aws

Catatan Rilis untuk Aurora MySQL

Amazon Aurora

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon Aurora: Catatan Rilis untuk Aurora MySQL

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Merek dagang dan tampilan dagang Amazon tidak boleh digunakan sehubungan dengan produk atau layanan apa pun yang bukan milik Amazon, dengan cara apa pun yang dapat menyebabkan kebingungan di antara para pelanggan, atau dengan cara apa pun yang menghina atau mendiskreditkan Amazon. Semua merek dagang lain yang tidak dimiliki oleh Amazon merupakan hak milik masing-masing pemiliknya, yang mungkin atau mungkin tidak terafiliasi, terkait dengan, atau disponsori oleh Amazon.

Table of Contents

Catatan rilis Aurora MySQL 1
Kalender rilis MySQL Aurora 2
Kalender rilis untuk Aurora MySQL versi utama 2
Kalender rilis untuk Aurora MySQL versi minor 4
Aurora MySQL versi 3 6
Pembaruan Aurora MySQL: 2026-02-17 (versi 3.12.0, kompatibel dengan MySQL 8.0.44) 7
Perbaikan 8
Integrasi perbaikan bug MySQL Community Edition 12
Pembaruan Aurora MySQL: 2025-12-16 (versi 3.11.1, kompatibel dengan MySQL 8.0.43) 12
Perbaikan 13
Pembaruan Aurora MySQL: 2025-11-13 (versi 3.11.0, kompatibel dengan MySQL 8.0.43) 13
Perbaikan 14
Integrasi perbaikan bug MySQL Community Edition 18
Pembaruan Aurora MySQL: 2026-01-02 (versi 3.10.3, kompatibel dengan MySQL 8.0.42) 18
Perbaikan 19
Pembaruan Aurora MySQL: 2025-11-20 (versi 3.10.2, kompatibel dengan MySQL 8.0.42) 19
Perbaikan 20
Pembaruan Aurora MySQL: 2025-09-30 (versi 3.10.1, kompatibel dengan MySQL 8.0.42) 21
Perbaikan 22
Integrasi perbaikan bug MySQL Community Edition 24
Pembaruan Aurora MySQL: 2025-07-31 (versi 3.10.0, kompatibel dengan MySQL 8.0.42) 24
Fitur baru 25
Perbaikan 26
Integrasi perbaikan bug MySQL Community Edition 28
Pembaruan Aurora MySQL: 2025-05-14 (versi 3.09.0, kompatibel dengan MySQL 8.0.40) 28
Sorotan rilis 29
Perbaikan 30
Integrasi perbaikan bug MySQL Community Edition 33
Pembaruan Aurora MySQL: 2025-04-07 (versi 3.08.2, kompatibel dengan MySQL 8.0.39) 33
Perbaikan 34
Integrasi perbaikan bug MySQL Community Edition 35
Pembaruan Aurora MySQL: 2025-02-17 (versi 3.08.1, kompatibel dengan MySQL 8.0.39) 35
Perbaikan 34
Integrasi perbaikan bug MySQL Community Edition 35

Pembaruan Aurora MySQL: 2024-11-18 (versi 3.08.0, kompatibel dengan MySQL 8.0.39) 37
Fitur baru 37
Perbaikan 39
Integrasi perbaikan bug MySQL Community Edition 44
Pembaruan Aurora MySQL: 2024-07-23 (versi 3.07.1) (Usang) 44
Perbaikan 45
Integrasi perbaikan bug MySQL Community Edition 46
Pembaruan Aurora MySQL: 2024-06-04 (versi 3.07.0) (Usang) 46
Perbaikan 46
Integrasi perbaikan bug MySQL Community Edition 50
Pembaruan Aurora MySQL: 2024-06-26 (versi 3.06.1) (Usang) 51
Perbaikan 52
Integrasi perbaikan bug MySQL Community Edition 54
Pembaruan Aurora MySQL: 2024-03-07 (versi 3.06.0) (Usang) 54
Fitur baru 55
Perbaikan 52
Integrasi perbaikan bug MySQL Community Edition 54
Pembaruan Aurora MySQL: 2024-01-31 (versi 3.05.2) (Usang) 60
Perbaikan 60
Integrasi perbaikan bug MySQL Community Edition 62
Pembaruan Aurora MySQL: 2023-11-21 (versi 3.05.1) (Usang) 62
Perbaikan 63
Integrasi perbaikan bug MySQL Community Edition 63
Pembaruan Aurora MySQL: 2023-10-30 (versi 3.05.0.1) (Usang) 63
Perbaikan 64
Pembaruan Aurora MySQL: 2023-10-25 (versi 3.05.0) (Usang) 64
Perbaikan 65
Integrasi perbaikan bug MySQL Community Edition 70
Pembaruan Aurora MySQL: 2026-01-02 (versi 3.04.6, kompatibel dengan MySQL 8.0.28) 70
Perbaikan 71
Integrasi perbaikan bug MySQL Community Edition 71
Pembaruan Aurora MySQL: 2025-05-05 (versi 3.04.4, kompatibel dengan MySQL 8.0.28) 71
Perbaikan 72
Integrasi perbaikan bug MySQL Community Edition 75
Pembaruan Aurora MySQL: 2024-06-26 (versi 3.04.3, kompatibel dengan MySQL 8.0.28) 75
Perbaikan 76

Integrasi perbaikan bug MySQL Community Edition 78
Pembaruan Aurora MySQL: 2024-03-15 (versi 3.04.2, kompatibel dengan MySQL 8.0.28) 78
Perbaikan 79
Integrasi perbaikan bug MySQL Community Edition 81
Pembaruan Aurora MySQL: 2023-11-13 (versi 3.04.1, kompatibel dengan MySQL 8.0.28) 82
Perbaikan 83
Integrasi perbaikan bug MySQL Community Edition 84
Pembaruan Aurora MySQL: 2023-07-31 (versi 3.04.0, kompatibel dengan MySQL 8.0.28) 85
Perbaikan 86
Integrasi perbaikan bug MySQL Community Edition 92
Pembaruan Aurora MySQL: 2023-12-08 (versi 3.03.3,) (Usang) 92
Perbaikan 93
Integrasi perbaikan bug MySQL Community Edition 95
Pembaruan Aurora MySQL: 2023-08-29 (versi 3.03.2) (Usang) 95
Perbaikan 96
Integrasi perbaikan bug MySQL Community Edition 97
Pembaruan Aurora MySQL: 2023-05-11 (versi 3.03.1) (Usang) 98
Perbaikan 98
Integrasi perbaikan bug MySQL Community Edition 100
Pembaruan Aurora MySQL: 2023-03-01 (versi 3.03.0) (Usang) 101
Perbaikan 101
Integrasi perbaikan bug MySQL Community Edition 104
Pembaruan Aurora MySQL: 2023-04-17 (versi 3.02.3) (Usang) 105
Perbaikan 106
Pembaruan Aurora MySQL: 2022-11-18 (versi 3.02.2) (Usang) 107
Perbaikan 107
Integrasi perbaikan bug MySQL Community Edition 104
Pembaruan Aurora MySQL: 2022-09-07 (versi 3.02.1) (Usang) 110
Perbaikan 111
Pembaruan Aurora MySQL: 2022-04-20 (versi 3.02.0) (Usang) 112
Perbaikan 113
Integrasi perbaikan bug MySQL Community Edition 115
Pembaruan Aurora MySQL: 2022-04-15 (versi 3.01.1) (Usang) 115
Perbaikan 116
Integrasi perbaikan bug MySQL Community Edition 118
Pembaruan Aurora MySQL: 2021-11-18 (versi 3.01.0) (Usang) 118

Perbaikan 119
Aurora MySQL versi 2 120
Pembaruan Aurora MySQL: 2025-04-09 (versi 2.12.5, kompatibel dengan MySQL 5.7.44) - Versi Dukungan Diperpanjang RDS 122
Perbaikan 123
Pembaruan Aurora MySQL: 2024-10-23 (versi 2.12.4, kompatibel dengan MySQL 5.7.44) - Versi Dukungan Diperpanjang RDS 124
Perbaikan 123
Integrasi perbaikan bug MySQL Community Edition 125
Fitur yang tidak didukung di Aurora MySQL versi 2 125
Kompatibilitas MySQL 5.7 125
Pembaruan Aurora MySQL: 2024-07-09 (versi 2.12.3, kompatibel dengan MySQL 5.7.44) - Versi Dukungan Diperpanjang RDS 126
Perbaikan 127
Integrasi perbaikan bug MySQL Community Edition 129
Fitur yang tidak didukung di Aurora MySQL versi 2 130
Kompatibilitas MySQL 5.7 130
Pembaruan Aurora MySQL: 2024-03-19 (versi 2.12.2, kompatibel dengan MySQL 5.7.44) - Versi Dukungan Diperpanjang RDS 130
Perbaikan 131
Integrasi perbaikan bug MySQL Community Edition 132
Fitur yang tidak didukung di Aurora MySQL versi 2 132
Kompatibilitas MySQL 5.7 133
Pembaruan Aurora MySQL: 2023-12-28 (versi 2.12.1, kompatibel dengan MySQL 5.7.40) - Versi Dukungan Diperpanjang RDS 133
Perbaikan 134
Integrasi perbaikan bug MySQL Community Edition 137
Fitur yang tidak didukung di Aurora MySQL versi 2 137
Kompatibilitas MySQL 5.7 137
Pembaruan Aurora MySQL: 2023-10-25 (versi 2.12.0.1, kompatibel dengan MySQL 5.7.40) - Versi Dukungan Diperpanjang RDS (Beta) 138
Perbaikan 138
Pembaruan Aurora MySQL: 2023-07-25 (versi 2.12.0, kompatibel dengan MySQL 5.7.40) - Versi Dukungan Diperpanjang RDS 139
Perbaikan 139
Integrasi perbaikan bug MySQL Community Edition 143

Fitur yang tidak didukung di Aurora MySQL versi 2 143
Kompatibilitas MySQL 5.7 143
Pembaruan Aurora MySQL: 2024-07-19 (versi 2.11.6, kompatibel dengan MySQL 5.7.12) - Versi Dukungan Diperpanjang RDS 144
Perbaikan 145
Integrasi perbaikan bug MySQL Community Edition 147
Fitur yang tidak didukung di Aurora MySQL versi 2 147
Kompatibilitas MySQL 5.7 147
Pembaruan Aurora MySQL: 2024-03-26 (versi 2.11.5, kompatibel dengan MySQL 5.7.12) - Versi Dukungan Diperpanjang RDS (Default) 148
Perbaikan 149
Integrasi perbaikan bug MySQL Community Edition 150
Fitur yang tidak didukung di Aurora MySQL versi 2 150
Kompatibilitas MySQL 5.7 151
Pembaruan Aurora MySQL: 2023-10-17 (versi 2.11.4, kompatibel dengan MySQL 5.7.12) - Versi Dukungan Diperpanjang RDS 151
Perbaikan 152
Integrasi perbaikan bug MySQL Community Edition 154
Fitur yang tidak didukung di Aurora MySQL versi 2 155
Kompatibilitas MySQL 5.7 155
Pembaruan Aurora MySQL: 2023-06-09 (versi 2.11.3, kompatibel dengan MySQL 5.7.12) - Versi Dukungan Diperpanjang RDS 155
Perbaikan 157
Fitur yang tidak didukung di Aurora MySQL versi 2 158
Kompatibilitas MySQL 5.7 158
Pembaruan Aurora MySQL: 2023-03-24 (versi 2.11.2, kompatibel dengan MySQL 5.7.12) - Versi Dukungan Diperpanjang RDS 159
Perbaikan 160
Fitur yang tidak didukung di Aurora MySQL versi 2 160
Kompatibilitas MySQL 5.7 160
Pembaruan Aurora MySQL: 2023-02-14 (versi 2.11.1, kompatibel dengan MySQL 5.7.12) - Versi Dukungan Diperpanjang RDS 161
Perbaikan 162
Perbandingan dengan Aurora MySQL versi 1 163
Kompatibilitas MySQL 5.7 164

Pembaruan Aurora MySQL: 2022-10-25 (versi 2.11.0, kompatibel dengan MySQL 5.7.12) - Versi Dukungan Diperpanjang RDS 164
Perbaikan 165
Integrasi perbaikan bug MySQL Community Edition 169
Perbandingan dengan Aurora MySQL versi 1 171
Kompatibilitas MySQL 5.7 171
Pembaruan Aurora MySQL: 2022-11-01 (versi 2.10.3) (Dihentikan) 172
Perbaikan 173
Integrasi perbaikan bug MySQL Community Edition 174
Perbandingan dengan Aurora MySQL versi 1 174
Kompatibilitas MySQL 5.7 175
Pembaruan Aurora MySQL: 2022-01-26 (versi 2.10.2) (Dihentikan) 175
Perbaikan 176
Integrasi perbaikan bug MySQL Community Edition 104
Perbandingan dengan Aurora MySQL versi 1 180
Kompatibilitas MySQL 5.7 180
Pembaruan Aurora MySQL: 2021-10-21 (versi 2.10.1) (Dihentikan) 181
Perbaikan 181
Integrasi perbaikan bug MySQL Community Edition 183
Perbandingan dengan Aurora MySQL versi 1 183
Kompatibilitas MySQL 5.7 183
Pembaruan Aurora MySQL: 2021-05-25 (versi 2.10.0) (Dihentikan) 184
Perbaikan 185
Integrasi perbaikan bug MySQL Community Edition 188
Perbandingan dengan Aurora MySQL versi 1 191
Kompatibilitas MySQL 5.7 192
Pembaruan Aurora MySQL: 2021-11-12 (versi 2.09.3) (Dihentikan) 192
Perbaikan 193
Integrasi perbaikan bug MySQL Community Edition 196
Perbandingan dengan Aurora MySQL versi 1 196
Kompatibilitas MySQL 5.7 197
Pembaruan Aurora MySQL: 2021-02-26 (versi 2.09.2) (Dihentikan) 197
Perbaikan 198
Perbandingan dengan Aurora MySQL versi 1 200
Kompatibilitas MySQL 5.7 200
Pembaruan Aurora MySQL: 2020-12-11 (versi 2.09.1) (Dihentikan) 201

Perbaikan 201
Integrasi perbaikan bug MySQL Community Edition 203
Perbandingan dengan Aurora MySQL versi 1 203
Kompatibilitas MySQL 5.7 204
Pembaruan Aurora MySQL: 2020-09-17 (versi 2.09.0) (Dihentikan) 204
Perbaikan 205
Integrasi perbaikan bug MySQL Community Edition 210
Perbandingan dengan Aurora MySQL versi 1 211
Kompatibilitas MySQL 5.7 212
Pembaruan Aurora MySQL: 2022-01-06 (versi 2.08.4) (Dihentikan) 213
Perbaikan 213
Perbandingan dengan Aurora MySQL versi 1 214
Kompatibilitas MySQL 5.7 214
Pembaruan Aurora MySQL: 2020-11-12 (versi 2.08.3) (Dihentikan) 215
Perbaikan 216
Integrasi perbaikan bug MySQL Community Edition 217
Perbandingan dengan Aurora MySQL versi 1 217
Kompatibilitas MySQL 5.7 217
Pembaruan Aurora MySQL: 2020-08-28 (versi 2.08.2) (Dihentikan) 218
Perbaikan 219
Perbandingan dengan Aurora MySQL versi 1 219
Kompatibilitas MySQL 5.7 220
Pembaruan Aurora MySQL: 2020-06-18 (versi 2.08.1) (Dihentikan) 220
Perbaikan 221
Perbandingan dengan Aurora MySQL versi 1 222
Kompatibilitas MySQL 5.7 222
Pembaruan Aurora MySQL: 2022-06-02 (versi 2.08.0) (Dihentikan) 223
Perbaikan 223
Integrasi perbaikan bug MySQL Community Edition 226
Perbandingan dengan Aurora MySQL versi 1 227
Kompatibilitas MySQL 5.7 227
Pembaruan Aurora MySQL: 2023-08-15 (versi 2.07.10) (Usang) 228
Perbaikan 229
Fitur yang tidak didukung di Aurora MySQL versi 2 230
Kompatibilitas MySQL 5.7 230
Pembaruan Aurora MySQL: 2023-05-04 (versi 2.07.9) (Usang) 231

Perbaikan 232
Fitur yang tidak didukung di Aurora MySQL versi 2 232
Kompatibilitas MySQL 5.7 232
Pembaruan Aurora MySQL: 2022-06-16 (versi 2.07.8) (Dihentikan) 233
Perbaikan 234
Integrasi perbaikan bug MySQL Community Edition 234
Perbandingan dengan Aurora MySQL versi 1 235
Kompatibilitas MySQL 5.7 235
Pembaruan Aurora MySQL: 2021-11-24 (versi 2.07.7) (Dihentikan) 236
Perbaikan 236
Perbandingan dengan Aurora MySQL versi 1 238
Kompatibilitas MySQL 5.7 238
Pembaruan Aurora MySQL: 2021-09-02 (versi 2.07.6) (Dihentikan) 239
Integrasi perbaikan bug MySQL Community Edition 239
Perbandingan dengan Aurora MySQL versi 1 239
Kompatibilitas MySQL 5.7 240
Pembaruan Aurora MySQL: 2021-07-06 (versi 2.07.5) (Dihentikan) 241
Perbaikan 241
Perbandingan dengan Aurora MySQL versi 1 242
Kompatibilitas MySQL 5.7 242
Pembaruan Aurora MySQL: 2021-03-04 (versi 2.07.4) (Dihentikan) 243
Perbaikan 244
Integrasi perbaikan bug MySQL Community Edition 245
Perbandingan dengan Aurora MySQL versi 1 245
Kompatibilitas MySQL 5.7 245
Pembaruan Aurora MySQL: 2020-11-10 (versi 2.07.3) (Dihentikan) 246
Perbaikan 247
Integrasi perbaikan bug MySQL Community Edition 248
Perbandingan dengan Aurora MySQL versi 1 249
Kompatibilitas MySQL 5.7 250
Pembaruan Aurora MySQL: 2020-04-17 (versi 2.07.2) (Dihentikan) 250
Perbaikan 251
Integrasi perbaikan bug MySQL Community Edition 252
Perbandingan dengan Aurora MySQL versi 1 252
Kompatibilitas MySQL 5.7 253
Pembaruan Aurora MySQL: 2019-12-23 (versi 2.07.1) (Dihentikan) 254

Perbaikan 255
Perbandingan dengan Aurora MySQL Versi 1 255
Kompatibilitas MySQL 5.7 255
Pembaruan Aurora MySQL: 2019-11-25 (versi 2.07.0) (Dihentikan) 256
Perbaikan 257
Integrasi perbaikan bug MySQL Community Edition 258
Perbandingan dengan Aurora MySQL versi 1 258
Kompatibilitas MySQL 5.7 259
Pembaruan Aurora MySQL: 2019-11-22 (versi 2.06.0) (Dihentikan) 259
Perbaikan 260
Perbandingan dengan Aurora MySQL versi 1 263
Kompatibilitas MySQL 5.7 263
Pembaruan Aurora MySQL: 2019-11-11 (versi 2.05.0) (Dihentikan) 264
Perbaikan 265
Integrasi perbaikan bug MySQL 266
Perbandingan dengan Aurora MySQL versi 1 266
Kompatibilitas MySQL 5.7 266
Pembaruan Aurora MySQL: 2020-08-14 (versi 2.04.9) (Dihentikan) 267
Perbaikan 268
Integrasi perbaikan bug MySQL 270
Perbandingan dengan Aurora MySQL versi 1 271
Kompatibilitas MySQL 5.7 272
Pembaruan Aurora MySQL: 2019-11-20 (versi 2.04.8) (Dihentikan) 272
Perbaikan 273
Perbandingan dengan Aurora MySQL versi 1 274
Kompatibilitas MySQL 5.7 274
Pembaruan Aurora MySQL: 2019-11-14 (versi 2.04.7) (Dihentikan) 275
Perbaikan 276
Perbandingan dengan Aurora MySQL versi 1 277
Kompatibilitas MySQL 5.7 277
Pembaruan Aurora MySQL: 2019-09-19 (versi 2.04.6) (Dihentikan) 278
Perbaikan 279
Integrasi perbaikan bug MySQL 279
Perbandingan dengan Aurora MySQL versi 1 279
Kompatibilitas MySQL 5.7 280
Pembaruan Aurora MySQL: 2019-07-08 (versi 2.04.5) (Dihentikan) 280

