procedure, or stored function, as well as for the alias of a table, view, or column. Beginning with this release, using “full” (regardless of letter case) in this fashion as an unquoted identifier is now deprecated, and raises a warning. This is to align more closely with the SQL standard, in which `FULL` is reserved as a keyword. For example, the following [CREATE TABLE](#) and [DROP TABLE](#) statements now raise warnings, as shown here:

```
mysql> CREATE TABLE full (c1 INT, c2 INT);
Query OK, 0 rows affected, 1 warning (0.03 sec)

mysql> SHOW WARNINGS\G
***** 1. row *****
Level: Warning
Code: 4119
Message: Using FULL as unquoted identifier is deprecated, please use quotes or
rename the identifier.
1 row in set (0.00 sec)

mysql> DROP TABLE full;
Query OK, 0 rows affected, 1 warning (0.01 sec)

mysql> SHOW WARNINGS\G
***** 1. row *****
Level: Warning
Code: 4119
Message: Using FULL as unquoted identifier is deprecated, please use quotes or
rename the identifier.
1 row in set (0.00 sec)
```

To execute the statements without the warnings, encase the table name in each of them with backtick characters (```), like this:

```
mysql> CREATE TABLE `full` (c1 INT, c2 INT);
Query OK, 0 rows affected (0.03 sec)

mysql> DROP TABLE `full`;
Query OK, 0 rows affected (0.02 sec)
```

The use of `FULL` in the right hand side of a value assignment *is not* affected by this change. For example,

the [SET](#) statement shown here remains valid:

```
mysql> SHOW VARIABLES LIKE '%metadata%';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| binlog_row_metadata | MINIMAL |
| innodb_stats_on_metadata | OFF |
| performance_schema_max_metadata_locks | -1 |
| resultset_metadata | FULL |
+-----+-----+
4 rows in set (0.00 sec)

mysql> SET @@global.binlog_row_metadata=FULL;
Query OK, 0 rows affected (0.00 sec)

mysql> SHOW VARIABLES LIKE '%metadata%';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| binlog_row_metadata | FULL |
| innodb_stats_on_metadata | OFF |
| performance_schema_max_metadata_locks | -1 |
| resultset_metadata | FULL |
+-----+-----+
4 rows in set (0.00 sec)
```

For more information, see [Keywords and Reserved Words](#). (WL #15241)

- It is now possible to set the default format for the output of any [EXPLAIN](#) statement which obtains a query execution plan, and which has no [FORMAT](#) option, using the [explain_format](#) system variable added in this release. Like the [FORMAT](#) option, this variable can take any of the values [TRADITIONAL](#), [JSON](#), or [TREE](#). [DEFAULT](#) is also supported as a synonym for [TRADITIONAL](#). ([DEFAULT](#) is *not* supported with the [FORMAT](#) option for [EXPLAIN](#).) Suppose the value of [explain_format](#) is [TREE](#); in this case, the output from any such [EXPLAIN](#) statement uses the tree-based format, as though [FORMAT=TREE](#) had been specified as part of the [EXPLAIN](#) statement.

Any value set for [explain_format](#) is overridden by a [FORMAT](#) option. This means that, if [explain_format](#) is set to [TREE](#), supplying [FORMAT=JSON](#) when invoking [EXPLAIN](#) causes the value of [explain_format](#) to be ignored, and the result is displayed using the [JSON](#) format.

[explain_format](#) also affects the behavior of [EXPLAIN ANALYZE](#); since this statement supports only the [TREE](#) format, if the value of [explain_format](#) is not [TREE](#), this means that any [EXPLAIN ANALYZE](#) statement that does not specify the [TREE](#) format explicitly raises the error 'This version of MySQL doesn't yet support 'EXPLAIN ANALYZE with *format* format'.

The new system variable has both global and session scope, can be persisted, and can be set from the command line (as `--explain-format`) or in a `my.cnf` option file.

See the description of [explain_format](#). See also [Obtaining Execution Plan Information](#), and [Obtaining Information with EXPLAIN ANALYZE](#), for further information and examples. (WL #15040)

References: See also: Bug #33629360.