Perbaikan 281
Perbandingan dengan Aurora MySQL versi 1 282
Kompatibilitas MySQL 5.7 283
Pembaruan Aurora MySQL: 2019-05-29 (versi 2.04.4) (Dihentikan) 283
Perbaikan 284
Perbandingan dengan Aurora MySQL versi 1 285
Kompatibilitas MySQL 5.7 285
Pembaruan Aurora MySQL: 2019-05-09 (versi 2.04.3) (Dihentikan) 286
Perbaikan 287
Perbandingan dengan Aurora MySQL versi 1 287
Kompatibilitas MySQL 5.7 288
Pembaruan Aurora MySQL: 2019-05-02 (versi 2.04.2) (Dihentikan) 288
Perbaikan 289
Integrasi perbaikan bug MySQL 289
Perbandingan dengan Aurora MySQL versi 1 290
Kompatibilitas MySQL 5.7 290
Pembaruan Aurora MySQL: 2019-03-25 (versi 2.04.1) (Dihentikan) 291
Perbaikan 292
Perbandingan dengan Aurora MySQL versi 1 292
Kompatibilitas MySQL 5.7 292
Pembaruan Aurora MySQL: 2019-03-25 (versi 2.04.0) (Dihentikan) 293
Perbaikan 294
Integrasi perbaikan bug MySQL 294
Perbandingan dengan Aurora MySQL versi 1 294
Kompatibilitas MySQL 5.7 295
Pembaruan Aurora MySQL: 2019-02-07 (versi 2.03.4) (Dihentikan) 296
Perbaikan 296
Perbandingan dengan Aurora MySQL versi 1 297
Kompatibilitas MySQL 5.7 297
Pembaruan Aurora MySQL: 2019-01-18 (versi 2.01.3) (Dihentikan) 298
Perbaikan 299
Integrasi perbaikan bug MySQL 300
Perbandingan dengan Aurora MySQL versi 1 300
Kompatibilitas MySQL 5.7 301
Pembaruan Aurora MySQL: 2019-01-09 (versi 2.03.2) (Dihentikan) 301
Perbaikan 302

Perbandingan dengan Aurora MySQL versi 1 303
Kompatibilitas MySQL 5.7 303
Pembaruan Aurora MySQL: 2018-10-24 (versi 2.03.1) (Dihentikan) 304
Perbaikan 305
Perbandingan dengan Aurora MySQL versi 1 305
Kompatibilitas MySQL 5.7 305
Pembaruan Aurora MySQL: 2018-10-11 (versi 2.03) (Dihentikan) 306
Perbaikan 307
Integrasi perbaikan bug MySQL Community Edition 307
Perbandingan dengan Aurora MySQL versi 1 307
Kompatibilitas MySQL 5.7 308
Pembaruan Aurora MySQL: 2018-10-08 (versi 2.02.5) (Dihentikan) 309
Perbaikan 310
Perbandingan dengan Aurora MySQL versi 1 310
Kompatibilitas MySQL 5.7 310
Pembaruan Aurora MySQL: 2018-09-21 (versi 2.02.4) (Dihentikan) 311
Perbaikan 312
Integrasi perbaikan bug MySQL Community Edition 312
Perbandingan dengan Aurora MySQL versi 1 313
Kompatibilitas MySQL 5.7 313
Pembaruan Aurora MySQL: 2018-08-23 (versi 2.02.3) (Dihentikan) 314
Perbandingan dengan Aurora MySQL versi 1 315
Kompatibilitas MySQL 5.7 315
Perbedaan CLI antara Aurora MySQL 2.x dan Aurora MySQL 1.x 316
Perbaikan 316
Pembaruan Aurora MySQL: 2018-06-04 (versi 2.02.2) (Dihentikan) 317
Perbaikan 318
Perbandingan dengan Aurora MySQL versi 5.6 318
Kompatibilitas MySQL 5.7 318
Perbedaan CLI antara Aurora MySQL 2.x dan Aurora MySQL 1.x 316
Perbaikan 318
Pembaruan Aurora MySQL: 2018-05-03 (versi 2.02) (Dihentikan) 320
Perbandingan dengan Aurora MySQL versi 5.6 320
Kompatibilitas MySQL 5.7 321
Perbedaan CLI antara Aurora MySQL 2.x dan Aurora MySQL 1.x 316
Perbaikan 322

Integrasi perbaikan bug MySQL 323
Pembaruan Aurora MySQL: 2018-03-13 (versi 2.01.1) (Dihentikan) 323
Perbandingan dengan Aurora MySQL versi 5.6 323
Kompatibilitas MySQL 5.7 324
Perbedaan CLI antara Aurora MySQL 2.x dan Aurora MySQL 1.x 325
Perbaikan 325
Pembaruan Aurora MySQL: 2018-02-06 (versi 2.01) (Dihentikan) 325
Perbandingan dengan Aurora MySQL versi 5.6 326
Kompatibilitas MySQL 5.7 326
Perbedaan CLI antara Aurora MySQL 2.x dan Aurora MySQL 1.x 316
Aurora MySQL versi 1 (Dihentikan) 328
Pembaruan Aurora MySQL: 2021-09-30 (versi 1.23.4) (Dihentikan) 330
Perbaikan 330
Pembaruan Aurora MySQL: 2021-06-28 (versi 1.23.3) (Dihentikan) 331
Perbaikan 331
Pembaruan Aurora MySQL: 2021-03-18 (versi 1.23.2) (Dihentikan) 331
Perbaikan 332
Integrasi perbaikan bug MySQL Community Edition 333
Pembaruan Aurora MySQL: 2020-11-24 (versi 1.23.1) (Dihentikan) 334
Perbaikan 334
Pembaruan Aurora MySQL: 2020-09-02 (versi 1.23.0) (Dihentikan) 335
Perbaikan 336
Integrasi perbaikan bug MySQL Community Edition 339
Pembaruan Aurora MySQL: 2021-06-03 (versi 1.22.5) (Dihentikan) 341
Perbaikan 341
Pembaruan Aurora MySQL: 2021-03-04 (versi 1.22.4) (Dihentikan) 342
Perbaikan 342
Pembaruan Aurora MySQL: 2020-11-09 (versi 1.22.3) (Dihentikan) 343
Perbaikan 344
Integrasi perbaikan bug MySQL Community Edition 345
Pembaruan Aurora MySQL: 2020-03-05 (versi 1.22.2) (Dihentikan) 346
Perbaikan 346
Pembaruan Aurora MySQL: 2019-12-23 (versi 1.22.1) (Dihentikan) 347
Perbaikan 348
Pembaruan Aurora MySQL: 2019-11-25 (versi 1.22.0) (Dihentikan) 348
Perbaikan 349

Integrasi perbaikan bug MySQL Community Edition 352
Pembaruan Aurora MySQL: 2019-11-25 (versi 1.21.0) (Dihentikan) 353
Perbaikan 354
Integrasi perbaikan bug MySQL Community Edition 355
Pembaruan Aurora MySQL: 2020-03-05 (versi 1.20.1) (Dihentikan) 355
Perbaikan 356
Pembaruan Aurora MySQL: 2019-11-11 (versi 1.20.0) (Dihentikan) 356
Perbaikan 357
Integrasi perbaikan bug MySQL Community Edition 358
Pembaruan Aurora MySQL: 2020-03-05 (versi 1.19.6) (Dihentikan) 359
Perbaikan 359
Pembaruan Aurora MySQL: 2019-09-19 (versi 1.19.5) (Dihentikan) 360
Perbaikan 361
Integrasi perbaikan bug MySQL Community Edition 361
Pembaruan Aurora MySQL: 2019-06-05 (versi 1.19.2) (Dihentikan) 362
Perbaikan 362
Pembaruan Aurora MySQL: 2019-05-09 (versi 1.19.1) (Dihentikan) 363
Perbaikan 364
Pembaruan Aurora MySQL: 2019-02-07 (versi 1.19.0) (Dihentikan) 364
Fitur 365
Perbaikan 365
Integrasi perbaikan bug MySQL Community Edition 366
Pembaruan Aurora MySQL: 2018-09-20 (versi 1.18.0) (Dihentikan) 366
Fitur 367
Pembaruan Aurora MySQL: 2020-03-05 (versi 1.17.9) (Dihentikan) 368
Perbaikan 368
Pembaruan Aurora MySQL: 2019-01-17 (versi 1.17.8) (Dihentikan) 369
Perbaikan 369
Integrasi perbaikan bug MySQL Community Edition 369
Pembaruan Aurora MySQL: 2018-10-08 (versi 1.17.7) (Dihentikan) 370
Perbaikan 370
Integrasi perbaikan bug MySQL Community Edition 371
Pembaruan Aurora MySQL: 2018-09-06 (versi 1.17.6) (Dihentikan) 371
Perbaikan 372
Integrasi perbaikan bug MySQL Community Edition 372
Pembaruan Aurora MySQL: 2018-08-14 (versi 1.17.5) (Dihentikan) 372

Perbaikan 373
Pembaruan Aurora MySQL: 2018-08-07 (versi 1.17.4) (Dihentikan) 373
Perbaikan 374
Pembaruan Aurora MySQL: 2018-06-05 (versi 1.17.3) (Dihentikan) 374
Perbaikan 375
Pembaruan Aurora MySQL: 2018-04-27 (versi 1.17.2) (Dihentikan) 376
Perbaikan 376
Pembaruan Aurora MySQL: 2018-03-23 (versi 1.17.1) (Dihentikan) 376
Perbaikan 377
Pembaruan Aurora MySQL: 2018-03-13 (versi 1.17) (Dihentikan) 377
Zero-downtime patching 378
Fitur baru 378
Perbaikan 378
Integrasi perbaikan bug MySQL 379
Pembaruan Aurora MySQL: 2017-12-11 (versi 1.16) (Dihentikan) 379
Zero-downtime patching 380
Fitur baru 380
Perbaikan 380
Integrasi perbaikan bug MySQL 381
Pembaruan Aurora MySQL: 2017-11-20 (versi 1.15.1) (Dihentikan) 381
Zero-downtime patching 381
Perbaikan 381
Integrasi perbaikan bug MySQL 382
Pembaruan Aurora MySQL: 2017-10-24 (versi 1.15) (Dihentikan) 382
Zero-downtime patching 383
Fitur baru 383
Perbaikan 383
Integrasi perbaikan bug MySQL 382
Pembaruan Aurora MySQL: 2018-03-13 (versi 1.14.4) (Dihentikan) 385
Zero-downtime patching 385
Fitur baru 386
Perbaikan 386
Integrasi perbaikan bug MySQL 386
Pembaruan Aurora MySQL: 2017-09-22 (versi 1.14.1) (Dihentikan) 386
Perbaikan 387
Pembaruan Aurora MySQL: 2017-08-07 (versi 1.14) (Dihentikan) 387

Zero-downtime patching 387
Perbaikan 387
Integrasi perbaikan bug MySQL 389
Pembaruan Aurora MySQL: 2017-05-15 (versi 1.13) (Dihentikan) 389
Zero-downtime patching 389
Fitur baru 389
Perbaikan 390
Integrasi perbaikan bug MySQL 391
Pembaruan Aurora MySQL: 2017-04-05 (versi 1.12) (Dihentikan) 391
Fitur baru 392
Perbaikan 392
Integrasi perbaikan bug MySQL 393
Pembaruan Aurora MySQL: 2017-02-23 (versi 1.11) (Dihentikan) 393
Fitur baru 394
Perbaikan 394
Integrasi perbaikan bug MySQL 396
Pembaruan Aurora MySQL: 2017-01-12 (versi 1.10.1) (Dihentikan) 397
Fitur baru 397
Perbaikan 397
Pembaruan Aurora MySQL: 2016-12-14 (versi 1.10) (Dihentikan) 397
Fitur baru 397
Perbaikan 399
Integrasi perbaikan bug MySQL 400
Pembaruan Aurora MySQL 2016-11-10 (versi 1.9.0, 1.9.1) (Dihentikan) 400
Fitur baru 400
Perbaikan 401
Pembaruan Aurora MySQL: 2016-10-26 (versi 1.8.1) (Dihentikan) 401
Perbaikan 401
Integrasi perbaikan bug MySQL 402
Pembaruan Aurora MySQL: 2016-10-18 (versi 1.8) (Dihentikan) 402
Fitur baru 402
Perbaikan 402
Integrasi perbaikan bug MySQL 403
Pembaruan Aurora MySQL: 2016-09-20 (versi 1.7.1) (Dihentikan) 404
Perbaikan 404
Pembaruan Aurora MySQL: 2016-08-30 (versi 1.7.0) (Dihentikan) 404

Fitur baru 404
Perbaikan 405
Integrasi perbaikan bug MySQL 405
Pembaruan Aurora MySQL: 2016-06-01 (versi 1.6.5) (Dihentikan) 406
Fitur baru 406
Perbaikan 406
Integrasi perbaikan bug MySQL 407
Pembaruan Aurora MySQL: 2016-04-06 (versi 1.6) (Dihentikan) 407
Fitur baru 407
Perbaikan 408
Integrasi perbaikan bug MySQL 409
Pembaruan Aurora MySQL: 2016-01-11 (versi 1.5) (Dihentikan) 409
Perbaikan 410
Integrasi perbaikan bug MySQL 410
Pembaruan Aurora MySQL: 2015-12-03 (versi 1.4) (Dihentikan) 410
Fitur baru 411
Perbaikan 411
Integrasi perbaikan bug MySQL 412
Pembaruan Aurora MySQL: 2015-10-16 (versi 1.2, 1.3) (Dihentikan) 412
Perbaikan 412
Perbaikan 413
Integrasi perbaikan bug MySQL 413
Pembaruan Aurora MySQL: 2015-08-24 (versi 1.1) (Dihentikan) 416
Bug MySQL yang diperbaiki oleh pembaruan Aurora MySQL 418
Bug MySQL yang diperbaiki oleh pembaruan Aurora MySQL 3.x 418
Bug MySQL yang diperbaiki oleh pembaruan Aurora MySQL 2.x 433
Bug MySQL yang diperbaiki oleh pembaruan Aurora MySQL 1.x 456
Kerentanan keamanan yang diperbaiki di Aurora MySQL 476
Riwayat dokumen 484
..... div

Catatan rilis untuk Amazon Aurora Edisi Kompatibel MySQL

Amazon Aurora Edisi Kompatibel MySQL merilis pembaruan secara rutin. Pembaruan diterapkan pada klaster DB Aurora MySQL selama jadwal pemeliharaan sistem. Waktu saat pembaruan diterapkan tergantung pada pengaturan jendela Wilayah AWS dan pemeliharaan untuk cluster DB, dan juga jenis pembaruan.

Rilis Amazon Aurora MySQL tersedia untuk semua Wilayah AWS selama beberapa hari. Untuk sementara waktu, beberapa Wilayah mungkin menampilkan versi mesin yang belum tersedia di Wilayah lain. Catatan rilis diterbitkan setelah versi baru tersedia di semua Wilayah.

Pembaruan diterapkan ke semua instans dalam klaster DB pada saat yang sama. Pembaruan memerlukan mulai ulang basis data di semua instans dalam klaster DB. Jadi Anda mengalami waktu henti selama 20 hingga 30 detik, setelah itu Anda dapat melanjutkan menggunakan klaster DB Anda. Anda dapat melihat atau mengubah pengaturan jadwal pemeliharaan dari Konsol Manajemen AWS.

Topik

- Kalender rilis untuk Amazon Aurora MySQL
- Pembaruan mesin basis data untuk Amazon Aurora MySQL versi 2
- Pembaruan mesin basis data untuk Amazon Aurora MySQL versi 2
- Pembaruan mesin basis data untuk Amazon Aurora MySQL versi 1 (Dihentikan)
- Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL
- Kerentanan keamanan yang diperbaiki di Aurora MySQL

Kalender rilis untuk Amazon Aurora MySQL

Kalender rilis di halaman ini dapat membantu Anda merencanakan peningkatan versi mayor dan minor Anda. Untuk informasi selengkapnya tentang peningkatan, pembuatan versi, dan siklus hidup Amazon Aurora, lihat versi Amazon Aurora.

Topik

- Kalender rilis untuk Aurora MySQL versi utama
- Kalender rilis untuk Aurora MySQL versi minor

Kalender rilis untuk Aurora MySQL versi utama

Versi utama Aurora MySQL tersedia di bawah dukungan standar setidaknya sampai akhir komunitas untuk versi komunitas yang sesuai. Anda dapat terus menjalankan versi mayor melewati tanggal akhir dukungan standar Aurora dengan biaya tertentu. Untuk informasi selengkapnya, lihat Menggunakan Amazon RDS Extended Support dan harga Amazon Aurora.

Aurora MySQL saat ini mendukung versi utama berikut.

> **Note**
>
> Anda juga dapat melihat informasi tentang tanggal dukungan untuk versi mesin utama dengan menggunakan AWS CLI atau RDS API. Untuk informasi selengkapnya, lihat Melihat tanggal dukungan untuk versi engine di Amazon RDS Extended Support.

Versi mayor komunitas	Versi mayor Aurora	Tanggal akhir masa penggunaan komunitas	Tanggal akhir dukungan standar Aurora	Tanggal mulai harga tahun ke-1 Dukungan yang Diperluas RDS	Tanggal mulai harga tahun ke-3 Dukungan yang Diperluas RDS	Tanggal akhir Dukungan yang Diperluas RDS	Versi minor yang memenuhi syarat untuk Dukungan yang Diperluas RDS
MySQL 8.0	Aurora MySQL versi 3	April 2026	30 April 2028	1 Mei 2028	Tidak berlaku	31 Juli 2029	Akan ditentukan
MySQL 5.7	Aurora MySQL versi 2	Oktober 2023	31 Oktober 2024	1 Desember 2024	Tidak berlaku	28 Februari 2027	Aurora MySQL 2.11 dan 2.12
MySQL 5.6 (sudah dihentikan)	Aurora MySQL versi 1 (usang)	5 Februari 2021	28 Februari 2023	Tidak berlaku	Tidak berlaku	Tidak berlaku	Tidak berlaku

Note

Amazon RDS Extended Support untuk Aurora MySQL versi 2 dimulai pada 1 November 2024, tetapi Anda tidak akan dikenakan biaya hingga 1 Desember 2024. Antara 1 November dan 30 November 2024, semua cluster DB Aurora MySQL versi 2 tercakup dalam Amazon RDS Extended Support.

Kalender rilis untuk Aurora MySQL versi minor

Aurora MySQL saat ini mendukung versi minor berikut.

Secara umum, versi minor Aurora dirilis setiap tiga bulan. Jadwal rilis mungkin bervariasi untuk menyertakan fitur atau perbaikan tambahan.

Versi minor dapat mencapai akhir dukungan standar sebelum versi utama yang sesuai melakukannya. Misalnya, versi 3.07 akan mencapai akhir tanggal dukungan standar pada Agustus 2025, sementara versi utama 3 akan mencapai akhir dukungan standar pada 30 April 2028. Amazon RDS akan mendukung versi minor 3.* tambahan yang dirilis antara tanggal-tanggal ini.

Biaya Amazon RDS Extended Support hanya berlaku untuk versi minor tertentu setelah versi utama memenuhi syarat untuk Extended Support. Untuk informasi selengkapnya tentang versi utama yang memenuhi syarat untuk Extended Support, lihatKalender rilis untuk Aurora MySQL versi utama.

Versi Aurora MySQL	Tanggal rilis Aurora MySQL	Aurora MySQL akhir dari tanggal dukungan standar
3.12 (Kompatibel dengan MySQL Komunitas 8.0.44)	Februari 17, 2026	Februari 17, 2027
3.11 (Kompatibel dengan MySQL Komunitas 8.0.43)	November 13, 2025	November 13, 2026
3.10 (Kompatibel dengan Komunitas MySQL 8.0.42) (LTS)	Juli 31, 2025	April 30, 2028
3.09 (Kompatibel dengan MySQL Komunitas 8.0.40)	14 Mei 2025	31 Mei 2026
3.08 (Kompatibel dengan MySQL Komunitas 8.0.39)	November 18, 2024	31 Mei 2026
3.04 (Kompatibel dengan Komunitas MySQL 8.0.28) (LTS)	31 Juli 2023	Oktober 31, 2026

Versi Aurora MySQL	Tanggal rilis Aurora MySQL	Aurora MySQL akhir dari tanggal dukungan standar
2.12 [1] (Kompatibel dengan Komunitas MySQL 5.7.40 atau 5.7.44 [2])	25 Juli 2023	Oktober 31, 2024
2.11 [1] (Kompatibel dengan MySQL Komunitas 5.7.12)	25 Oktober 2022	Oktober 31, 2024

LTS - Versi dukungan jangka panjang Aurora MySQL (LTS). Untuk informasi selengkapnya, lihat Dukungan jangka panjang (LTS) dan rilis beta untuk Amazon Aurora MySQL.

[1] Versi minor ini akan terus tersedia ketika versi utama ada di Amazon RDS Extended Support.

[2] Aurora MySQL 2.12 versi melalui 2.12.1 kompatibel dengan MySQL versi 5.7.40, dan versi 2.12.2 dan yang lebih tinggi kompatibel dengan MySQL versi 5.7.44.

Pembaruan mesin basis data untuk Amazon Aurora MySQL versi 2

Berikut ini adalah pembaruan mesin basis data untuk Amazon Aurora MySQL versi 3.

Topik

- Pembaruan mesin database Aurora MySQL 2026-02-17 (versi 3.12.0, kompatibel dengan MySQL 8.0.44)
- Pembaruan mesin database Aurora MySQL 2025-12-16 (versi 3.11.1, kompatibel dengan MySQL 8.0.43)
- Pembaruan mesin database Aurora MySQL 2025-11-13 (versi 3.11.0, kompatibel dengan MySQL 8.0.43)
- Pembaruan mesin database Aurora MySQL 2026-01-02 (versi 3.10.3, kompatibel dengan MySQL 8.0.42)
- Pembaruan mesin database Aurora MySQL 2025-11-20 (versi 3.10.2, kompatibel dengan MySQL 8.0.42)
- Pembaruan mesin database Aurora MySQL 2025-09-30 (versi 3.10.1, kompatibel dengan MySQL 8.0.42)
- Pembaruan mesin database Aurora MySQL 2025-07-31 (versi 3.10.0, kompatibel dengan MySQL 8.0.42)
- Pembaruan mesin database Aurora MySQL 2025-05-14 (versi 3.09.0, kompatibel dengan MySQL 8.0.40)
- Pembaruan mesin database Aurora MySQL 2025-04-07 (versi 3.08.2, kompatibel dengan MySQL 8.0.39)
- Pembaruan mesin database Aurora MySQL 2025-02-17 (versi 3.08.1, kompatibel dengan MySQL 8.0.39)
- Pembaruan mesin database Aurora MySQL 2024-11-18 (versi 3.08.0, kompatibel dengan MySQL 8.0.39)
- Pembaruan mesin database Aurora MySQL 2024-07-23 (versi 3.07.1) (Usang)
- Pembaruan mesin database Aurora MySQL 2024-06-04 (versi 3.07.0) (Usang)
- Pembaruan mesin database Aurora MySQL 2024-06-26 (versi 3.06.1) (Usang)

- Pembaruan mesin database Aurora MySQL 2024-03-07 (versi 3.06.0) (Usang)
- Pembaruan mesin database Aurora MySQL 2024-01-31 (versi 3.05.2) (Usang)
- Pembaruan mesin database Aurora MySQL 2023-11-21 (versi 3.05.1) (Usang)
- Pembaruan mesin database Aurora MySQL 2023-10-30 (versi 3.05.0.1) (Usang)
- Pembaruan mesin database Aurora MySQL 2023-10-25 (versi 3.05.0) (Usang)
- Pembaruan mesin database Aurora MySQL 2026-01-02 (versi 3.04.6, kompatibel dengan MySQL 8.0.28)
- Pembaruan mesin database Aurora MySQL 2025-05-05 (versi 3.04.4, kompatibel dengan MySQL 8.0.28)
- Pembaruan mesin database Aurora MySQL 2024-06-26 (versi 3.04.3, kompatibel dengan MySQL 8.0.28)
- Pembaruan mesin database Aurora MySQL 2024-03-15 (versi 3.04.2, kompatibel dengan MySQL 8.0.28)
- Pembaruan mesin database Aurora MySQL 2023-11-13 (versi 3.04.1, kompatibel dengan MySQL 8.0.28)
- Pembaruan mesin database Aurora MySQL 2023-07-31 (versi 3.04.0), kompatibel dengan MySQL 8.0.28)
- Pembaruan mesin database Aurora MySQL 2023-12-08 (versi 3.03.3) (Usang)
- Pembaruan mesin database Aurora MySQL 2023-08-29 (versi 3.03.2) (Usang)
- Pembaruan mesin database Aurora MySQL 2023-05-11 (versi 3.03.1) (Usang)
- Pembaruan mesin database Aurora MySQL 2023-03-01 (versi 3.03.0) (Usang)
- Pembaruan mesin database Aurora MySQL 2023-04-17 (versi 3.02.3) (Usang)
- Pembaruan mesin database Aurora MySQL 2022-11-18 (versi 3.02.2) (Usang)
- Pembaruan mesin database Aurora MySQL 2022-09-07 (versi 3.02.1) (Usang)
- Pembaruan mesin database Aurora MySQL 2022-04-20 (versi 3.02.0) (Usang)
- Pembaruan mesin database Aurora MySQL 2022-04-15 (versi 3.01.1) (Usang)
- Pembaruan mesin database Aurora MySQL 2021-11-18 (versi 3.01.0) (Usang)

Pembaruan mesin database Aurora MySQL 2026-02-17 (versi 3.12.0, kompatibel dengan MySQL 8.0.44)

Versi: 3.12.0

Aurora MySQL 3.12.0 umumnya tersedia. Versi Aurora MySQL 3.12 kompatibel dengan MySQL 8.0.44. Untuk informasi selengkapnya tentang perubahan komunitas yang telah terjadi, lihat Catatan Rilis MySQL 8.0.

Untuk detail fitur-fitur baru di Aurora MySQL versi 3, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0. Untuk perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2, lihat Perbandingan Aurora MySQL versi 2 dan Aurora MySQL versi 3. Untuk perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition, lihat Perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition di Panduan Pengguna Amazon Aurora.

Anda dapat melakukan pemutakhiran di tempat dengan memanfaatkan Zero Downtime Patching (ZDP), memulihkan snapshot, atau memulai pemutakhiran terkelola blue/green menggunakan Penyebaran Amazon RDS dari Blue/Green klaster Aurora MySQL versi 2 yang saat ini didukung ke dalam klaster Aurora MySQL versi 3.12.0.

Untuk informasi tentang merencanakan peningkatan ke Aurora MySQL versi 3, lihat Merencanakan peningkatan versi utama untuk klaster MySQL Aurora. Untuk informasi umum tentang peningkatan Aurora MySQL, lihat Meningkatkan klaster DB Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Untuk informasi pemecahan masalah, lihat Pemecahan Masalah untuk peningkatan Aurora MySQL di tempat di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan keamanan

- Memperbaiki masalah yang dapat menyebabkan beberapa pernyataan SQL tidak masuk log dalam log audit.
- Memperbaiki plugin caching_sha2_password untuk memastikan pembaca Aurora MySQL dan instance penulis memvalidasi kata sandi yang diperbarui secara konsisten.

Sedang CVEs:

- CVE-2025-53040
- CVE-2025-53042
- CVE-2025-53044
- CVE-2025-53045
- CVE-2025-53053
- CVE-2025-53054
- CVE-2025-53062
- CVE-2025-53069

Perbaikan ketersediaan:

- Memperbaiki masalah yang dapat menyebabkan instance database restart ketika tabel kunci penuh selama pembacaan semi-konsisten.
- Memperbaiki masalah dalam pemulihan binlog yang dapat menyebabkan instance replika binlog dimulai ulang saat menggunakan `aurora_in_memory_relaylog` dengan replikasi multi-utas.
- Memperbaiki masalah yang dapat menyebabkan instance penulis dimulai ulang saat penerusan penulisan global atau penerusan penulisan lokal dinonaktifkan.
- Memperbaiki masalah di mana konfigurasi thread_stack yang terlalu besar dapat mencegah bootup server MySQL Aurora selama restart atau upgrade. Server Aurora MySQL sekarang secara otomatis me-reset thread_stack ke nilai default engine (1 MB) ketika melebihi memori sistem, mencegah kegagalan startup.
- Memperbaiki masalah yang, dapat menyebabkan mesin restart saat berjalan `KILL` setelah menjalankan kueri paralel yang `EXPLAIN FOR CONNECTION` sedang berjalan.
- Memperbaiki masalah yang dapat memaksa kueri paralel pada kueri acak dan menyebabkan mesin restart jika kueri ini dihentikan secara paksa.
- Memperbaiki masalah yang mencegah pengguna dengan hak istimewa CONNECTION_ADMIN atau SUPER melakukan koneksi tambahan di luar batas max_connections, seperti yang didukung di MySQL Community Edition.
- Memperbaiki masalah yang dapat menyebabkan restart pada replika log biner (binlog) saat memproses sejumlah besar file log relai selama pemulihan log relai.
- Memperbaiki masalah yang menyebabkan instans DB penulis dimulai ulang saat menjalankan kueri paralel pada instans DB pembaca.

- Memperbaiki masalah yang, dalam kondisi langka, dapat menonaktifkan logging biner ketika terjadi kesalahan selama komit transaksi besar.
- Memperbaiki masalah yang dapat menyebabkan replika baca dimulai ulang jika terjadi pesanan komit transaksi langka tertentu pada instans DB penulis.
- Memperbaiki masalah yang dapat menyebabkan database dimulai ulang saat peristiwa terjadwal dibatalkan selama eksekusi pada instance yang telah mengaktifkan binlog yang disempurnakan.
- Memperbaiki masalah saat instance database menggunakan ukuran penyimpanan multi TiB, mungkin mengalami peningkatan waktu henti selama restart karena kegagalan validasi bufferpool InnoDB.

Perbaikan umum:

- Mengecualikan perubahan komunitas yang diperkenalkan di MySQL 8.0.44 di mana upaya koneksi yang gagal secara tak terduga menambahkan baris ke tabel performance_schema yang menyebabkan konsumsi memori yang berlebihan. Ref komunitas Bug #119766
- Memperbaiki masalah yang memengaruhi prosedur tersimpan replikasi khusus Aurora saat mengonfigurasi replikasi binlog pada instance dengan pengaturan pemeriksaan khusus.
- Memperbaiki masalah untuk mengurangi overhead CPU saat membuat Enkripsi di Transit antara mesin database dan lapisan penyimpanan.
- Peningkatan kinerja menulis IOPS ketika variabel sistem innodb_flush_log_at_trx_commit diatur ke 0.
- Secara otomatis menonaktifkan tindakan aurora_oom_response (kecuali cetak, jika dikonfigurasi) ketika aurora_oom_response gagal menyelesaikan tekanan memori setelah batas waktu (dalam urutan beberapa menit).
- Menyelesaikan kondisi balapan yang dapat menyebabkan pembacaan halaman salah dari kumpulan buffer selama operasi penskalaan tanpa server Aurora atau selama penggusuran halaman dari kumpulan buffer. Ref komunitas Bug #116305.
- Memperbaiki masalah out-of-memory (OOM) yang dapat menyebabkan pembaca memulai ulang saat menjalankan perintah terkait hak istimewa pada penulis.
- Hak istimewa berikut telah ditambahkan `kerds_superuser_role`:`FLUSH_OPTIMIZER_COSTS, FLUSH_STATUS, FLUSH_TABLES, FLUSH_USER_RESOURCES`. Untuk informasi tentang ini`rds_superuser_role`, silakan lihat Akun Pengguna Amazon Master dengan dokumentasi Aurora. Untuk informasi lebih lanjut tentang hak istimewa dinamis ini, silakan lihat dokumentasi MySQL.

- Memperbaiki masalah yang menyebabkan pelacakan permintaan query paralel yang tidak akurat saat menjalankan `EXPLAIN ANALYZE` pernyataan di mana `Aurora_pq_request_in_progress` penghitung tidak diperbarui secara akurat.
- Memperbaiki masalah saat koneksi yang diawetkan ditangani secara tidak benar selama zero-downtime patching (ZDP) /zero-downtime restart (ZDR) yang dapat menyebabkan klien menunggu tanpa batas waktu untuk menyelesaikan kueri.
- Perbaiki masalah saat baris menjadi tidak dapat dibaca melalui indeks spasial selama pembaruan.
- Memperbaiki masalah saat kueri yang berisi petunjuk pengoptimal yang dibatalkan selama operasi zero-downtime restart (ZDR) atau zero-downtime patching (ZDP) mungkin salah ditangani.
- Perbaiki masalah di mana latensi komit tidak diukur saat innodb_flush_log_at_trx_commit disetel ke 0.
- Memperkenalkan optimasi untuk mengurangi penggunaan memori selama pemulihan kamus data logis ketika ada sejumlah besar tabel.
- Memperbaiki masalah yang menyebabkan `SHOW BINARY LOGS` perintah memakan waktu lebih lama untuk dijalankan di cluster tempat Enhanced Binlog diaktifkan atau sebelumnya diaktifkan. Masalah ini juga dapat menyebabkan peningkatan latensi komit jika beberapa `SHOW BINARY LOGS` perintah berjalan secara bersamaan.
- Perbaiki masalah memori yang terkait dengan peran default dari view definer.
- Memperbaiki masalah yang dapat menyebabkan kegagalan dalam menyelesaikan proses menonaktifkan fitur “tulis penerusan”.
- Memperbaiki masalah yang dapat menyebabkan instance database penulis dimulai ulang saat instance pembaca menggunakan penerusan tulis mengeksekusi pernyataan DMLyang berisi nilai stempel waktu dan parameter database time_zone disetel ke “UTC”.
- Memperbaiki masalah, yang dalam kasus yang jarang terjadi, menyebabkan tidak tersedianya Replika Baca Aurora atau ketidakkonsistenan definisi tabel dengan kesalahan 'Tabel tidak ada' pada replika karena kueri baca bersamaan pada operasi replika dan DDL pada penulis.
- Memperbaiki masalah yang dapat menyebabkan set hasil yang tidak lengkap saat menjalankan kueri yang melibatkan operasi LEFT- atau RIGHT-JOIN menggunakan algoritma hash-join dengan query paralel.

Integrasi perbaikan bug MySQL Community Edition

Rilis ini mencakup semua perbaikan bug komunitas hingga dan termasuk 8.0.44. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 3.x.

- Memperbaiki masalah di mana kueri formulir `SELECT 1 FROM t WHERE CAST(a AS UNSIGNED INTEGER) = 1 AND a = (SELECT 1 FROM t)` menyebabkan kegagalan pernyataan di. `item_func.cc` (Perbaikan Bug Komunitas #36128964)
- Memperbaiki masalah yang menyelesaikan kebuntuan saat FLUSH STATUS, COM_CHANGE_USER dan SHOW PROCESS LIST dijalankan secara bersamaan. (Bug #35218030)

Pembaruan mesin database Aurora MySQL 2025-12-16 (versi 3.11.1, kompatibel dengan MySQL 8.0.43)

Versi: 3.11.1

Aurora MySQL 3.11.1 umumnya tersedia. Versi Aurora MySQL 3.11 kompatibel dengan MySQL 8.0.43. Untuk informasi selengkapnya tentang perubahan komunitas yang telah terjadi, lihat Catatan Rilis MySQL 8.0.

Untuk detail fitur-fitur baru di Aurora MySQL versi 3, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0. Untuk perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2, lihat Perbandingan Aurora MySQL versi 2 dan Aurora MySQL versi 3. Untuk perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition, lihat Perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition di Panduan Pengguna Amazon Aurora.

Anda dapat melakukan pemutakhiran di tempat dengan memanfaatkan Zero Downtime Patching (ZDP), memulihkan snapshot, atau memulai pemutakhiran terkelola blue/green menggunakan Penyebaran Amazon RDS dari Blue/Green klaster Aurora MySQL versi 2 yang saat ini didukung ke dalam klaster Aurora MySQL versi 3.11.1.

Untuk informasi tentang merencanakan peningkatan ke Aurora MySQL versi 3, lihat Merencanakan peningkatan versi utama untuk klaster MySQL Aurora. Untuk informasi umum tentang peningkatan Aurora MySQL, lihat Memutakhirkan klaster DB MySQL Aurora di Panduan Pengguna Amazon Aurora.

Untuk informasi pemecahan masalah, lihat Pemecahan Masalah untuk peningkatan Aurora MySQL di tempat di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah, Dukungan tersedia di forum komunitas dan melalui Dukungan. Untuk informasi selengkapnya, lihat Memelihara klaster DB Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan umum

- Memperbaiki masalah yang dapat menyebabkan cluster database membuat atau memulihkan membutuhkan waktu lama untuk diselesaikan.
- Mengurangi overhead penggunaan CPU saat membuat Enkripsi di Transit antara instance database dan lapisan penyimpanan.
- Memperbaiki masalah yang dapat menyebabkan beberapa pernyataan SQL tidak masuk log dalam log audit.

Pembaruan mesin database Aurora MySQL 2025-11-13 (versi 3.11.0, kompatibel dengan MySQL 8.0.43)

Versi: 3.11.0

Aurora MySQL 3.11.0 umumnya tersedia. Versi Aurora MySQL 3.11 kompatibel dengan MySQL 8.0.43. Untuk informasi selengkapnya tentang perubahan komunitas yang telah terjadi, lihat Catatan Rilis MySQL 8.0.

Untuk detail fitur-fitur baru di Aurora MySQL versi 3, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0. Untuk perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2, lihat Perbandingan Aurora MySQL versi 2 dan Aurora MySQL versi 3. Untuk perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition, lihat Perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition di Panduan Pengguna Amazon Aurora.

Anda dapat melakukan upgrade di tempat dengan memanfaatkan zero-downtime patching (ZDP), memulihkan snapshot, atau memulai upgrade terkelola blue/green menggunakan Amazon RDS Deployment dari klaster Aurora MySQL versi 2 yang saat ini didukung ke dalam klaster Aurora Blue/Green MySQL versi 3.11.0.

Untuk informasi tentang merencanakan peningkatan ke Aurora MySQL versi 3, lihat Merencanakan peningkatan versi utama untuk klaster MySQL Aurora. Untuk informasi umum tentang peningkatan Aurora MySQL, lihat Memutakhirkan klaster DB MySQL Aurora di Panduan Pengguna Amazon Aurora.