쓰레드 풀 참고 사항

- Whenever a connection was terminated due to inactivity, the thread pool plugin printed only a generic message about connections timing out; this often made analysis of such timeouts more difficult than necessary. A new `INFO_LEVEL` message makes it clear that a connection has been terminated due to inactivity in the thread pool, as well as which timeout value was used to make this determination. (Bug #34767607)
- Two columns added to the Performance Schema `tp_thread_state` table in this release make it possible to identify a thread's type, and to map threads in this table to those in the Performance Schema `threads` table. The type of thread is now shown in the `tp_thread_state` table's `TP_THREAD_TYPE` column, and the thread's unique ID in the `THREAD_ID` column. For more information, see [The tp_thread_state Table](#). (Bug #34020058)

추가 또는 변경된 기능

- Important Change:** For platforms on which OpenSSL libraries are bundled, the linked OpenSSL library for MySQL Server has been updated to version 1.1.1s. Issues fixed in OpenSSL version 1.1.1s are described at <https://www.openssl.org/news/C111.txt>. (Bug #34828308)
- Binary packages that include curl rather than linking to the system curl library have been upgraded to use curl 7.86.0. (Bug #34828111)
- The internal resource-group enhancement added in MySQL 8.0.31 is refactored, but it continues to support

the [Resource_group_supported](#) status variable. (Bug #34702833, Bug #34699751)
References: Reverted patches: Bug #34264356.

버그 수정

- **Important Change:** The implementation of the [max_join_size](#) system variable, although documented as a maximum number of rows or disk seeks, did not check the number of rows or disk seeks directly, but instead treated [max_join_size](#) as the maximum estimated cost to permit. While cost and row count are correlated, they are not the same, and this could lead to unexpected results when some large queries were allowed to proceed.
In this release, we change how [max_join_size](#) is used, so that it now actually limits the maximum number of row accesses in base tables. If the estimate indicates that a greater number of rows must be read from the base tables, an error is raised. This makes the actual behavior better reflect what is documented. (Bug #83885, Bug #25118903)
- **InnoDB:** Undetectable problems after upgrade from 8.0.28, crash and corruption.
Any new row inserted after upgrade will have all columns added with [ALGORITHM=INSTANT](#) materialized and have version=0.
In the new implementation, a column added with [ALGORITHM=INSTANT](#) will fail if the maximum possible size of a row exceeds the row size limit. So new rows with materialized [ALGORITHM=INSTANT](#) columns will always be within row size limit. (Bug #34558510)
- **InnoDB:** No more garbled UTF characters in SHOW ENGINE INNODB STATUS (Bug #34486877, Bug #108111)
- **InnoDB:** Alter handler adjusted so that adding more than 1024 columns with [ALGORITHM=INSTANT](#) no longer causes a crash. (Bug #34378513, Bug #107854)
- **InnoDB:** After a column added with [ALGORITHM=INSTANT](#), an online rebuild DDL no longer crashes. (Bug #33788578, Bug #106279)
- **InnoDB:** Several adaptive hash index (AHI) code optimizations and improvements were implemented, addressing various issues including potential race conditions. (Bug #33601434)
- **Replication:** When [SOURCE_HEARTBEAT_PERIOD](#) was set to a very small value (such as 1 microsecond) on the server using [CHANGE REPLICATION SOURCE TO](#), and the [mysqlbinlog](#) client program was started with [--read-from-remote-server](#) and [--stop-never=1](#), it was possible for the binary log dump thread to send an EOF packet to the client before all events had been sent. (Bug #34860923)
- **Replication:** Removed an assert from [sql/rpl_group_replication.cc](#) which triggered a false error in testing. (Bug #34619134)
- **Replication:** After MySQL was started with [--server-id=0](#), trying to change the server ID by using [SET PERSIST server_id=N](#) (where N is an integer greater than zero) and restarting the server had the following results:
 - [SELECT @@server_id](#) returned [N](#).
 - Any replication SQL statement such as [START REPLICA](#) was rejected with [ER_SLAVE_CONFIGURATION](#).