Untuk informasi pemecahan masalah, lihat Pemecahan Masalah untuk peningkatan Aurora MySQL di tempat di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah,Dukungan tersedia di forum komunitas dan melalui Dukungan. Untuk informasi selengkapnya, lihat Memelihara klaster DB Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Memperbaiki masalah keamanan dan CVEs

Sedang CVEs:

- CVE-2025-50077
- CVE-2025-50078
- CVE-2025-50079
- CVE-2025-50080
- CVE-2025-50082
- CVE-2025-50083
- CVE-2025-50084
- CVE-2025-50085
- CVE-2025-50086
- CVE-2025-50087
- CVE-2025-50088
- CVE-2025-50091
- CVE-2025-50092
- CVE-2025-50093
- CVE-2025-50094
- CVE-2025-50096

- CVE-2025-50097
- CVE-2025-50099
- CVE-2025-50101
- CVE-2025-50102
- CVE-2025-53023

Rendah CVEs:

- CVE-2025-50098
- CVE-2025-50100
- CVE-2025-50104

Peningkatan ketersediaan

- Memperbaiki masalah dalam pemulihan logis dengan `aurora_enhanced_binlog` diaktifkan yang dapat mencegah restart basis data jika terjadi transaksi yang dibatalkan.
- Memperbaiki masalah yang dapat menyebabkan instance penulis dimulai ulang saat penerusan penulisan global atau penerusan penulisan lokal dinonaktifkan.
- Memperbaiki masalah yang dapat menyebabkan penulis menjadi tidak responsif saat penerusan penulisan dinonaktifkan sementara dua atau lebih replika meneruskan permintaan.
- Memperbaiki masalah yang dapat menyebabkan mesin restart saat berjalan `KILL <query-id>` setelah menjalankan kueri paralel yang `EXPLAIN FOR CONNECTION <query-id>` sedang berjalan.
- Memperbaiki masalah di mana penghindaran out-of-memory (OOM) tidak mempertahankan nilai parameter `aurora_oom_response` DB yang dikonfigurasi setelah database dimulai ulang.
- Memperbaiki masalah yang mencegah pengguna dengan `CONNECTION_ADMIN` atau `SUPER` hak istimewa melakukan koneksi tambahan di luar `max_connections` batas, seperti yang didukung dalam MySQL Community Edition.
- Memperbaiki masalah dengan penskalaan Aurora Serverless V2 yang mengakibatkan instans DB dimulai ulang dengan mencegah halaman memori kritis ditukar.
- Memperbaiki masalah yang dapat menyebabkan instance Aurora MySQL Serverless v2 dimulai ulang saat `innodb_purge_threads` parameter dikonfigurasi secara manual ke nilai yang berbeda dari default. `innodb_purge_threads`Parameter sekarang dikelola secara otomatis untuk instance Aurora Serverless v2 dan tidak dapat dimodifikasi.

- Memperbaiki masalah yang dapat menyebabkan restart instance pembaca gagal saat penulis menjalankan sejumlah besar operasi DDL.
- Memperbaiki masalah yang dapat menyebabkan instance penulis menjadi tidak responsif jika instance pembaca dimulai ulang saat menggunakan Global Write Forwarding atau Local Write Forwarding.
- Memperbaiki masalah yang dapat menyebabkan engine restart selama zero-downtime patching (ZDP) /zero-downtime restart (ZDR), saat mempertahankan koneksi dengan transaksi aktif. SSL/TLS
- Memperbaiki masalah yang, dalam kondisi langka, dapat menyebabkan perubahan pada instance penulis dikirim ke instance pembaca dalam urutan yang tidak kompatibel dengan utas pemrosesan kueri. Hal ini dapat menyebabkan kebuntuan antara thread pemrosesan kueri dan utas penerapan replikasi, yang pada gilirannya dapat menyebabkan instance pembaca dimulai ulang.
- Memperbaiki masalah untuk mencegah restart server database yang tidak perlu yang terjadi karena penilaian kesehatan agen pemantauan yang salah.

Perbaikan umum

- Memperbaiki masalah di mana penulisan ke database mungkin terhenti saat menjalankan transaksi yang berjalan lama yang mengarah ke restart basis data atau upgrade versi utama gagal.
- Memperbaiki masalah dengan replika yang salah di-restart saat bergabung dengan penulis.
- Kinerja database yang dioptimalkan untuk instans database dengan volume Penyimpanan Aurora yang besar.
- Memperbaiki masalah dalam pengoptimal rentang di mana ekspresi konstan tidak dikenali saat menghitung batas rentang. (Bug Komunitas #112737)
- Memperbaiki masalah saat mengekspor snapshot DB ke Amazon S3 berkinerja lebih lambat saat mengekspor tabel yang berisi kolom JSON dengan nilai nol.
- Memperbaiki masalah di mana file log biner sementara tidak dibersihkan dengan benar setelah rollback transaksi saat menggunakan logging biner. Perbaikan ini mencegah konsumsi penyimpanan dari file sementara yang disimpan dan dalam kasus tertentu juga dapat mencegah anomali dalam file log biner.
- Peningkatan kinerja menulis IOPS ketika variabel sistem `innodb_flush_log_at_trx_commit` diatur ke`0`.

- Memperbaiki masalah saat zero downtime patching (ZDP) /zero downtime restart (ZDR) dapat mengakibatkan instans DB restart saat memulihkan peringatan dengan kode kesalahan yang tidak valid.
- Secara otomatis menonaktifkan `aurora_oom_response` tindakan (kecuali cetak, jika dikonfigurasi) ketika `aurora_oom_response` gagal menyelesaikan tekanan memori setelah waktu ambang batas (dalam urutan beberapa menit).
- Memperbaiki masalah di mana kolom dengan pembaruan JSON sebagian gagal ekspor paralel yang menyebabkan fallback internal ke ekspor Amazon RDS yang jauh lebih lambat.
- Memperbaiki masalah yang dapat menyebabkan masalah manajemen memori saat operasi query paralel pada tabel dengan bidang gumpalan.
- Memperbaiki masalah yang dapat menyebabkan peristiwa `Previous_gtids` binlog mengecualikan tertentu GTIDs dengan Enhanced Binlog diaktifkan dan `gtid_mode` disetel ke `ON` atau. `ON_PERMISSIVE`
- Memperbaiki masalah yang menyebabkan “Kesalahan penerusan penulisan internal” yang tidak terduga pada instance pembaca saat penerusan penulisan diaktifkan.
- Menyelesaikan kondisi balapan yang dapat menyebabkan pembacaan halaman salah dari kumpulan buffer selama operasi penskalaan tanpa server Aurora atau selama penggusuran halaman dari kumpulan buffer. Ref komunitas Bug #116305.
- Memperbaiki masalah saat mengekspor snapshot DB ke Amazon S3 akan lebih lambat saat mengekspor tabel yang berisi indeks sekunder dengan ekspresi yang dihasilkan.
- Memperbaiki masalah yang menyebabkan `AbortedClients` metrik tidak akurat saat beberapa koneksi terputus secara tak terduga.
- Menambahkan variabel global baru `aurora_lambda_request_timeout` untuk memungkinkan pengguna mengonfigurasi batas waktu AWS Lambda permintaan (default: 10 detik). Untuk informasi selengkapnya tentang menjalankan fungsi Lambda dari cluster DB MySQL Aurora, lihat Memanggil fungsi Lambda dari cluster DB MySQL Aurora.
- Memperbaiki masalah saat penerusan penulisan lokal berhenti bekerja setelah instance database dimulai ulang dengan restart zero-downtime.
- Memperbaiki masalah yang dapat menyebabkan operasi restart instance database gagal jika `max_user_connections` disetel ke nilai rendah.
- Peningkatan kinerja ekspor paralel dengan mengoptimalkan proses bootstrap untuk volume besar (> 64 TB), mengurangi waktu operasi ekspor secara keseluruhan.
- Memperbaiki masalah yang dapat menyebabkan instance penulis dimulai ulang saat melakukan `ALTER TABLE` paralel dengan kueri baca.

- Memperbaiki masalah yang dapat menyebabkan ekspor cluster DB memakan waktu lebih lama dari yang diharapkan ketika ada tabel yang lebih besar dari 14 TB.
- Memperbaiki masalah yang menyebabkan pelacakan permintaan query paralel yang tidak akurat saat menjalankan `EXPLAIN ANALYZE` pernyataan di mana `Aurora_pq_request_in_progress` penghitung tidak diperbarui secara akurat.
- Memperbaiki masalah yang, dalam kondisi yang tidak biasa, dapat menyebabkan instance database restart ketika volume database mendekati ukuran maksimum yang diizinkan.
- Memperbaiki masalah yang dapat menyebabkan instans database dimulai ulang saat memulihkan koneksi selama restart zero-downtime dan patching zero-downtime.

Integrasi perbaikan bug MySQL Community Edition

Rilis ini mencakup semua perbaikan bug komunitas hingga dan termasuk 8.0.43. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 3.x.

- Memperbaiki masalah di mana kueri formulir `SELECT 1 FROM t WHERE CAST(a AS UNSIGNED INTEGER) = 1 AND a = (SELECT 1 FROM t)` menyebabkan kegagalan pernyataan di. `item_func.cc` (Perbaikan Bug Komunitas #36128964)
- Permintaan formulir `SELECT 1 FROM t WHERE CAST(a AS UNSIGNED INTEGER) = 1 AND a = (SELECT 1 FROM t)` menyebabkan pernyataan di. `item_func.cc` (Bug #36128964)

Untuk informasi selengkapnya tentang perubahan komunitas, lihat Catatan Rilis MySQL 8.0.43.

Pembaruan mesin database Aurora MySQL 2026-01-02 (versi 3.10.3, kompatibel dengan MySQL 8.0.42)

Versi: 3.10.3

Aurora MySQL 3.10.3 umumnya tersedia. Versi Aurora MySQL 3.10 kompatibel dengan MySQL 8.0.42. Untuk informasi selengkapnya tentang perubahan komunitas yang telah terjadi, lihat Catatan Rilis MySQL 8.0.

Untuk detail fitur-fitur baru di Aurora MySQL versi 3, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0. Untuk perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2, lihat

Perbandingan Aurora MySQL versi 2 dan Aurora MySQL versi 3. Untuk perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition, lihat Perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition di Panduan Pengguna Amazon Aurora.

Anda dapat melakukan pemutakhiran di tempat dengan memanfaatkan Zero Downtime Patching (ZDP), memulihkan snapshot, atau memulai pemutakhiran terkelola blue/green menggunakan Penyebaran Amazon RDS dari Blue/Green klaster Aurora MySQL versi 2 yang saat ini didukung ke dalam klaster Aurora MySQL versi 3.10.3.

Untuk informasi tentang merencanakan peningkatan ke Aurora MySQL versi 3, lihat Merencanakan peningkatan versi utama untuk klaster MySQL Aurora. Untuk informasi umum tentang peningkatan Aurora MySQL, lihat Memutakhirkan klaster DB MySQL Aurora di Panduan Pengguna Amazon Aurora.

Untuk informasi pemecahan masalah, lihat Pemecahan Masalah untuk peningkatan Aurora MySQL di tempat di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah, Dukungan tersedia di forum komunitas dan melalui Dukungan. Untuk informasi selengkapnya, lihat Memelihara klaster DB Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan umum:

- Mengurangi overhead penggunaan CPU saat membuat Enkripsi di Transit antara instance database dan lapisan penyimpanan.
- Memperbaiki masalah yang dapat menyebabkan beberapa pernyataan SQL tidak masuk log dalam log audit.

Pembaruan mesin database Aurora MySQL 2025-11-20 (versi 3.10.2, kompatibel dengan MySQL 8.0.42)

Versi: 3.10.2

Aurora MySQL 3.10.2 umumnya tersedia. Versi Aurora MySQL 3.10 kompatibel dengan MySQL 8.0.42. Untuk informasi selengkapnya tentang perubahan komunitas yang telah terjadi, lihat Catatan Rilis MySQL 8.0.

Untuk detail fitur-fitur baru di Aurora MySQL versi 3, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0. Untuk perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2, lihat Perbandingan Aurora MySQL versi 2 dan Aurora MySQL versi 3. Untuk perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition, lihat Perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition di Panduan Pengguna Amazon Aurora.

Anda dapat melakukan pemutakhiran di tempat dengan memanfaatkan Zero Downtime Patching (ZDP), memulihkan snapshot, atau memulai pemutakhiran terkelola blue/green menggunakan Penyebaran Amazon RDS dari Blue/Green klaster Aurora MySQL versi 2 yang saat ini didukung ke dalam klaster Aurora MySQL versi 3.10.2.

Untuk informasi tentang merencanakan peningkatan ke Aurora MySQL versi 3, lihat Merencanakan peningkatan versi utama untuk klaster MySQL Aurora. Untuk informasi umum tentang peningkatan Aurora MySQL, lihat Memutakhirkan klaster DB MySQL Aurora di Panduan Pengguna Amazon Aurora.

Untuk informasi pemecahan masalah, lihat Pemecahan Masalah untuk peningkatan Aurora MySQL di tempat di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah,Dukungan tersedia di forum komunitas dan melalui Dukungan. Untuk informasi selengkapnya, lihat Memelihara klaster DB Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan ketersediaan:

- Memperbaiki masalah yang dapat menyebabkan engine restart selama Zero-downtime patching (ZDP) /Zero-downtime restart (ZDR), saat mempertahankan koneksi dengan transaksi aktif. SSL/TLS
- Memperbaiki masalah yang, dapat menyebabkan mesin restart saat berjalan `KILL <query-id>` setelah berjalan `EXPLAIN FOR CONNECTION <query-id>` pada Kueri Paralel yang sedang berjalan.
- Memperbaiki masalah yang dapat menyebabkan penulis menjadi tidak responsif saat penerusan penulisan dinonaktifkan sementara dua atau lebih replika meneruskan permintaan.
- Memperbaiki masalah yang dapat menyebabkan instance penulis dimulai ulang saat penerusan penulisan global atau penerusan penulisan lokal dinonaktifkan.

Perbaikan umum:

- Memperbaiki masalah yang menyebabkan pelacakan permintaan Kueri Paralel yang tidak akurat saat menjalankan `EXPLAIN ANALYZE` pernyataan di mana `Aurora_pq_request_in_progress` penghitung tidak diperbarui secara akurat.
- Secara otomatis menonaktifkan `aurora_oom_response` tindakan (kecuali cetak, jika dikonfigurasi) ketika `aurora_oom_response` gagal menyelesaikan tekanan memori setelah waktu ambang batas (dalam urutan beberapa menit).
- Peningkatan kinerja menulis IOPS ketika variabel sistem `innodb_flush_log_at_trx_commit` diatur ke`0`.
- Memperbaiki masalah dengan replika yang salah di-restart saat bergabung dengan penulis.
- Memperbaiki masalah di mana penulisan ke database mungkin terhenti saat menjalankan transaksi yang berjalan lama yang mengarah ke restart basis data atau upgrade versi utama gagal.

Pembaruan mesin database Aurora MySQL 2025-09-30 (versi 3.10.1, kompatibel dengan MySQL 8.0.42)

Versi: 3.10.1

Aurora MySQL 3.10.1 umumnya tersedia. Versi Aurora MySQL 3.10 kompatibel dengan MySQL 8.0.42. Untuk informasi selengkapnya tentang perubahan komunitas yang telah terjadi, lihat Catatan Rilis MySQL 8.0.

Untuk detail fitur-fitur baru di Aurora MySQL versi 3, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0. Untuk perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2, lihat Perbandingan Aurora MySQL versi 2 dan Aurora MySQL versi 3. Untuk perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition, lihat Perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition di Panduan Pengguna Amazon Aurora.

Anda dapat melakukan pemutakhiran di tempat dengan memanfaatkan Zero Downtime Patching (ZDP), memulihkan snapshot, atau memulai pemutakhiran terkelola blue/green menggunakan Penyebaran Amazon RDS dari Blue/Green klaster Aurora MySQL versi 2 yang saat ini didukung ke dalam klaster Aurora MySQL versi 3.10.1.

Untuk informasi tentang merencanakan peningkatan ke Aurora MySQL versi 3, lihat Merencanakan peningkatan versi utama untuk klaster MySQL Aurora. Untuk informasi umum tentang peningkatan

Aurora MySQL, lihat Meningkatkan klaster DB Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Untuk informasi pemecahan masalah, lihat Pemecahan Masalah untuk peningkatan Aurora MySQL di tempat di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan ketersediaan:

- Memperbaiki masalah dengan urutan penguncian halaman yang dikirim ke instance pembaca. Ini dapat menyebabkan instance pembaca memulai ulang karena deadlatch saat menjalankan kueri SELECT di atas tabel yang diperbarui oleh penulisnya.
- Memperbaiki masalah yang dapat menyebabkan restart instance pembaca gagal saat penulis menjalankan sejumlah besar operasi DDL.
- Memperbaiki masalah yang dapat menyebabkan instance Aurora MySQL Serverless v2 dimulai ulang saat parameter innodb_purge_threads dikonfigurasi secara manual ke nilai yang berbeda dari default. Parameter innodb_purge_threads sekarang dikelola secara otomatis untuk instance Aurora Serverless v2 dan tidak dapat dimodifikasi.
- Memperbaiki masalah dengan penskalaan Aurora Serverless V2 yang mengakibatkan instans DB dimulai ulang dengan mencegah halaman memori kritis ditukar.
- Memperbaiki masalah saat penghindaran Aurora kehabisan memori (OOM) tidak mempertahankan nilai parameter DB aurora_oom_response yang dikonfigurasi setelah database restart.
- Memperbaiki kondisi balapan yang dapat menyebabkan masalah ketersediaan setelah failover di klaster multi-AZ.
- Memperbaiki masalah dalam pemulihan logis dengan aurora_enhanced_binlog diaktifkan yang dapat mencegah restart basis data jika transaksi dibatalkan.
- Menyelesaikan masalah di mana instance database bisa menjadi tidak responsif selama beban kerja Parallel Query (PQ).
- Memperbaiki masalah yang mencegah pengguna dengan hak istimewa CONNECTION_ADMIN atau SUPER melakukan koneksi tambahan di luar batas max_connections, seperti yang didukung di MySQL Community Edition.

Perbaikan umum:

- Memperbaiki masalah yang dapat menyebabkan server database berkelanjutan dimulai ulang ketika volume diizinkan untuk tumbuh ke ukuran maksimumnya karena jenis kueri tertentu yang keliru diizinkan setelah volume lebih besar dari ambang batas tertentu.
- Memperbaiki masalah yang dapat menyebabkan instance penulis menjadi tidak responsif jika instance pembaca dimulai ulang saat menggunakan Global Write Forwarding atau Local Write Forwarding.
- Memperbaiki masalah yang dapat menyebabkan instance penulis dimulai ulang saat melakukan ALTER TABLE secara paralel dengan kueri baca.
- Memperbaiki masalah untuk meningkatkan ketersediaan dengan bootstrap volume besar (> 64 TB) selama operasi ekspor paralel.
- Memperbaiki masalah yang dapat menyebabkan operasi restart instance database gagal jika max_user_connections disetel ke nilai rendah.
- Memperbaiki masalah yang menyebabkan AbortedClients metrik tidak akurat saat beberapa koneksi terputus secara tak terduga.
- Menyelesaikan kondisi balapan yang dapat menyebabkan pembacaan halaman salah dari kumpulan buffer selama operasi penskalaan tanpa server Aurora atau selama penggusuran halaman dari kumpulan buffer. Ref komunitas Bug #116305.
- Memperbaiki masalah yang menyebabkan “Kesalahan penerusan penulisan internal” yang tidak terduga pada instance pembaca saat penerusan penulisan diaktifkan.
- Memperbaiki masalah yang dapat menyebabkan peristiwa binlog previous_GTIDS mengecualikan tertentu GTIDs dengan Enhanced Binlog diaktifkan dan gtid_mode disetel ke ON atau ON_PERMISSIVE.
- Memperbaiki masalah yang dapat menyebabkan masalah manajemen memori saat operasi query paralel pada tabel dengan bidang gumpalan.
- Memperbaiki masalah di mana kolom dengan pembaruan JSON sebagian akan gagal Ekspor Paralel menyebabkan fallback internal ke Ekspor RDS yang jauh lebih lambat.
- Memperbaiki masalah yang dapat menyebabkan instance tak terduga dimulai ulang ketika ada beban kerja penerusan penulisan bersamaan yang tinggi.
- Memperbaiki masalah di mana Zero Downtime Patching (ZDP) /Zero Downtime Restart (ZDR) dapat mengakibatkan instans DB restart saat memulihkan peringatan dengan kode kesalahan yang tidak valid.

- Memperbaiki masalah di mana file log biner sementara tidak dibersihkan dengan benar setelah rollback transaksi saat menggunakan logging biner. Perbaikan ini mencegah konsumsi penyimpanan yang tidak perlu dari file sementara yang disimpan dan dalam kasus tertentu juga dapat mencegah anomali dalam file log biner.

Integrasi perbaikan bug MySQL Community Edition

Rilis ini mencakup semua perbaikan bug komunitas hingga dan termasuk 8.0.42. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 3.x.

- Kueri formulir SELECT 1 FROM t WHERE CAST (A AS UNSIGNED INTEGER) = 1 AND a = (SELECT 1 FROM t) menyebabkan pernyataan di item_func.cc. (Bug #36128964)

Pembaruan mesin database Aurora MySQL 2025-07-31 (versi 3.10.0, kompatibel dengan MySQL 8.0.42)

Versi: 3.10.0

Aurora MySQL 3.10.0 umumnya tersedia. Versi Aurora MySQL 3.10 kompatibel dengan MySQL 8.0.42. Untuk informasi selengkapnya tentang perubahan komunitas yang telah terjadi, lihat Catatan Rilis MySQL 8.0.

> **Note**
>
> Versi ini ditetapkan sebagai rilis dukungan jangka panjang (LTS). Untuk informasi selengkapnya, lihat Rilis dukungan jangka panjang (LTS) Aurora MySQL di Panduan Pengguna Amazon Aurora.
> Kami menyarankan agar Anda tidak menyetel `AutoMinorVersionUpgrade` parameter ke `true` (atau mengaktifkan peningkatan versi minor Otomatis di Konsol Manajemen AWS) untuk versi LTS. Melakukannya dapat menyebabkan cluster DB Anda ditingkatkan ke versi target berikutnya untuk kampanye Peningkatan Versi Kecil Otomatis, yang mungkin bukan versi LTS.

Untuk detail fitur-fitur baru di Aurora MySQL versi 3, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0. Untuk perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2, lihat

Perbandingan Aurora MySQL versi 2 dan Aurora MySQL versi 3. Untuk perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition, lihat Perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition di Panduan Pengguna Amazon Aurora.

Anda dapat melakukan pemutakhiran di tempat dengan memanfaatkan Zero Downtime Patching (ZDP), memulihkan snapshot, atau memulai pemutakhiran terkelola blue/green menggunakan Penyebaran Amazon RDS dari Blue/Green klaster Aurora MySQL versi 2 yang saat ini didukung ke dalam klaster Aurora MySQL versi 3.10.0.

Untuk informasi tentang merencanakan peningkatan ke Aurora MySQL versi 3, lihat Merencanakan peningkatan versi utama untuk klaster MySQL Aurora. Untuk informasi umum tentang peningkatan Aurora MySQL, lihat Meningkatkan klaster DB Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Untuk informasi pemecahan masalah, lihat Pemecahan Masalah untuk peningkatan Aurora MySQL di tempat di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Aurora di Panduan Pengguna Amazon Aurora.

Fitur baru

- Aurora MySQL versi 3.10 memperluas dukungan cache log relai dalam memori untuk replika log biner. Fitur ini, pertama kali diperkenalkan di versi 3.05, dapat meningkatkan throughput replikasi log biner hingga 40%. Cache log relai dalam memori diaktifkan secara default untuk replikasi log biner ulir tunggal, replikasi multi-utas dengan pemosisian otomatis GTID diaktifkan, dan dimulai dengan versi 3.10, ini juga diaktifkan untuk replikasi multi-utas dengan (bahkan tanpa). `replica_preserve_commit_order = ON` GTIDs Anda dapat mengontrol fitur ini menggunakan parameter baru di 3.10, `aurora_in_memory_relaylog`. Untuk informasi selengkapnya, lihat Optimasi log biner di Aurora MySQL.
- Amazon Aurora telah menggandakan kapasitas penyimpanan maksimumnya dari 128 TiB menjadi 256 TiB, memungkinkan beban kerja yang lebih besar dalam satu cluster database. Untuk menggunakan batas penyimpanan yang ditingkatkan dengan Aurora MySQL, tingkatkan klaster Anda ke versi 3.10 (kompatibel dengan MySQL 8.0.42) atau lebih tinggi. Setelah upgrade, penyimpanan Aurora secara otomatis menskalakan hingga 256 TiB berdasarkan jumlah data dalam volume cluster Anda.

Perbaikan

Perbaikan keamanan

Sedang CVEs:

- CVE-2025-21501
- CVE-2025-21500
- CVE-2025-21543
- CVE-2025-21540
- CVE-2025-21491
- CVE-2025-21490
- CVE-2025-21559
- CVE-2025-21555
- CVE-2025-21497
- CVE-2025-21519
- CVE-2025-21529
- CVE-2025-21505
- CVE-2025-21531
- CVE-2025-21523
- CVE-2025-21503
- CVE-2025-21522
- CVE-2025-21518
- CVE-2025-21577
- CVE-2025-30682
- CVE-2025-30687
- CVE-2025-30688
- CVE-2025-21574
- CVE-2025-21575
- CVE-2025-30693
- CVE-2025-30695
- CVE-2025-30715

- CVE-2025-21584
- CVE-2025-21580
- CVE-2025-21581
- CVE-2025-21585
- CVE-2025-30689
- CVE-2025-21579
- CVE-2025-30696
- CVE-2025-30705
- CVE-2025-30683
- CVE-2025-30684
- CVE-2025-30685
- CVE-2025-30699
- CVE-2025-30704
- CVE-2025-30721

Rendah CVEs:

- CVE-2025-21520
- CVE-2025-21546
- CVE-2025-30703
- CVE-2025-30681

Perbaikan ketersediaan:

- Memperbaiki masalah yang menyebabkan restart server database yang tidak perlu yang terjadi karena penilaian kemajuan pemulihan yang salah.

Perbaikan umum:

- Mengurangi latensi komit saat I/O dioptimalkan diaktifkan.
- Memperbaiki masalah yang menyebabkan instance pembaca tidak menghasilkan log kesalahan saat penerusan penulisan diaktifkan dan parameter `aurora_replica_read_consistency` diubah.

- Memperbaiki masalah yang dapat menyebabkan instance pembaca dimulai ulang karena interaksi antara utas replikasi dan kueri yang mengakses tabel yang tidak ada di cache buffer.
- Memperbaiki masalah saat penerusan penulisan lokal berhenti bekerja setelah instance database dimulai ulang dengan restart zero-downtime.
- Memperbaiki masalah dalam penerusan tulis di mana kueri yang diteruskan mungkin berperilaku tidak benar untuk pernyataan yang berisi opsi set.
- Memperbaiki masalah stabilitas saat memasukkan metadata untuk undo tablespace akan memicu restart database yang tidak terduga.
- Memperbaiki masalah yang dapat menyebabkan ekspor cluster DB memakan waktu lebih lama dari yang diharapkan ketika ada tabel yang lebih besar dari 14 TB.
- Memperbaiki masalah yang dapat menyebabkan pelaporan variabel `Innodb_buffer_pool_pages_misc` status yang salah.
- Menambahkan dukungan untuk melestarikan `LAST_INSERT_ID` selama zero-downtime patching (ZDP) atau zero-downtime restart (ZDR).
- Aurora MySQL menggunakan nilai 8-bit untuk indeks virtual untuk IDs mencegah masalah format undo MySQL, karena melebihi batas ini dapat menyebabkan tidak tersedianya cluster. Saat mendekati batas ini, sistem sekarang menulis pesan peringatan ke log kesalahan MySQL. Jika batas tercapai, upaya untuk menambahkan indeks baru mengembalikan kesalahan. Untuk informasi selengkapnya tentang praktik terbaik indeks virtual, lihat Kesalahan luapan ID indeks virtual dalam dokumentasi MySQL Aurora.

Integrasi perbaikan bug MySQL Community Edition

Rilis ini mencakup semua perbaikan bug komunitas hingga dan termasuk 8.0.42. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 3.x.

- Keluar dari server dapat dihasilkan dari upaya simultan oleh beberapa utas untuk mendaftarkan dan membatalkan pendaftaran objek Skema Kinerja metadata, atau untuk memperoleh dan melepaskan kunci metadata. (Bug #26502135)

Pembaruan mesin database Aurora MySQL 2025-05-14 (versi 3.09.0, kompatibel dengan MySQL 8.0.40)

Versi: 3.09.0

Aurora MySQL 3.09.0 umumnya tersedia. Versi Aurora MySQL 3.09 kompatibel dengan MySQL 8.0.40. Untuk informasi selengkapnya tentang perubahan komunitas yang telah terjadi, lihat Catatan Rilis MySQL 8.0.

Untuk detail fitur-fitur baru di Aurora MySQL versi 3, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0. Untuk perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2, lihat Perbandingan Aurora MySQL versi 2 dan Aurora MySQL versi 3. Untuk perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition, lihat Perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition di Panduan Pengguna Amazon Aurora.

Anda dapat melakukan pemutakhiran di tempat yang memanfaatkan zero-downtime-patch, memulihkan snapshot, atau memulai Blue/Green pemutakhiran terkelola blue/green menggunakan Penyebaran Amazon RDS dari klaster Aurora MySQL versi 2 yang saat ini didukung ke dalam klaster Aurora MySQL versi 3.09.0.

Untuk informasi tentang merencanakan peningkatan ke Aurora MySQL versi 3, lihat Merencanakan peningkatan versi utama untuk klaster MySQL Aurora. Untuk informasi umum tentang peningkatan Aurora MySQL, lihat Meningkatkan klaster DB Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Untuk informasi pemecahan masalah, lihat Pemecahan Masalah untuk peningkatan Aurora MySQL di tempat di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB di Panduan Pengguna Amazon Aurora.

Sorotan rilis

- Basis data global MySQL Aurora yang disempurnakan untuk memungkinkan instance pembaca sekunder menyelesaikan startup dan melayani permintaan baca selama peristiwa yang tidak direncanakan (kegagalan perangkat keras, gangguan jaringan). Sebelumnya, instance pembaca sekunder tidak dapat dimulai ulang selama acara tersebut. Untuk informasi selengkapnya, lihat Ketahanan Lintas Wilayah untuk klaster sekunder Database Global di Panduan Pengguna Amazon Aurora.
- Mengurangi waktu henti penulis selama pengalihan lintas wilayah Aurora MySQL Global Database menjadi biasanya di bawah satu menit, meminimalkan waktu henti selama sakelar regional yang direncanakan.

Perbaikan

Perbaikan keamanan

Kritis CVEs:

- CVE-2024-11053
- CVE-2024-37371

Sedang CVEs:

- CVE-2024-7264
- CVE-2024-21193
- CVE-2024-21194
- CVE-2024-21196
- CVE-2024-21197
- CVE-2024-21198
- CVE-2024-21199
- CVE-2024-21201
- CVE-2024-21203
- CVE-2024-21207
- CVE-2024-21212
- CVE-2024-21213
- CVE-2024-21218
- CVE-2024-21219
- CVE-2024-21230
- CVE-2024-21236
- CVE-2024-21238
- CVE-2024-21239
- CVE-2024-21241
- CVE-2025-21494
- CVE-2025-21504
- CVE-2025-21525

- CVE-2025-21534
- CVE-2025-21536

Perbaikan ketersediaan:

- Memperbaiki masalah di mana kueri multi-pernyataan yang diteruskan dari pembaca ke instance penulis dapat menggantung ketika `innodb_flush_log_at_trx_commit` disetel ke penulis dan bukan nol `0` pada pembaca, mencegah potensi kegagalan penerusan penulisan.
- Memperbaiki masalah kebuntuan dengan Enhanced Binlog diaktifkan yang dapat menyebabkan reboot database `SHOW BINARY LOGS` saat berjalan bersamaan dengan melakukan transaksi ke mesin BLACKHOLE atau mengeksekusi `XA PREPARE` pernyataan, mencegah potensi penulisan macet dan masalah ketersediaan instance.
- Kondisi balapan tetap dalam penerusan tulis yang dapat menyebabkan instance penulis Aurora restart dengan mencegah permintaan baru diterima sebelum permintaan sebelumnya sepenuhnya selesai, meningkatkan stabilitas operasi penerusan tulis.
- Memperbaiki masalah pada replika di mana gangguan jaringan mungkin tidak membangun kembali koneksi dengan penulis dengan benar yang menyebabkan replikasi macet dan potensi instance dimulai ulang.
- Respons Aurora MySQL Out of Memory (OOM) sekarang mengimplementasikan pengubahan ukuran kumpulan buffer bertahap yang secara bertahap mengurangi penggunaan memori berdasarkan status memori sistem (LOW/RESERVED) saat diaktifkan melalui `aurora_oom_response` parameter DB, memberikan manajemen memori yang lebih baik selama situasi tekanan memori.
- Meningkatkan waktu pemulihan file Binlog selama restart database dengan mengoptimalkan proses pemulihan untuk mengambil waktu konstan terlepas dari ukuran file Binlog. Sebelumnya, dalam beberapa kasus, waktu pemulihan akan sebanding dengan ukuran file Binlog terakhir.
- Memperbaiki masalah yang dapat menyebabkan server MySQL yang tidak terduga dimulai ulang saat menjalankan operasi pemotongan tabel InnoDB bersamaan saat melakukan kueri. `performance_schema.data_lock_waits`
- Memperbaiki masalah yang dapat menyebabkan instance database dimulai ulang saat melakukan peristiwa binlog besar selama kondisi penyimpanan rendah.
- Memperbaiki masalah di mana operasi pengubahan ukuran kumpulan buffer yang dipicu selama penghindaran Kehabisan Memori (OOM) dapat menjadi tidak responsif selama skenario beban kerja tinggi, yang mengarah ke kemungkinan restart basis data.

- Memperbaiki masalah yang dapat menyebabkan loop restart database saat membuat pemicu. Masalah ini juga dapat terjadi ketika Binlog baru atau file Relaylog ditambahkan atau file-file ini diputar.
- Memperbaiki masalah yang dapat menyebabkan instance pembaca Aurora dimulai ulang saat menggunakan penerusan tulis dengan kueri komit multi-pernyataan atau implisit.

Perbaikan umum:

- Memperbaiki masalah di mana `ALTER TABLE ... REBUILD / OPTIMIZE TABLE` operasi dapat menghabiskan memori yang berlebihan dengan mengalokasikan `innodb_ddl_buffer_size` byte per utas DDL alih-alih membagi ukuran buffer di antara utas, mencegah potensi penggunaan memori selama operasi DDL.
- Mengubah nilai default untuk`aurora_oom_response`, pada semua kelas instans DB yang memiliki lebih dari 4 GiB memori, dari cetak ke cetak, tolak, kill_connect. Untuk informasi selengkapnya, lihat masalah Amazon Aurora MySQL di Panduan Pengguna out-of-memory Amazon Aurora.
- Hak istimewa berikut telah ditambahkan ke`rds_superuser_role`:`FLUSH_OPTIMIZER_COSTS`,, `FLUSH_STATUS`FLUSH_TABLES,`FLUSH_USER_RESOURCES`. Untuk informasi tentang ini`rds_superuser_role`, lihat Akun Pengguna Amazon Master dengan dokumentasi Amazon Aurora. Untuk informasi lebih lanjut tentang hak istimewa dinamis ini, silakan lihat dokumentasi MySQL.
- Dimulai dengan versi Aurora MySQL ini, optimasi sisipan cepat tidak lagi diaktifkan. Untuk informasi selengkapnya, lihat peningkatan kinerja Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Memperbaiki masalah dengan pelanggaran `max_user_connections` ambang batas yang salah, yang mengakibatkan kesalahan koneksi untuk beberapa pengguna. Ini terjadi dalam beberapa kasus tepi, seperti ketika koneksi dibuat dan dibunuh segera.
- Memperbaiki masalah pencatatan audit yang menyebabkan pemanfaatan CPU tinggi, yang mengarah ke instance server database yang tidak responsif.
- Memperbaiki masalah manajemen memori saat menggunakan transaksi XA, mencegah kemungkinan restart instance saat Enhanced Binlog diaktifkan.
- Memperbaiki masalah di mana kinerja kueri akan menurun ketika pengoptimal membuat estimasi biaya yang salah karena Statistik Indeks Bufferpool diperbarui secara tidak benar setelah server database dimulai ulang.

- Memperbaiki masalah yang mencegah pelanggan mematikan fungsionalitas penerusan tulis lokal karena utas pekerja macet.
- Memperbaiki masalah yang menyebabkan `SHOW BINARY LOGS` perintah memakan waktu lebih lama untuk dijalankan di cluster tempat Enhanced Binlog diaktifkan atau sebelumnya diaktifkan. Masalah ini juga dapat menyebabkan peningkatan latensi komit jika beberapa `SHOW BINARY LOGS` perintah berjalan secara bersamaan.

Peningkatan dan migrasi:

- Memperbaiki masalah di mana Zero Downtime Patching (ZDP) mungkin tidak berhasil saat mencoba mempertahankan koneksi, milik pengguna yang telah dijatuhkan. Informasi lebih lanjut tentang `DROP USER` perintah, dan pengaruhnya pada koneksi aktif dapat ditemukan di dokumentasi MySQL.

Integrasi perbaikan bug MySQL Community Edition

Rilis ini mencakup semua perbaikan bug komunitas hingga dan termasuk 8.0.40. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 3.x.

- Sementara transaksi besar diterima dan diterapkan, dan permintaan untuk menghentikan saluran replikasi dibuat menggunakan`STOP REPLICA`, MySQL tidak melakukannya dengan benar, dan kemudian tidak memproses perintah saluran apa pun. Selain itu, proses shutdown server tidak selesai dengan anggun, dan mengharuskan proses MySQL dimatikan atau sistem host untuk di-restart. (Bug #115966, Bug #37008345)

Pembaruan mesin database Aurora MySQL 2025-04-07 (versi 3.08.2, kompatibel dengan MySQL 8.0.39)

Versi: 3.08.2

Aurora MySQL 3.08.2 umumnya tersedia. Versi Aurora MySQL 3.08 kompatibel dengan MySQL 8.0.39. Untuk informasi selengkapnya tentang perubahan komunitas yang telah terjadi, lihat Catatan Rilis MySQL 8.0.

Untuk detail fitur-fitur baru di Aurora MySQL versi 3, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0. Untuk perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2, lihat Perbandingan Aurora MySQL versi 2 dan Aurora MySQL versi 3. Untuk perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition, lihat Perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition di Panduan Pengguna Amazon Aurora.

Rilis Aurora MySQL yang didukung saat ini adalah 2.11.*, 2.12.*, 3.04.*, 3.05.*, 3.06.*, 3.07.*, dan 3.08.*.

Anda dapat melakukan upgrade di tempat, memulihkan snapshot, atau memulai upgrade terkelola blue/green menggunakan Amazon RDS Blue/Green Deployment dari klaster Aurora MySQL versi 2 yang didukung saat ini ke dalam klaster Aurora MySQL versi 3.08.2.