To fix this problem, we now ensure that such checks use the value of the server variable rather than the value passed to the startup option. (Bug #34412816)

- **Replication:** When replicating compressed binary log events generated by the [NDB](#) binary log injector, relay log positions were not updated in the multithreaded applier, thus causing replication to hang. (Bug #33889030)
References: See also: Bug #33784241.
- **Replication:** Issuing [STOP REPLICA SQL THREAD](#) while the SQL thread was handling a transaction caused replication to stop immediately, instead of waiting 60 seconds for the event group to complete before shutting down the SQL thread as expected.
The root cause of this issue was due to the internal variable storing the last event start time not being reset after the SQL thread was restarted.
We fix this by resetting the variable holding the last event start time whenever the SQL thread is started. (Bug #33646899)
- **Replication:** While their wording might imply otherwise, the log messages [Setting super_read_only=ON](#) ([ER_GRP_RPL_SUPER_READ_ON](#)) and [Setting super_read_only=OFF](#) ([ER_GRP_RPL_SUPER_READ_OFF](#)) were written only after the operations were attempted, and not beforehand, or while the operations were ongoing. This sometimes led to confusion when setting the variable was rejected, and this was logged prior to the set attempt itself being logged. To keep this from happening, these messages are now logged just prior to attempting the operation. (Bug #108843, Bug #34728079)
- **Replication:** The [relay_log_space_limit](#) system variable is a 64-bit value, but its valid maximum was specified internally as that of a 32-bit value. (Bug #106323, Bug #33799840)
- **Replication:** Eliminated an unnecessary update of the [gtid_executed](#) table which was performed when rotating the binary logs. (Bug #106116, Bug #33759477)
- **Group Replication:** The [WRITE_CONSENSUS_SINGLE_LEADER_CAPABLE](#) column of the MySQL Performance Schema's [replication_group_communication_information](#) table reflects the runtime value of a Paxos Single Leader setup in a group, letting users know what the value of [group_replication_paxos_single_leader](#) must be on joining members. A group that was bootstrapped with single-leader enabled but with its protocol version downgraded to one that did not support it reported [WRITE_CONSENSUS_SINGLE_LEADER_CAPABLE](#) equal to 0, as expected, but attempting to join an instance to the group using [group_replication_paxos_single_leader=0](#) was not possible. To solve this problem, we change the behaviour and make the value of [group_replication_paxos_single_leader](#) consistent with the communication version that the group is running. Since this variable was introduced in MySQL 8.0.27, it is not known or used in any previous version, and so we now enforce the following rules:
 - When a node tries to join a group that is running MySQL 8.0.26 or earlier and we are version 8.0.27 or later, we reject the attempt with an error stating that [group_replication_paxos_single_leader](#) must be [OFF](#) before joining the group
 - When we try to use [group_replication_set_communication_protocol\(\)](#) to set a version less than 8.0.27 and we are of version 8.0.27 or later, we reject the function call if [group_replication_paxos_single_leader](#) is not [OFF](#).

In addition, we also change the value checked to determine whether changing the group leader is allowed after running `group_replication_set_communication_protocol()`. Previously, this was the runtime value of `group_replication_paxos_single_leader`, which takes effect only after a group reboot. Instead, when we run `group_replication_set_communication_protocol()`, we now use the value shown by the `replication_group_communication_information` table's `WRITE_CONSENSUS_SINGLE_LEADER_CAPABLE` column, described previously. (Bug #34555045, Bug #34828311)

- **Group Replication:** In a 3 node cluster, all nodes were killed due to running out of memory. Subsequently, after all nodes were restarted successfully, attempting to bring the cluster back online caused the node that had been the primary to hang.

For more information, see [Rebooting a Cluster from a Major Outage](#) (Bug #108339, Bug #34564856)

- **Group Replication:** When a group was run with `group_replication_consistency = AFTER` and a secondary failed due to external conditions such as an unstable network, the secondary could sometimes encounter the error Transaction 'GTID' does not exist on Group Replication consistency manager while receiving remote transaction prepare.

The root cause of this issue was that the primary might log out of order the `View_change_log_event` with which the secondary rejoined; when the secondary used the primary as the group donor, this could cause the secondary to catch up with the group improperly and, eventually, generate incorrect GTIDs for the group transactions. The group replication primary ensures that the `View_change_log_event` is logged after all preceding transactions, but there was a window during which transactions ordered after the `View_change_log_event` on the group global order could be logged before the event.