Untuk informasi tentang merencanakan peningkatan ke Aurora MySQL versi 3, lihat Merencanakan peningkatan versi utama untuk klaster MySQL Aurora. Untuk informasi umum tentang peningkatan Aurora MySQL, lihat Meningkatkan klaster DB Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Untuk informasi pemecahan masalah, lihat Pemecahan Masalah untuk peningkatan Aurora MySQL di tempat di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan keamanan:

- CVE-2024-11053

Perbaikan ketersediaan:

- Memperbaiki masalah pada replika di mana gangguan jaringan mungkin tidak membangun kembali koneksi dengan penulis dengan benar.
- Memperbaiki masalah di mana kueri multi-pernyataan yang diteruskan dapat gagal jika `innodb_flush_log_at_trx_commit` disetel ke instance penulis`0`.
- Memperbaiki masalah yang dapat menyebabkan instans DB penulis Aurora dimulai ulang saat penerusan penulisan diaktifkan.

- Memperbaiki masalah yang dapat menyebabkan instans DB pembaca Aurora dimulai ulang saat menggunakan penerusan tulis dengan kueri komit multi-pernyataan atau implisit.

Perbaikan umum

- Dimulai dengan versi Aurora MySQL ini, optimasi sisipan cepat tidak lagi diaktifkan. Untuk informasi selengkapnya, lihat peningkatan kinerja Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Memperbaiki masalah dengan pelanggaran `max_user_connections` ambang batas yang salah, yang mengakibatkan kesalahan koneksi untuk beberapa pengguna. Ini terjadi dalam beberapa kasus tepi, seperti ketika koneksi dibuat dan dibunuh segera.

Integrasi perbaikan bug MySQL Community Edition

Rilis ini mencakup semua perbaikan bug komunitas hingga dan termasuk 8.0.39, selain yang berikut ini. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 3.x.

- Sementara transaksi besar sedang diterima dan diterapkan, dan permintaan untuk menghentikan saluran replikasi dibuat menggunakan`STOP REPLICA`, MySQL tidak melakukannya dengan benar, dan kemudian tidak memproses perintah saluran apa pun. Selain itu, proses shutdown server tidak selesai dengan anggun, dan mengharuskan proses MySQL dimatikan atau sistem host untuk di-restart. (Perbaikan Bug Komunitas #115966 dan #37008345)

Pembaruan mesin database Aurora MySQL 2025-02-17 (versi 3.08.1, kompatibel dengan MySQL 8.0.39)

Versi: 3.08.1

Aurora MySQL 3.08.1 umumnya tersedia. Aurora MySQL 3.08 versi kompatibel dengan MySQL 8.0.39. Untuk informasi selengkapnya tentang perubahan komunitas yang telah terjadi, lihat Catatan Rilis MySQL 8.0.

Untuk detail fitur-fitur baru di Aurora MySQL versi 3, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0. Untuk perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2, lihat Perbandingan Aurora MySQL versi 2 dan Aurora MySQL versi 3. Untuk perbandingan Aurora MySQL

versi 3 dan MySQL 8.0 Community Edition, lihat Perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition di Panduan Pengguna Amazon Aurora.

Rilis Aurora MySQL yang didukung saat ini adalah 2.11.*, 2.12.*, 3.04.*, 3.05.*, 3.06.*, 3.07.*, dan 3.08.*.

Anda dapat melakukan upgrade di tempat, memulihkan snapshot, atau memulai upgrade terkelola blue/green menggunakan Amazon RDS Blue/Green Deployment dari klaster Aurora MySQL versi 2 yang didukung saat ini ke dalam klaster Aurora MySQL versi 3.08.1.

Untuk informasi tentang merencanakan peningkatan ke Aurora MySQL versi 3, lihat Merencanakan peningkatan versi utama untuk klaster MySQL Aurora. Untuk informasi umum tentang peningkatan Aurora MySQL, lihat Meningkatkan klaster DB Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Untuk informasi pemecahan masalah, lihat Pemecahan Masalah untuk peningkatan Aurora MySQL di tempat di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan ketersediaan:

- Membuat perbaikan yang mengurangi waktu restart instans DB selama restart database, patch atau upgrade versi minor, dan failover.

Upgrade dan migrasi

- Memperbaiki masalah yang dapat menyebabkan kegagalan Zero Downtime Patching (ZDP) saat mencoba memulihkan peran dan hibah pengguna yang dijatuhkan. Untuk informasi selengkapnya tentang `DROP USER` operasi, lihat Pernyataan DROP USER di dokumentasi MySQL.

Integrasi perbaikan bug MySQL Community Edition

Rilis ini mencakup semua perbaikan bug komunitas hingga dan termasuk 8.0.39, selain yang berikut ini. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 3.x.

Pembaruan mesin database Aurora MySQL 2024-11-18 (versi 3.08.0, kompatibel dengan MySQL 8.0.39)

Versi: 3.08.0

Aurora MySQL 3.08.0 umumnya tersedia. Aurora MySQL 3.08 versi kompatibel dengan MySQL 8.0.39. Untuk informasi selengkapnya tentang perubahan komunitas yang telah terjadi, lihat Catatan Rilis MySQL 8.0.

Untuk detail fitur-fitur baru di Aurora MySQL versi 3, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0. Untuk perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2, lihat Perbandingan Aurora MySQL versi 2 dan Aurora MySQL versi 3. Untuk perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition, lihat Perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition di Panduan Pengguna Amazon Aurora.

Rilis Aurora MySQL yang didukung saat ini adalah 2.11.*, 2.12.*, 3.04.*, 3.05.*, 3.06.*, 3.07.*, dan 3.08.*.

Anda dapat melakukan pemutakhiran di tempat, memulihkan snapshot, atau memulai pemutakhiran terkelola blue/green menggunakan Amazon RDS Blue/Green Deployment dari klaster Aurora MySQL versi 2 yang saat ini didukung ke dalam klaster Aurora MySQL versi 3.08.0.

Untuk informasi tentang merencanakan peningkatan ke Aurora MySQL versi 3, lihat Merencanakan peningkatan versi utama untuk klaster MySQL Aurora. Untuk informasi umum tentang peningkatan Aurora MySQL, lihat Meningkatkan klaster DB Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Untuk informasi pemecahan masalah, lihat Pemecahan Masalah untuk peningkatan Aurora MySQL di tempat di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Fitur baru

- Menambahkan tiga CloudWatch metrik Amazon baru untuk memungkinkan pengguna memantau proses pembersihan InnoDB:

  - `PurgeBoundary`
  - `PurgeFinishedPoint`
  - `TruncateFinishedPoint`

  Untuk informasi selengkapnya, lihat CloudWatchmetrik Amazon untuk Amazon Aurora di Panduan Pengguna Amazon Aurora.
- Menambahkan CloudWatch metrik baru `TransactionAgeMaximum` untuk memungkinkan pengguna mengidentifikasi transaksi jangka panjang yang mungkin menahan batas pembersihan. Untuk informasi selengkapnya, lihat CloudWatch metrik Amazon untuk Amazon Aurora di Panduan Pengguna Amazon Aurora.
- Menambahkan tiga CloudWatch metrik baru untuk penghindaran out-of-memory (OOM):
  - `AuroraMillisecondsSpentInOomRecovery`
  - `AuroraNumOomRecoverySuccessful`
  - `AuroraNumOomRecoveryTriggered`

  Untuk informasi selengkapnya, lihat CloudWatchmetrik Amazon untuk Amazon Aurora di Panduan Pengguna Amazon Aurora.
- Mengubah tiga CloudWatch metrik untuk penghindaran out-of-memory (OOM) dari menjalankan total menjadi penghitung tambahan:
  - `AuroraMemoryNumDeclinedSqlTotal`
  - `AuroraMemoryNumKillConnTotal`
  - `AuroraMemoryNumKillQueryTotal`

  Untuk informasi selengkapnya, lihat CloudWatchmetrik Amazon untuk Amazon Aurora di Panduan Pengguna Amazon Aurora.
- Menambahkan dua variabel status global untuk menunjukkan jumlah memori yang digunakan oleh tabel sementara internal: `aurora_temptable_ram_allocation` dan`aurora_temptable_max_ram_allocation`. Variabel status global ini meningkatkan observabilitas dan membantu mendiagnosis masalah yang terkait dengan penggunaan memori tabel sementara internal.

  Untuk informasi selengkapnya, lihat Variabel status global Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Memperkenalkan variabel sistem baru `aurora_optimizer_trace_print_before_purge` untuk mencetak jejak pengoptimal ke log kesalahan sebelum server membersihkan jejak dari

memori. Pembersihan dapat dipicu berdasarkan ambang batas yang ditetapkan oleh variabel sistem optimizer_trace_offset dan optimizer_trace_limit.

- Ditambahkan dukungan untuk kelas instans DB berikut:
  - db.r7i
  - db.r8g

  Lihat informasi yang lebih lengkap di Mesin basis data yang didukung untuk kelas-kelas instans basis data dalam Panduan Pengguna Amazon Aurora.

Perbaikan

Memperbaiki masalah keamanan dan CVEs:

- Memperkenalkan pengguna baru untuk replikasi log biner (binlog),. `rdsrepladmin_priv_checks_user` Untuk informasi selengkapnya, lihat Privilege memeriksa pengguna untuk replikasi log biner di Panduan Pengguna Amazon Aurora.
- Memperbaiki masalah di mana parameter input ke prosedur tersimpan MySQL Aurora mungkin ditangani secara tidak benar.

Rilis ini mencakup semua perbaikan CVE komunitas hingga dan termasuk MySQL 8.0.39. Perbaikan CVE berikut disertakan:

- CVE-2023-44487
- CVE-2024-0853
- CVE-2024-20996
- CVE-2024-21000
- CVE-2024-21013
- CVE-2024-21125
- CVE-2024-21127
- CVE-2024-21129
- CVE-2024-21130
- CVE-2024-21134
- CVE-2024-21135
- CVE-2024-21137

- CVE-2024-21142
- CVE-2024-21157
- CVE-2024-21159
- CVE-2024-21160
- CVE-2024-21162
- CVE-2024-21163
- CVE-2024-21165
- CVE-2024-21166
- CVE-2024-21173
- CVE-2024-21177
- CVE-2024-21179
- CVE-2024-21185
- CVE-2024-25062
- CVE-2024-37371
- CVE-2025-21492

Perbaikan ketersediaan:

- Memperbaiki cacat yang dapat menyebabkan server restart secara tak terduga setelah menjalankan perintah Data Manipulation Language (DML/Data Manipulation Language) pada tabel yang sudah langsung menjatuhkan kolom, sehingga jumlah kolom yang dijatuhkan dan tidak dijatuhkan lebih besar dari 1017.
- Memperbaiki masalah yang dapat menyebabkan file log database tidak diputar dengan benar, sehingga meningkatkan penggunaan ruang penyimpanan lokal pada instans DB.
- Memperbaiki masalah yang dapat menyebabkan instans DB berhenti menutup koneksi dalam kondisi memori rendah saat aurora_oom_response diaktifkan, yang mengarah ke reboot yang sebaliknya dapat dihindari. out-of-memory
- Memperbaiki masalah yang dapat menyebabkan instans DB pembaca dimulai ulang saat membebaskan memori yang digunakan untuk aplikasi log.
- Memperbaiki masalah dalam menghitung metrik internal untuk indeks pencarian teks lengkap (FTS) yang dapat menyebabkan restart basis data.
- Memperbaiki masalah komunitas yang terkadang dapat menghasilkan hasil kueri yang salah saat `LIMIT` klausa digunakan dalam subkueri bersama dengan pushdown kondisi indeks.

- Memperbaiki masalah yang menyebabkan pelanggaran akses memori yang menyebabkan pelepasan objek mutex yang tidak lagi dimiliki oleh utas.
- Memperbaiki masalah yang dapat menyebabkan restart pada replika log biner (binlog) saat memproses sejumlah besar file log relai selama pemulihan log relai.
- Memperbaiki masalah yang dapat menyebabkan instans DB pembaca Aurora dimulai ulang saat penerusan penulisan diaktifkan.
- Memperbaiki masalah di mana kueri yang berisi beberapa `UNION` klausa dapat mengalokasikan sejumlah besar memori, yang mengarah ke restart instans DB.
- Memperbaiki masalah yang menyebabkan instans DB penulis memulai ulang saat menjalankan kueri paralel pada instans DB pembaca.
- Memperbaiki masalah di mana replikasi binlog akan terhenti pada replika karena kebuntuan yang ditemui pada I/O utas replika saat perintah dijalankan. `FLUSH RELAY LOGS`
- Memperbaiki masalah yang menyebabkan instans DB dimulai ulang saat menangani set GTID besar pada cluster DB dengan binlog yang ditingkatkan diaktifkan.
- Memperbaiki masalah yang dapat menyebabkan restart pada replika binlog saat cache log relai dalam memori diaktifkan. Cache log relai dalam memori diaktifkan pada replika binlog terkelola Aurora MySQL saat menggunakan replikasi log biner ulir tunggal, atau replikasi multithreaded dengan pemosisian otomatis GTID diaktifkan.
- Memperbaiki masalah yang dapat menyebabkan instance replika binlog dimulai ulang saat menerapkan pernyataan Data Control Language (DCL) selama startup mesin database.
- Memperbaiki masalah yang, dalam kondisi langka, dapat menyebabkan instans DB pembaca dimulai ulang karena deadlatch saat menjalankan `SELECT` kueri pada tabel yang diperbarui oleh instance DB penulisnya.
- Memperbaiki masalah yang dapat menyebabkan instance pembaca Aurora Global Database dimulai ulang dengan sesi penerusan tulis aktif.
- Memperbaiki masalah yang dapat menyebabkan replika baca Aurora dimulai ulang jika terjadi pesanan komit transaksi langka tertentu pada instans DB penulis.
- Memperbaiki masalah yang dapat menyebabkan database dimulai ulang saat acara terjadwal dibatalkan saat berjalan pada instans DB yang telah mengaktifkan binlog yang disempurnakan.

Perbaikan umum:

- Memperbaiki masalah di mana koneksi klien dapat macet selama zero-downtime patching (ZDP) atau zero-downtime restart (ZDR).

- Memperbaiki masalah yang, dalam kasus yang jarang terjadi, menyebabkan instance database dimulai ulang karena masalah manajemen memori yang dapat terjadi saat menangani kegagalan tabel terbuka.
- Memperbaiki masalah yang dapat menyebabkan pernyataan SQL mengalami kesalahan atau peringatan pelanggaran kunci primer yang tidak terduga pada beberapa baris saat melakukan `INSERT` pernyataan bersamaan pada tabel yang memiliki kolom kunci `AUTO_INCREMENT` utama dan kolom kunci unik, dan ketika `INSERT` pernyataan memiliki pelanggaran kunci unik pada baris yang berbeda.
- Instrumen Skema Kinerja `memory/sql/sp_head::main_mem_root` sekarang menjadi instrumen yang dikendalikan. Akibatnya, memori yang dialokasikan untuk penguraian dan representasi program yang disimpan sekarang berkontribusi terhadap batas memori per koneksi.
- Memperbaiki masalah saat baris menjadi tidak dapat dibaca melalui indeks spasial selama pembaruan.
- Memperbaiki masalah yang mencegah pengguna menonaktifkan penerusan penulisan lokal.
- Memperbaiki masalah di mana `SELECT COUNT` kueri dapat mengembalikan hasil yang salah saat menggunakan penerusan tulis.
- Memperbaiki masalah yang dapat menyebabkan hasil kueri yang salah ketika ZDP salah mengembalikan variabel sesi yang ditetapkan sebagai petunjuk dalam kueri.
- Memperbaiki masalah dengan pemotongan otomatis dari undo tablespaces ketika mereka lebih besar dari ambang innodb_max_undo_log_size dalam skenario peningkatan.
- Memperbaiki masalah saat latensi komit dan throughput komit tidak diukur saat `innodb_flush_log_at_trx_commit` disetel ke. `0`
- Memperbaiki masalah yang dapat menyebabkan kelambatan replika Aurora dilaporkan secara tidak benar.
- Memperbaiki masalah yang memberikan nilai yang salah untuk `threads_running` variabel saat menggunakan Aurora Global Database.
- Memperbaiki masalah di mana replika binlog Aurora MySQL dengan optimasi indeks sekunder paralel diaktifkan akan mengalami restart saat menerapkan perubahan replikasi pada tabel dengan kunci asing.
- Memperbaiki masalah restart yang disebabkan oleh pengubahan ukuran tabel hash kunci yang berkepanjangan selama penskalaan atau penskalaan peristiwa.
- Memperbaiki masalah yang dapat menyebabkan instans DB dimulai ulang setelah menonaktifkan logging biner, saat binlog yang disempurnakan sebelumnya diaktifkan.

- Memperkenalkan optimasi untuk mengurangi penggunaan memori selama pemulihan kamus data logis ketika ada sejumlah besar tabel.
- Memperbaiki masalah di mana pengguna mungkin mengalami `ERROR 1377 (HY000): Fatal error during log purge` kesalahan saat menjalankan prosedur tersimpan mysql.rds_set_external_source pada replika binlog, ketika replikasi log biner sudah dikonfigurasi.
- Memperbaiki cacat yang mencegah persistensi hak istimewa peran pengguna setelah ZDR.
- Memperbaiki masalah memori yang terkait dengan peran default dari view definer.
- Memperbaiki masalah yang dapat menyebabkan instans DB dimulai ulang saat `SHOW VOLUME STATUS` dijalankan.
- Memperbaiki masalah restart yang disebabkan oleh pengubahan ukuran kumpulan buffer yang berkepanjangan selama penskalaan atau penskalaan peristiwa.
- Memperbaiki masalah restart yang disebabkan oleh logical read ahead (LRA) yang mengakses halaman yang dibebaskan karena pengubahan ukuran buffer selama acara penskalaan turun.
- Memperbaiki masalah yang mengatasi `SELECT` kueri yang mengembalikan hasil yang salah saat kueri digunakan `LEFT OUTER JOIN` dengan transformasi semijoin yang menggunakan materialisasi sebagai strategi.
- Memperbaiki masalah yang dapat menyebabkan kegagalan dalam menyelesaikan proses menonaktifkan penerusan tulis.
- Memperbaiki masalah saat `BlockedTransactions` CloudWatch metrik `ActiveTransactions` dan melaporkan nilai yang lebih rendah dari yang diharapkan.
- Memperbaiki masalah saat replikasi binlog rusak saat replika memproses `DELETE` pernyataan multitable yang secara eksplisit dihapus dari tabel induk dan anak.
- Memperbaiki masalah yang dapat menyebabkan instans DB dimulai ulang saat memproses pemicu dengan fungsi yang ditentukan pengguna yang `enum` mengembalikan tipe.
- Memperbaiki masalah di mana `DMLLatency` CloudWatch metrik akan menunjukkan nilai yang salah untuk instance replika log biner saat mereplikasi menggunakan `binlog_format` set to. `row`
- Memperbaiki masalah di mana lambat`INSERT`,`DELETE`, dan `UPDATE` kueri yang dijalankan oleh Penjadwal Acara MySQL tidak direkam dalam log kueri lambat kecuali didahului oleh kueri lambat. `SELECT`
- Memperbaiki masalah yang, dalam kasus yang jarang terjadi, menyebabkan tidak tersedianya replika baca Aurora yang terputus-putus atau ketidakkonsistenan definisi tabel, terkadang dengan kesalahan, pada replika. `Table does not exist` Ini karena kueri baca bersamaan pada operasi replika dan Data Definition Language (DDL) pada instance DB penulis.

- Memperbaiki masalah yang menyebabkan `SHOW BINARY LOGS` perintah membutuhkan waktu lebih lama untuk berjalan di cluster DB di mana binlog yang disempurnakan diaktifkan atau sebelumnya diaktifkan. Masalah ini juga dapat menyebabkan peningkatan latensi komit jika beberapa `SHOW BINARY LOGS` perintah berjalan secara bersamaan.