To solve this issue, we now make sure that transactions ordered before a view are always logged before the `View_change_log_event`, and that transactions ordered after a view are always logged after this event. This is now done by the binary log ticket manager, which guarantees the order in which transactions in the binary log group commit are committed. (Bug #104980, Bug #33405699)

References: See also: Bug #34746357.

- **Microsoft Windows:** When compiling MySQL on Windows platforms, the CMake `-DWITH_WIN_JEMALLOC` option was not always handled correctly. (Bug #108341, Bug #34698376)
- **JSON:** While saving the result of `JSON_ARRAYAGG()` or `JSON_OBJECTAGG()` in a column, the data type information was lost due to the result being an item of type `SUM_FUNC_ITEM`. To fix this, we remove the type check and this way retain the original type information. (Bug #108326, Bug #34548259)
- Some remote connections to the server were not handled correctly. This issue arose as the result of a previous fix for an issue with `require_secure_transport`. (Bug #34857411)
References: This issue is a regression of: Bug #34094706.
- Some query plans were not stable due to nondeterministic sorting of `Key_use_array` in `sql_optimizer.cc`; now we sort it with `std::stable_sort()` instead of `std::sort()`. (Bug #34823952)
References: This issue is a regression of: Bug #25965593.
- Binary packages that include OpenLDAP rather than linking to the system OpenLDAP library were upgraded to use version 2.5.13. (Bug #34815046)
- In some cases, an unexpected packet sent by the server to a MySQL client program during authentication could result in an infinite loop. (Bug #34805922)
- GIS data was not always handled correctly in windowing functions. (Bug #34778646)
- A thread remained bound to the CPU of a dropped resource group even after it was assigned to the user default resource group (`USR_default`). `USR_default` has 0 CPU priority and no CPU affinity, so with this fix, the thread now is able to run any CPU with `USR_default`. (Bug #34748973)
- With JSON logging enabled, calling the `audit_log_rotate()` function did not rotate the file as expected. A rotated file name consists of the timestamp from the last event logged into the file. When the file is empty, last timestamp is the same as the timestamp in the already created file. To fix this issue, the function now uses the current time to name the file if the file is empty. (Bug #34733508)
- Some queries having multiple lateral derived tables did not produce the expected result. (Bug #34716246)
- The bundled zlib library has been upgraded to zlib 1.2.13; zlib 1.2.13 is now the minimum zlib version supported. (Bug #34711762, Bug #34711758)
- Certain `INTERSECT` queries were not handled correctly. (Bug #34642435)
- Using the `MAX_EXECUTION_TIME` optimizer hint with a value greater than the stated maximum kept an upgrade to MySQL 8.0.30 from completing; this caused the server to report a warning which was interpreted by the upgrade process as an unrecoverable error. (Bug #34607401)
- In certain cases, evaluation of window functions was not performed correctly. (Bug #34572136)
References: This issue is a regression of: Bug #32644631, Bug #32802301.
- Some CTEs were not processed correctly. (Bug #34572040, Bug #34634469)
References: This issue is a regression of: Bug #33856374.
- Values returned from functions or operators that convert a value to `FLOAT` (`CAST(... AS FLOAT)`, `CONVERT(..., FLOAT)`, `JSON_VALUE(... RETURNING FLOAT)`) may have extra precision in their internal representation, since they are stored in double precision internally. This sometimes caused unexpected results when checking such values for equality, such as `SELECT DISTINCT` returning duplicates and comparison operators incorrectly reporting two equal values as unequal.
We fix this problem by stripping off the extra double precision from the values before returning them, and by making any conversion from float to string in these conversion operators use float format instead of double format. (Bug #34554755)
- Removed an assertion in `query_expression::assert_not_fully_clean()`. (Bug #34526104)
- Upgrading from MySQL 5.7 to MySQL 8.0 with a very large number of tables in a single database caused the server to consume excessive memory. It was found that, during the process of checking whether tables could be upgraded, we fetched all the data dictionary `Table` objects upfront, processing each and fetching its name, then performed `CHECK TABLE ... FOR UPGRADE` on the list. Fetching all objects beforehand was not necessary in this case, and contributed greatly to memory consumption.
To correct this problem, we now fetch one `Table` object at a time in such cases, performing any required checks, fetching its name, and releasing the object, before proceeding with the next one. (Bug #34526001)
- When creating natural join columns, a hidden column added as part of a materialized derived table is used in constructing the join condition, which is later used to check whether the join eligible for pushdown to derived tables. The current issue arose when this column was not retrieved from the derived table due to it being hidden; this occurred even when the condition pushdown optimization was not enabled. We solve this problem by rejecting all hidden columns added internally, and not merely for hidden columns added for functional indexes. (Bug #34523627)