Peningkatan dan migrasi:

- Meningkatkan kinerja upgrade versi utama dari Aurora MySQL versi 2 ke versi 3 untuk cluster DB dengan sejumlah besar objek database (seperti tabel, pemicu, dan rutinitas).

  Untuk kelas instans DB yang lebih besar, proses pemutakhiran database meningkatkan metadata objek MySQL tradisional ke kamus data MySQL 8.0 atom yang baru secara paralel menggunakan beberapa utas.
- Memperbaiki masalah yang menyebabkan peningkatan atau migrasi gagal saat target versi mesin Aurora MySQL DB adalah 3.04.0 atau lebih tinggi. Hal ini terjadi ketika parameter cluster `lower_case_table_names` DB diatur ke`1`, dan MySQL database collation tidak kompatibel dengan nama tabel huruf kecil.

Integrasi perbaikan bug MySQL Community Edition

Rilis ini mencakup semua perbaikan bug komunitas hingga dan termasuk 8.0.39, selain yang berikut ini. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 3.x.

- Memperbaiki masalah yang menyebabkan `NULL` nilai dihilangkan secara tidak benar dari kumpulan hasil untuk kueri tertentu yang memiliki keduanya `JOIN` dan operasi. `UNION` (Perbaikan Bug Komunitas #114301)

Pembaruan mesin database Aurora MySQL 2024-07-23 (versi 3.07.1) (Usang)

Versi: 3.07.1

Aurora MySQL 3.07.1 umumnya tersedia. Versi Aurora MySQL 3.07 kompatibel dengan MySQL 8.0.36. Untuk informasi selengkapnya tentang perubahan komunitas yang telah terjadi, lihat Catatan Rilis MySQL 8.0.

Untuk detail fitur-fitur baru di Aurora MySQL versi 3, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0. Untuk perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2, lihat Membandingkan Aurora MySQL versi 2 dan Aurora MySQL versi 3. Untuk perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition, lihat Membandingkan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition di Panduan Pengguna Amazon Aurora.

Rilis Aurora MySQL yang didukung saat ini adalah 2.11.*, 2.12.*, 3.03.*, 3.04.*, 3.05.*, 3.06.*, dan 3.07.*.

Anda dapat melakukan upgrade di tempat, memulihkan snapshot, atau memulai upgrade terkelola blue/green menggunakan Amazon RDS Blue/Green Deployment dari klaster Aurora MySQL versi 2 yang didukung saat ini ke dalam klaster Aurora MySQL versi 3.07.1.

Untuk informasi tentang merencanakan peningkatan ke Aurora MySQL versi 3, lihat Merencanakan peningkatan versi utama untuk klaster MySQL Aurora. Untuk informasi umum tentang peningkatan Aurora MySQL, lihat Meningkatkan klaster DB Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Untuk informasi pemecahan masalah, lihat Pemecahan Masalah untuk peningkatan Aurora MySQL di tempat di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Memperbaiki masalah keamanan dan CVEs:

- Memperkenalkan pengguna baru untuk replikasi log biner (binlog),. `rdsrepladmin_priv_checks_user` Untuk informasi selengkapnya, lihat Privilege memeriksa pengguna untuk replikasi log biner di Panduan Pengguna Amazon Aurora.

Rilis ini mencakup semua perbaikan CVE komunitas hingga dan termasuk MySQL 8.0.36.

Perbaikan ketersediaan:

- Memperbaiki masalah yang dapat menyebabkan instans DB pembaca dimulai ulang saat membebaskan memori yang digunakan untuk aplikasi log.

- Memperbaiki masalah dalam menghitung metrik internal untuk indeks pencarian teks lengkap (FTS) yang dapat menyebabkan restart basis data.
- Memperbaiki masalah yang dapat menonaktifkan logging biner saat terjadi kesalahan saat melakukan transaksi besar.

Integrasi perbaikan bug MySQL Community Edition

Rilis ini mencakup semua perbaikan bug komunitas hingga dan termasuk 8.0.36. Lihat informasi yang lebih lengkap di Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 3.x.

Pembaruan mesin database Aurora MySQL 2024-06-04 (versi 3.07.0) (Usang)

Versi: 3.07.0

Aurora MySQL 3.07.0 umumnya tersedia. Versi Aurora MySQL 3.07 kompatibel dengan MySQL 8.0.36. Untuk informasi selengkapnya tentang perubahan komunitas yang telah terjadi, lihat Catatan Rilis MySQL 8.0.

Untuk detail fitur-fitur baru di Aurora MySQL versi 3, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0. Untuk perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2, lihat Membandingkan Aurora MySQL versi 2 dan Aurora MySQL versi 3. Untuk perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition, lihat Membandingkan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition di Panduan Pengguna Amazon Aurora.

Rilis Aurora MySQL yang didukung saat ini adalah 2.07.9, 2.07.10, 2.11.*, 2.12.*, 3.03.*, 3.04.*, 3.05.*, 3.06.*, dan 3.07.*.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Memperbaiki masalah keamanan dan CVEs:

- Dukungan yang diaktifkan untuk kriptografi yang divalidasi FIPS, implementasi yang dimiliki sepenuhnya. AWS Untuk informasi lebih lanjut, lihat AWS-LC sekarang FIPS 140-3 bersertifikat di AWS Blog Keamanan.

Rilis ini mencakup semua perbaikan CVE komunitas hingga dan termasuk MySQL 8.0.36. Perbaikan CVE berikut disertakan:

- CVE-2020-11104
- CVE-2020-11105
- CVE-2023-38545
- CVE-2023-38546
- CVE-2023-39975

Perbaikan ketersediaan:

- Memperbaiki masalah yang dapat menyebabkan instans DB pembaca dimulai ulang saat membaca tabel yang sedang diubah atau dijatuhkan pada instance DB penulis.
- Memperbaiki masalah yang dapat menyebabkan instance DB penulis MySQL Aurora memulai ulang saat sesi penerusan tulis ditutup saat menjalankan kueri yang diteruskan.
- Memperbaiki masalah yang menyebabkan instans DB dimulai ulang saat menangani set GTID besar pada instance berkemampuan log biner.
- Memperbaiki masalah saat memproses `INSERT` kueri pada tabel partisi InnoDB yang dapat menyebabkan penurunan memori bebas secara bertahap dalam instance.
- Memperbaiki masalah yang, dalam kondisi langka, dapat menyebabkan instans DB pembaca dimulai ulang.
- Memperbaiki masalah yang dapat menyebabkan instance database dimulai ulang saat menjalankan pernyataan SHOW STATUS dan PURGE BINARY LOGS secara bersamaan. `PURGE BINARY LOGS`adalah pernyataan terkelola yang dijalankan untuk menghormati periode retensi binlog yang dikonfigurasi pengguna.
- Memperbaiki masalah yang dapat menyebabkan server tiba-tiba menutup setelah menjalankan pernyataan Data Manipulation Language (DML/Data Manipulation Language) pada tabel yang kolom nonvirtualnya disusun ulang dengan pernyataan atau. `MODIFY COLUMN` `CHANGE COLUMN`
- Memperbaiki masalah yang, selama restart instance database, dapat menyebabkan restart tambahan.
- Memperbaiki masalah yang dapat menyebabkan instans DB pembaca yang menggunakan penerusan tulis untuk memulai ulang saat pernyataan komit implisit yang diteruskan mengalami kesalahan.

- Memperbaiki masalah yang, dalam kondisi langka, dapat menyebabkan instance pembaca dimulai ulang saat melakukan `SELECT` kueri pada tabel dengan kendala kunci asing.
- Memperbaiki masalah saat instans DB yang menggunakan volume cluster Aurora multi-TB dapat mengalami peningkatan waktu henti selama restart karena kegagalan validasi kumpulan buffer InnoDB.
- Memperbaiki masalah yang dapat menyebabkan database dimulai ulang ketika kendala cascading `UPDATE` atau kunci `DELETE` asing didefinisikan pada tabel di mana kolom virtual terlibat baik sebagai kolom dalam batasan kunci asing, atau sebagai anggota tabel referensi.
- Memperbaiki masalah yang dapat mengganggu pemulihan database selama startup jika restart terjadi saat menjalankan operasi penyisipan berat yang melibatkan `AUTO_INCREMENT` kolom.
- Memperbaiki masalah Aurora Serverless v2 yang dapat menyebabkan restart basis data saat meningkatkan skala.

Perbaikan umum:

- Mengurangi I/O penggunaan dan meningkatkan kinerja untuk subset kueri pemindaian rentang kunci primer yang menggunakan query paralel.
- Aurora MySQL versi 3.06.0 menambahkan dukungan untuk integrasi Amazon Bedrock. Sebagai bagian dari ini, kata kunci baru yang dicadangkan (`acceptaws_bedrock_invoke_model`,`aws_sagemaker_invoke_endpoint`,`content_type`,, dan`timeout_ms`) ditambahkan. Dalam Aurora MySQL versi 3.07.0, kata kunci ini telah diubah menjadi kata kunci nonreserved, yang diizinkan sebagai pengidentifikasi tanpa kutipan. Untuk informasi selengkapnya tentang cara MySQL menangani kata kunci reserved dan nonreserved, lihat Kata kunci dan kata cadangan dalam dokumentasi MySQL.
- Memperbaiki masalah yang tidak mengembalikan pesan kesalahan dengan jelas ke klien saat menjalankan layanan Amazon Bedrock dari cluster DB MySQL Aurora di Wilayah AWS tempat Amazon Bedrock belum tersedia.
- Memperbaiki masalah yang dapat menyebabkan konsumsi memori berlebihan saat menanyakan `BLOB` kolom menggunakan kueri paralel Aurora.
- Ditambahkan dukungan untuk `connection_memory_limit` dan `connection_memory_chunk_size` parameter yang akan ditetapkan pada tingkat sesi untuk berperilaku sama seperti di MySQL Community Edition. `connection_memory_limit`Ini digunakan untuk mengatur jumlah maksimum memori yang dapat digunakan oleh koneksi pengguna tunggal. `connection_memory_chunk_size`Parameter dapat digunakan untuk mengatur ukuran chunking untuk pembaruan ke penghitung penggunaan memori global.

- Memperbaiki masalah saat pengguna tidak dapat mengganggu kueri apa pun atau mengatur batas waktu sesi untuk `performance_schema` kueri.
- Memperbaiki masalah di mana replikasi log biner (binlog) dikonfigurasi untuk menggunakan sertifikat SSL khusus (mysql.rds_import_binlog_ssl_material) dapat gagal saat instance replikasi sedang menjalani penggantian host.
- Menambahkan variabel status `Aurora_fts_cache_memory_used` global untuk melacak penggunaan memori untuk sistem pencarian teks lengkap di semua tabel. Untuk informasi selengkapnya, lihat Variabel status global Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Memperbaiki masalah saat klaster Amazon Redshift yang dikonfigurasi sebagai tujuan nol-ETL mungkin mengalami peningkatan sementara saat cluster DB MySQL Amazon IntegrationLagAurora dikonfigurasi sebagai replika log biner, dengan integrasi Binlog yang Ditingkatkan dan Nol-ETL diaktifkan.
- Memperbaiki masalah yang terkait dengan manajemen file log audit yang dapat menyebabkan file log tidak dapat diakses untuk diunduh atau dirotasi, dan dalam beberapa kasus meningkatkan penggunaan CPU.
- Pemulihan `AUTO_INCREMENT` kunci yang dioptimalkan untuk mengurangi waktu penyelesaian untuk memulihkan snapshot, melakukan point-in-time pemulihan, dan mengkloning cluster DB dengan sejumlah besar tabel dalam database.
- Memperbaiki masalah saat acara wait/io/redo_log_flush tidak ditampilkan dalam tabel ringkasan acara tunggu Skema Kinerja.
- Memperbaiki masalah yang dapat menyebabkan kesalahan kunci duplikat untuk `AUTO_INCREMENT` kolom yang menggunakan indeks turun setelah pemulihan snapshot, backtrack, atau operasi kloning database.
- Memperbaiki masalah yang dapat menyebabkan instans DB penulis dimulai ulang saat instance DB pembaca menggunakan penerusan tulis menjalankan pernyataan Data Manipulation Language (DML/Data Manipulation Language) yang berisi nilai stempel waktu dan parameter `time_zone` database disetel ke. `UTC`
- Memperbaiki masalah di mana `SELECT` kueri pada instance pembaca Aurora mungkin gagal dengan tabel kesalahan tidak ada ketika tabel memiliki setidaknya satu indeks pencarian teks lengkap (FTS) dan `TRUNCATE` pernyataan sedang dijalankan pada instance DB penulis Aurora.
- Memperbaiki masalah yang, dalam kasus yang jarang terjadi, menyebabkan patching zero-downtime (ZDP) gagal.

- Memperbaiki masalah yang dapat menyebabkan set hasil yang tidak lengkap saat menjalankan kueri yang melibatkan `LEFT JOIN` atau `RIGHT JOIN` operasi menggunakan algoritma gabungan hash dengan query paralel.

Peningkatan dan migrasi:

- Memperbaiki masalah yang dapat menyebabkan kegagalan upgrade dari Aurora MySQL versi 2 ke Aurora MySQL versi 3 ketika ada kolom yang ditentukan pengguna hadir dalam skema tabel. `FTS_DOC_ID`
- Memperbaiki masalah yang dapat menyebabkan kegagalan upgrade dari Aurora MySQL versi 2 ke Aurora MySQL versi 3 karena masalah sinkronisasi saat memproses ruang meja InnoDB.
- Memperbaiki masalah yang dapat menyebabkan peningkatan versi utama ke Aurora MySQL versi 3 gagal karena adanya entri yatim piatu untuk ruang meja yang sudah dihapus di tabel sistem InnoDB di Aurora MySQL versi 2.
- Memperbaiki masalah saat nilai SERVER_ID tidak diperbarui setelah peralihan Penerapan Amazon Blue/Green RDS. Hal ini menyebabkan masalah di mana driver pintar seperti Amazon Web Services (AWS) JDBC Driver tidak dapat menemukan topologi cluster DB setelah peralihan. blue/green Dengan perbaikan ini, cluster Aurora DB diganti namanya sebagai bagian dari Blue/Green Penyebaran RDS, yang berjalan pada Aurora MySQL versi 3.07 dan lebih tinggi, akan memiliki nilai diperbarui sebagai bagian dari peralihan. `SERVER_ID` Untuk versi sebelumnya, instans DB di cluster biru dan hijau dapat di-boot ulang untuk memperbarui nilai. `SERVER_ID`

Integrasi perbaikan bug MySQL Community Edition

Rilis ini mencakup semua perbaikan bug komunitas hingga dan termasuk 8.0.36, selain yang berikut ini. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 3.x.

- Memperbaiki masalah di mana nilai baris cache dapat dihitung secara tidak benar, menyebabkan kegagalan selama restart database pada instance berbasis Graviton. (Perbaikan Bug Komunitas #35479763)
- Memperbaiki masalah di mana beberapa contoh subkueri dalam rutinitas tersimpan tidak ditangani dengan benar. (Perbaikan Bug Komunitas #35377192)
- Memperbaiki masalah yang dapat menyebabkan penggunaan CPU lebih tinggi karena rotasi sertifikat TLS latar belakang (Perbaikan Bug Komunitas #34284186).

- Memperbaiki masalah di mana InnoDB mengizinkan penambahan `INSTANT` kolom ke tabel dalam skema sistem MySQL di versi MySQL Aurora MySQL lebih rendah dari 3.05, yang dapat menyebabkan server tiba-tiba menutup (instance database restart) setelah upgrade ke Aurora MySQL versi 3.05.0. (Perbaikan Bug Komunitas #35625510).

Pembaruan mesin database Aurora MySQL 2024-06-26 (versi 3.06.1) (Usang)

Versi: 3.06.1

Aurora MySQL 3.06.1 umumnya tersedia. Versi Aurora MySQL 3.06 kompatibel dengan MySQL 8.0.34. Untuk informasi selengkapnya tentang perubahan komunitas yang telah terjadi, lihat Catatan Rilis MySQL 8.0.

Untuk detail fitur-fitur baru di Aurora MySQL versi 3, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0. Untuk perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2, lihat Membandingkan Aurora MySQL versi 2 dan Aurora MySQL versi 3. Untuk perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition, lihat Membandingkan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition di Panduan Pengguna Amazon Aurora.

Rilis Aurora MySQL yang didukung saat ini adalah 2.07.9, 2.07.10, 2.11.*, 2.12.*, 3.03.*, 3.04.*, 3.05.*, 3.06.*, dan 3.07.*.

Anda dapat melakukan pemutakhiran di tempat, memulihkan snapshot, atau memulai pemutakhiran terkelola blue/green menggunakan Amazon RDS Blue/Green Deployment dari klaster Aurora MySQL versi 2 yang saat ini didukung ke dalam klaster Aurora MySQL versi 3.06.1.

Untuk informasi tentang merencanakan peningkatan ke Aurora MySQL versi 3, lihat Merencanakan peningkatan versi utama untuk cluster DB MySQL Aurora. Untuk informasi umum tentang peningkatan Aurora MySQL, lihat Meningkatkan klaster DB Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Untuk informasi pemecahan masalah, lihat Pemecahan Masalah untuk peningkatan Aurora MySQL di tempat di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Memperbaiki masalah keamanan dan CVEs:

Rilis ini mencakup semua perbaikan CVE komunitas hingga dan termasuk MySQL 8.0.34. Perbaikan CVE berikut disertakan:

- CVE-2023-44487
- CVE-2024-0853

Perbaikan ketersediaan:

- Memperbaiki masalah yang menyebabkan instance Aurora MySQL DB dimulai ulang saat menjalankan kueri paralel.
- Memperbaiki masalah yang dapat menyebabkan instans DB pembaca dimulai ulang saat membaca tabel yang sedang diubah atau dijatuhkan pada instance DB penulis.
- Memperbaiki masalah yang menyebabkan pelanggaran akses memori yang menyebabkan pelepasan objek mutex yang tidak lagi dimiliki oleh utas.
- Memperbaiki masalah yang dapat menyebabkan instance DB penulis MySQL Aurora memulai ulang saat sesi penerusan tulis ditutup saat menjalankan kueri yang diteruskan.
- Memperbaiki masalah yang menyebabkan instans DB dimulai ulang saat menangani set GTID besar pada instance berkemampuan log biner.
- Memperbaiki masalah yang, dalam kondisi langka, dapat menyebabkan instance pembaca dimulai ulang saat melakukan `SELECT` kueri pada tabel dengan kendala kunci asing.
- Memperbaiki masalah yang menyebabkan instans DB dimulai ulang saat mencoba memulihkan kamus data InnoDB selama pemulihan database.
- Memperbaiki masalah Aurora Serverless v2 yang dapat menyebabkan restart database saat meningkatkan skala.

Perbaikan umum:

- Memperbaiki masalah dalam kode penerbitan metrik di mana memori dapat digunakan setelah dibebaskan.
- Memperbaiki masalah yang menyebabkan restart mesin DB berulang karena objek tablespace undo yang tidak ada.