- Types were not derived consistently from user variables. This could be seen, for example, by executing the following statements repeatedly:

```
CREATE TABLE t AS SELECT @max_error_count UNION SELECT 'a';

SHOW CREATE TABLE t;
```

In this particular case, the output of the `SHOW CREATE TABLE` statement showed `@max_error_count` text`` the first time, and `@max_error_count` mediumblob`` in successive iterations. (The second is correct.) (Bug #34523475)

- Following work done in MySQL 8.0.23 to improve host resolution for user accounts, the time needed for `CREATE USER` to complete increased significantly, particularly when running many such statements in close succession.

Prior to upgrading to this release, you can work around this issue when issuing many such statements in succession by preceding them with a single `CREATE USER 'fakeuser' ACCOUNT LOCK` (you can use any user name that does not conflict with existing ones for this). When finished, you can (and should) clean up by issuing the following statements:

```
DROP USER 'fakeuser';
FLUSH PRIVILEGES;
```

For more information, see [Access Control, Stage 1: Connection Verification](#) (Bug #34449016)

- The `data_masking` server-side plugin could emit a runtime error and halt unexpectedly. (Bug #34445632)
- Some multiply nested queries were not performed correctly. (Bug #34377854)
- When merging a derived table, a nested join condition is added to the derived table and the underlying tables are added to this join nest. In addition, the join condition is associated with the derived table. Evaluation for range access is skipped if the table is an inner table of an outer join or if the table is an inner table and the join is not a semijoin. For a derived table, the underlying base table was treated as being of the latter kind, range analysis was skipped, and the range access method was thus unavailable. To fix this problem, we now evaluate for range access when the embedding table is a derived table, and ensure that the join condition associated with the derived table is used for the range optimization. (Bug #34347116)
- A `LOAD DATA INFILE` statement issued with a subquery could cause the server to return an incorrect warning (`Subquery returns more than 1 row`). (Bug #34336033)
- Improved handling of resource allocation for internal temporary tables. (Bug #34174001)
- A specific column added after a drop using the `INSTANT` algorithm could cause a data error and a server exit. (Bug #34122122)
- A query such as `SELECT 1 AS one FROM t WHERE 1=(SELECT 1 UNION SELECT 2)` is transformed to this:

```
SELECT 1 AS one
FROM t
JOIN ( SELECT 1 AS col1 UNION SELECT 2 ) derived
WHERE 1 = derived.col1;
```

The optimizer in this case pushed down `1 = derived.col1` into the union, removing the contribution from `SELECT 2`, which led to an erroneous result. We now no longer push the condition down in such cases. (Bug #33910786)

- Some parenthesized query expressions with either or both of `ORDER BY` and `LIMIT` were not always handled correctly. (Bug #33725530)
- Reimplemented retention of item trees having multiple references, by using a reference count in every `Item` object. Also removed old code no longer needed due to this change. (Bug #33725415)
- An assert occurred in `InnoDB` during upgrade of the data dictionary when the definition of the `innodb_ddl_log` table changed, even when such changes were effectively null operations, such as updating `utf8` and `utf8_bin` in table and column definitions to `utf8mb3` and `utf8mb3_bin`, respectively. (Bug #33688509)

References: This issue is a regression of: Bug #33787300.

- Data and GTIDs backed up by `mysqldump` were inconsistent when the options `--single-transaction` and `--set-gtid-purged=ON` were both used. This was because, between the start of the transaction by `mysqldump` and the fetching of `GTID_EXECUTED`, GTIDs on the server could have increased already. With this fix, a `FLUSH TABLES WITH READ LOCK` is performed before fetching `GTID_EXECUTED`, to ensure that its value is consistent with the snapshot taken by `mysqldump`.

Our thanks to Marcelo Altmann for the contribution.

Limitation: This fix adds a requirement for the `RELOAD` privilege when using `--single-transaction` to execute `FLUSH TABLES WITH READ LOCK`; the MySQL team is investigating a solution. (Bug #33630199, Bug #105761)

- `SET PERSIST` accepted dot-separated names of variables registered by components as well as `MyISAM` multiple key cache variables, but `RESET PERSIST` rejected the same names with a syntax error. To fix this discrepancy, we add support in this to `RESET PERSIST` for variable names containing a dot character (`.`). (Bug #33417357)
- Some grouped queries were not always handled correctly. (Bug #33294005, Bug #33349994)
- When multifactor authentication used the `auth_socket` authentication plugin for the first factor, the server executed the wrong code during the second-factor authentication workflow and returned an error message. The second factor could be any authentication plugin. (Bug #33192223)
- Use of a wild card as a column identifier in an `INSERT` statement was allowed by the parser even though the syntax is not supported, which led to an assert in debug builds and a silent rejection of the statement in release builds. This construction has been removed as a possibility from the grammar, and is now handled strictly as a syntax error. (Bug #33142665)

References: This issue is a regression of: Bug #30528450.

- In prepared statements, some types of subqueries could cause a server exit. (Bug #33100586)
- Some floating-point literals were not always handled correctly. (Bug #32824429)
- A `DELETE` statement with a table alias could result in an intermittent server exit. (Bug #32762229)
- Moved `INFO_SRC` and `INFO_BIN` from the `mysql-common` package to the `mysql-community-server-core` package, the same package as `mysqld` and more consistent with RPM packaging. (Bug #32752147)

- Some queries making use of `MATCH()` within a `HAVING` clause were not handled correctly. (Bug #32616816, Bug #32934558, Bug #34782389)
- A `CREATE VIEW` statement that contained a subquery sometimes led to an assertion in debug builds. (Bug #108783, Bug #34703610)
- The deduction of data types for dynamic parameters passed as parameters to a user-defined SQL function was correct only for a single parameter; with more than one parameter, no such deduction was performed for the second and following parameters, with the result that their types were always reported erroneously to clients as `MYSQL_TYPE_INVALID`. (Bug #108545, Bug #34629157)
- The internal function `clone_os_copy_file_to_buf()` did not advance the buffer position in the event of a partial read. Our thanks to Laurynas Biveinis for the contribution. (Bug #108317, Bug #34543194)
- Views that access system views could encounter an access-denied error during normal use if the pushdown condition included expressions that used native functions from the system view. (Bug #108202, Bug #34515868)
- When using window functions, the current row could reevaluate itself based on the wrong record in some cases. (Bug #108008, Bug #34431996)
- A condition pushdown into a `UNION` of queries having `LIKE` clauses did not preserve the correct character set, leading to an (erroneous) empty result.

We solve this problem in two parts:

1. By refactoring resolution of `LIKE` expressions, in which character set determination and propagation were previously performed in two separate blocks of the code that were not always consistent with one another.
2. By adding, in the internal `parse_expression()` function, a character set prefix to any literal character string that is cloned.

(Bug #107787, Bug #34359297, Bug #34589153)

- The `audit_log` server-side plugin always logged an entire multiple query, rather than logging only the specific part of the query that was executed. Changing when the query length is set resolves the issue. (Bug #107390, Bug #34207811)
- Following an upgrade to MySQL 8.0.27 a specific query started consuming comparatively high amounts of memory whenever it was run within a stored procedure. (Bug #107327, Bug #34190122)
- When a **mysqld** startup option was used with the `maximum-` prefix, the upper bound for the corresponding system variable was set but its current value was not checked against or adjusted according to the new limit and thus could in some cases be greater than the stated maximum. We fix this by adjusting the current value if it is larger than the new user defined maximum value. (Bug #99029, Bug #31072098)
- The `mysql_stmt_close()` C API function could stop responding after a prepared statement was canceled using `KILL QUERY`. (Bug #84470, Bug #25584097)