- Memperbaiki masalah dengan pemotongan otomatis undo tablespaces ketika mereka lebih besar dari ambang innodb_max_undo_log_size dalam skenario peningkatan.
- Memperbaiki masalah yang memberikan nilai yang salah untuk variabel `threads_running` status saat menggunakan Aurora Global Database.
- Memperbaiki masalah di mana replika baca log biner Aurora MySQL (binlog) dengan optimasi indeks sekunder paralel diaktifkan akan mengalami restart saat menerapkan perubahan replikasi pada tabel dengan kunci asing.
- Aurora MySQL versi 3.06.0 menambahkan dukungan untuk integrasi Amazon Bedrock. Sebagai bagian dari ini, kata kunci baru yang dicadangkan (`acceptaws_bedrock_invoke_model`,`aws_sagemaker_invoke_endpoint`,`content_type`,, dan`timeout_ms`) ditambahkan. Dalam Aurora MySQL versi 3.06.1, kata kunci ini telah diubah menjadi kata kunci nonreserved, yang diizinkan sebagai pengidentifikasi tanpa kutipan. Untuk informasi selengkapnya tentang cara MySQL menangani kata kunci reserved dan nonreserved, lihat Kata kunci dan kata cadangan dalam dokumentasi MySQL.
- Memperbaiki masalah yang tidak mengembalikan pesan kesalahan dengan jelas ke klien saat menjalankan layanan Amazon Bedrock dari cluster DB MySQL Aurora di Wilayah AWS tempat Amazon Bedrock belum tersedia.
- Memperbaiki masalah yang menyebabkan instans DB dimulai ulang karena informasi pemegang kunci yang tidak akurat `rw_lock` saat menggunakan pembacaan paralel.
- Memperbaiki masalah yang dapat menyebabkan instans DB dimulai ulang saat `SHOW VOLUME STATUS` dijalankan.
- Memperbaiki masalah manajemen memori yang menyebabkan penurunan memori yang dapat dibebaskan dari waktu ke waktu saat menjalankan `SELECT ... INTO OUTFILE ...` kueri.
- Menambahkan dukungan untuk `connection_memory_chunk_size` parameter `connection_memory_limit` dan yang akan diatur pada tingkat sesi agar berperilaku mirip dengan fungsionalitas yang sesuai di MySQL Community Edition. `connection_memory_limit`Parameter menetapkan jumlah maksimum memori yang dapat digunakan oleh koneksi pengguna tunggal. `connection_memory_chunk_size`Parameter menetapkan ukuran chunking untuk pembaruan ke penghitung penggunaan memori global.
- Memperbaiki masalah yang dapat menyebabkan instans DB dimulai ulang saat penyimpanan lokal pada instans DB mencapai kapasitas penuh.
- Memperbaiki masalah saat Skema Kinerja tidak diaktifkan saat manajemen otomatis Performance Insights diaktifkan untuk instans DB db.t4g.medium dan db.t4g.large.

- Memperbaiki masalah yang dapat menyebabkan instans DB penulis dimulai ulang saat instance DB pembaca menggunakan penerusan tulis menjalankan pernyataan Data Manipulation Language (DML/Data Manipulation Language) yang berisi nilai stempel waktu dan parameter `time_zone` database disetel ke. `UTC`
- Memperbaiki masalah selama zero-downtime patching (ZDP) yang mencegah instans DB menutup koneksi klien setelah mencapai nilai minimum yang dikonfigurasi pelanggan dari salah satu atau. `wait_timeout` `interactive_timeout`

Peningkatan dan migrasi:

- Memperbaiki masalah yang menyebabkan peningkatan atau migrasi gagal saat target versi mesin Aurora MySQL DB adalah 3.04.0 atau lebih tinggi. Hal ini terjadi ketika parameter cluster `lower_case_table_names` DB diatur ke`1`, dan MySQL database collation tidak kompatibel dengan nama tabel huruf kecil.

Integrasi perbaikan bug MySQL Community Edition

Rilis ini mencakup semua perbaikan bug komunitas hingga dan termasuk 8.0.34. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 3.x.

Pembaruan mesin database Aurora MySQL 2024-03-07 (versi 3.06.0) (Usang)

Versi: 3.06.0

Aurora MySQL 3.06.0 umumnya tersedia. Versi Aurora MySQL 3.06 kompatibel dengan MySQL 8.0.34. Untuk informasi selengkapnya tentang perubahan komunitas yang telah terjadi, lihat Catatan Rilis MySQL 8.0.

Untuk detail fitur-fitur baru di Aurora MySQL versi 3, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0. Untuk perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2, lihat Membandingkan Aurora MySQL versi 2 dan Aurora MySQL versi 3. Untuk perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition, lihat Membandingkan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition di Panduan Pengguna Amazon Aurora.

Rilis Aurora MySQL yang didukung saat ini adalah 2.07.9, 2.07.10, 2.11.*, 2.12.*, 3.03.*, 3.04.*, 3.05.*, dan 3.06.*.

Anda dapat melakukan upgrade di tempat, memulihkan snapshot, atau memulai upgrade terkelola blue/green menggunakan Amazon RDS Blue/Green Deployment dari klaster Aurora MySQL versi 2 yang didukung saat ini ke dalam klaster Aurora MySQL versi 3.06.0.

Untuk informasi tentang merencanakan peningkatan ke Aurora MySQL versi 3, lihat Merencanakan peningkatan versi utama untuk cluster DB MySQL Aurora. Untuk informasi umum tentang peningkatan Aurora MySQL, lihat Meningkatkan klaster DB Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Untuk informasi pemecahan masalah, lihat Pemecahan Masalah untuk peningkatan Aurora MySQL di tempat di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Fitur baru

- Aurora MySQL versi 3.06.0 mendukung integrasi Amazon Bedrock dan memperkenalkan kata kunci baru yang dicadangkan,,,, dan. `accept` `aws_bedrock_invoke_model` `aws_sagemaker_invoke_endpoint` `content_type` `timeout_ms` Periksa definisi objek untuk penggunaan kata kunci cadangan baru sebelum meningkatkan ke versi 3.06.0. Untuk mengurangi konflik dengan kata kunci baru yang dicadangkan, kutip kata kunci yang dicadangkan yang digunakan dalam definisi objek. Untuk informasi selengkapnya tentang integrasi Amazon Bedrock dan penanganan kata kunci yang dicadangkan, lihat Apa itu Amazon Bedrock? di Panduan Pengguna Amazon Aurora. Untuk informasi tambahan, lihat Kata Kunci dan Kata Cadangan, Tabel KATA KUNCI INFORMATION_SCHEMA, dan Nama Objek Skema dalam dokumentasi MySQL.
- Peningkatan kinerja untuk replika log biner saat mereplikasi transaksi untuk tabel besar dengan lebih dari satu indeks sekunder. Fitur ini memperkenalkan kumpulan utas untuk menerapkan perubahan indeks sekunder secara paralel pada replika binlog. Fitur ini dikendalikan oleh parameter cluster `aurora_binlog_replication_sec_index_parallel_workers` DB, yang mengontrol jumlah total thread paralel yang tersedia untuk menerapkan perubahan indeks sekunder. Untuk informasi selengkapnya, lihat Mengoptimalkan replikasi log biner di Panduan Pengguna Amazon Aurora.

- Menambahkan prosedur tersimpan baru `mysql.rds_set_read_only` yang memungkinkan mengubah nilai variabel sistem global pada instance database `read_only` di cluster Aurora MySQL Anda. Untuk informasi selengkapnya, lihat Mereplikasi di Panduan Pengguna Amazon Aurora.
- Menambahkan prosedur tersimpan baru `mysql.rds_set_binlog_source_ssl` yang memungkinkan pengaturan enkripsi pada replika log biner dengan menentukan nilai untuk. `SOURCE_SSL` Untuk informasi selengkapnya, lihat Mereplikasi di Panduan Pengguna Amazon Aurora.
- Amazon Aurora Machine Learning adalah integrasi yang dioptimalkan antara database Aurora MySQL dan layanan machine learning (ML). AWS Amazon Bedrock sekarang didukung, memungkinkan Anda untuk memanggil model pembelajaran mesin di Amazon Bedrock langsung dari cluster Aurora MySQL DB Anda menggunakan SQL. Untuk informasi selengkapnya tentang penggunaan Amazon Bedrock dengan cluster DB MySQL Aurora Anda, lihat Menggunakan pembelajaran mesin Amazon Aurora dengan Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Aurora MySQL versi 3.06 menambahkan dukungan untuk pemotongan tablespace undo otomatis. Pengoptimalan ini memungkinkan Anda untuk merebut kembali ruang yang tidak digunakan di undo tablespaces setelah log undo telah dibersihkan.

Perbaikan

Memperbaiki masalah keamanan dan CVEs:

Perbaikan CVE berikut disertakan dalam rilis ini:

- CVE-2020-11104
- CVE-2020-11105
- CVE-2023-38545
- CVE-2023-38546
- CVE-2023-39975

Perbaikan ketersediaan:

- Memperbaiki masalah di mana instans DB replika baca tidak dapat diluncurkan dengan sukses ketika ada beban kerja yang tinggi di instans DB penulis.

- Memperbaiki masalah di mana instance DB penulis MySQL Aurora dapat gagal karena cacat dalam komunikasi dengan penyimpanan Aurora. Cacat terjadi sebagai akibat dari gangguan dalam komunikasi antara instans DB dan penyimpanan yang mendasarinya setelah pembaruan perangkat lunak dari instance penyimpanan Aurora.
- Memperbaiki masalah saat memproses `INSERT` kueri pada tabel partisi InnoDB yang dapat menyebabkan penurunan memori bebas secara bertahap dalam instance.
- Memperbaiki masalah yang dapat menyebabkan instance Aurora MySQL DB dimulai ulang atau gagal karena penurunan memori yang dapat dibebaskan saat bergabung dengan hash digunakan saat menjalankan kueri.
- Memperbaiki masalah yang dapat menyebabkan instance database dimulai ulang saat menjalankan pernyataan SHOW STATUS dan PURGE BINARY LOGS secara bersamaan. `PURGE BINARY LOG`Sadalah pernyataan terkelola yang dijalankan untuk menghormati periode retensi binlog yang dikonfigurasi pengguna.
- Memperbaiki masalah yang dapat menyebabkan server tiba-tiba menutup setelah menjalankan pernyataan Data Manipulation Language (DML/Data Manipulation Language) pada tabel yang kolom nonvirtualnya disusun ulang dengan pernyataan atau. `MODIFY COLUMN` `CHANGE COLUMN`
- Memperbaiki masalah yang, selama restart instance database, dapat menyebabkan restart tambahan.
- Memperbaiki masalah yang dapat menyebabkan database dimulai ulang ketika kendala cascading `UPDATE` atau kunci `DELETE` asing didefinisikan pada tabel di mana kolom virtual terlibat baik sebagai kolom dalam batasan kunci asing, atau sebagai anggota tabel referensi.
- Di Aurora MySQL 2.10, kami menambahkan dukungan untuk me-reboot cluster Aurora DB dengan ketersediaan baca. Fitur ini memungkinkan instans DB pembaca untuk tetap online saat instance DB penulis di-boot ulang. Fitur ini sekarang didukung pada sekunder Wilayah AWS di database global Aurora MySQL, memastikan bahwa Anda masih dapat melayani permintaan baca selama restart instance penulis di cluster utama. Sebelumnya, ketika instance penulis dimulai ulang, semua instance pembaca di cluster sekunder Aurora MySQL juga dimulai ulang. Dengan rilis ini, instance pembaca klaster sekunder terus menyajikan permintaan baca selama instance penulis dimulai ulang, meningkatkan ketersediaan baca di cluster. Untuk informasi selengkapnya, lihat Mem-boot ulang klaster Aurora dengan ketersediaan baca.
- Memperbaiki masalah yang dapat mengganggu pemulihan database selama startup jika restart terjadi saat menjalankan operasi penyisipan berat yang melibatkan `AUTO_INCREMENT` kolom.

Perbaikan umum:

- Memperbaiki masalah yang dapat menyebabkan kueri paralel gagal karena masalah jaringan sementara saat membaca data dari volume cluster Aurora.
- Memperbaiki masalah saat pengguna tidak dapat mengganggu kueri apa pun atau mengatur batas waktu sesi untuk `performance_schema` kueri.
- Memperbaiki masalah di mana replikasi log biner (binlog) dikonfigurasi untuk menggunakan sertifikat SSL khusus (mysql.rds_import_binlog_ssl_material) dapat gagal saat instance replikasi sedang menjalani penggantian host.
- Instans DB kecil dengan memori kurang dari atau sama dengan 4 GiB sekarang menutup koneksi yang memakan memori teratas saat instans DB berada di bawah tekanan memori. Anda juga dapat menyetel kolam penyangga untuk mengurangi ukurannya. Untuk informasi selengkapnya, lihat masalah Amazon Aurora MySQL di Panduan Pengguna out-of-memory Amazon Aurora.
- Mengubah respons default untuk`aurora_oom_response`, pada semua kelas instans DB yang memiliki lebih dari 4 GiB memori, dari kosong menjadi. `print` Untuk informasi selengkapnya, lihat masalah Amazon Aurora MySQL di Panduan Pengguna out-of-memory Amazon Aurora.
- Memperbaiki masalah yang terkait dengan manajemen file log audit yang dapat menyebabkan file log tidak dapat diakses untuk diunduh atau dirotasi, dan dalam beberapa kasus meningkatkan penggunaan CPU.
- Pemulihan `AUTO_INCREMENT` kunci yang dioptimalkan untuk mengurangi waktu penyelesaian untuk memulihkan snapshot, melakukan point-in-time pemulihan, dan mengkloning cluster DB dengan sejumlah besar tabel dalam database.
- Memperbaiki masalah saat acara wait/io/redo_log_flush tidak ditampilkan dalam tabel ringkasan acara tunggu Skema Kinerja.
- Menambahkan `Aurora_lockmgr_buffer_pool_memory_used` metrik `Aurora_lockmgr_memory_used` dan untuk melacak penggunaan memori pengelola kunci. Untuk informasi selengkapnya, lihat Variabel status global Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Memperbaiki masalah saat instance replika baca kecil dapat mengalami peningkatan kelambatan replikasi setelah memutakhirkan dari versi Aurora MySQL yang lebih rendah dari 2.11.*.
- Memperbaiki masalah yang dapat menyebabkan kesalahan kunci duplikat untuk `AUTO_INCREMENT` kolom yang menggunakan indeks turun setelah pemulihan snapshot, backtrack, atau operasi kloning database.
- Memperbaiki masalah di mana `SELECT` kueri pada instance pembaca Aurora mungkin gagal dengan tabel kesalahan tidak ada ketika tabel memiliki setidaknya satu indeks pencarian teks lengkap (FTS) dan `TRUNCATE` pernyataan sedang dijalankan pada instance DB penulis Aurora.

- Memperbaiki masalah yang dapat menyebabkan set hasil yang tidak lengkap saat menjalankan kueri yang melibatkan `LEFT JOIN` atau `RIGHT JOIN` operasi menggunakan algoritma gabungan hash dengan query paralel.

Peningkatan dan migrasi:

- Memperbaiki masalah yang dapat menyebabkan upgrade versi mayor gagal jika ada `FTS_DOC_ID` kolom yang ditentukan pengguna hadir dalam skema tabel.
- Memperbaiki masalah yang dapat menyebabkan kegagalan pemutakhiran dari Aurora My SQLversion 2 ke Aurora MySQL versi 3 karena masalah sinkronisasi saat memproses ruang meja InnoDB.
- Memperbaiki masalah yang dapat menyebabkan peningkatan versi utama ke Aurora MySQL versi 3 gagal karena adanya entri yatim piatu untuk ruang meja yang sudah dihapus di tabel sistem InnoDB di Aurora MySQL versi 2.

Integrasi perbaikan bug MySQL Community Edition

Rilis ini mencakup semua perbaikan bug komunitas hingga dan termasuk 8.0.34, selain yang berikut ini. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 3.x.

- Memperbaiki masalah di mana nilai baris cache dapat dihitung secara tidak benar, menyebabkan kegagalan selama restart database pada instance berbasis Graviton. (Perbaikan Bug Komunitas #35479763)
- Memperbaiki masalah di mana beberapa contoh subkueri dalam rutinitas tersimpan tidak selalu ditangani dengan benar. (Perbaikan Bug Komunitas #35377192)
- Memperbaiki masalah yang dapat menyebabkan penggunaan CPU lebih tinggi karena rotasi sertifikat TLS latar belakang (Perbaikan Bug Komunitas #34284186).
- Memperbaiki masalah di mana InnoDB mengizinkan penambahan `INSTANT` kolom ke tabel dalam skema sistem MySQL di versi MySQL Aurora MySQL lebih rendah dari 3.05, yang dapat menyebabkan server tiba-tiba menutup (instance database restart) setelah upgrade ke Aurora MySQL versi 3.05.0. (Perbaikan Bug Komunitas #35625510).

Pembaruan mesin database Aurora MySQL 2024-01-31 (versi 3.05.2) (Usang)

Versi: 3.05.2

Aurora MySQL 3.05.2 umumnya tersedia. Aurora MySQL versi 3.05 kompatibel dengan MySQL 8.0.32. Untuk informasi selengkapnya tentang perubahan komunitas yang telah terjadi, lihat Catatan Rilis MySQL 8.0.

Untuk detail fitur-fitur baru di Aurora MySQL versi 3, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0. Untuk perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2, lihat Membandingkan Aurora MySQL versi 2 dan Aurora MySQL versi 3. Untuk perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition, lihat Membandingkan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition di Panduan Pengguna Amazon Aurora.

Rilis Aurora MySQL yang didukung saat ini adalah 2.07.9, 2.07.10, 2.11.*, 2.12.*, 3.03.*, 3.04.*, dan 3.05.*.

Anda dapat melakukan upgrade di tempat, memulihkan snapshot, atau memulai upgrade terkelola blue/green menggunakan Amazon RDS Blue/Green Deployment dari klaster Aurora MySQL versi 2 yang didukung saat ini ke dalam klaster Aurora MySQL versi 3.05.2.

Untuk informasi tentang merencanakan upgrade ke Aurora MySQL versi 3, lihat Perencanaan upgrade untuk Aurora MySQL versi 3. Untuk informasi umum tentang peningkatan Aurora MySQL, lihat Meningkatkan klaster DB Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Untuk informasi pemecahan masalah, lihat Memecahkan masalah pemutakhiran dengan Aurora MySQL versi 3 di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Memperbaiki masalah keamanan dan CVEs:

Perbaikan CVE berikut disertakan dalam rilis ini:

- CVE-2020-11104

- CVE-2020-11105
- CVE-2023-38545
- CVE-2023-39975

Perbaikan ketersediaan:

- Memperbaiki masalah saat memproses `INSERT` kueri pada tabel yang dipartisi InnoDB dapat menyebabkan penurunan memori bebas secara bertahap dalam instance.
- Memperbaiki masalah yang dapat menyebabkan instance database dimulai ulang saat menjalankan pernyataan SHOW STATUS dan PURGE BINARY LOGS secara bersamaan. `PURGE BINARY LOGS`Adalah pernyataan terkelola yang dijalankan untuk menghormati periode retensi binlog yang dikonfigurasi pengguna.
- Memperbaiki masalah yang dapat menyebabkan server tiba-tiba menutup setelah menjalankan pernyataan Data Manipulation Language (DML/Data Manipulation Language) pada tabel yang kolom nonvirtualnya disusun ulang dengan pernyataan atau. `MODIFY COLUMN` `CHANGE COLUMN`
- Memperbaiki masalah yang, selama restart instance database, dapat menyebabkan restart tambahan.

Perbaikan umum:

- Memperbaiki masalah saat pengguna tidak dapat mengganggu kueri apa pun atau mengatur batas waktu sesi untuk `performance_schema` kueri.
- Memperbaiki masalah di mana penyiapan replikasi log biner (binlog) menggunakan sertifikat SSL khusus (mysql.rds_import_binlog_ssl_material) dapat gagal saat instance replikasi sedang menjalani penggantian host.
- Memperbaiki masalah yang terkait dengan manajemen file log audit yang dapat menyebabkan file log tidak dapat diakses untuk diunduh atau dirotasi, dan dalam beberapa kasus meningkatkan penggunaan CPU.
- Dalam versi MySQL Aurora yang lebih rendah dari 3.05.2, pengguna tidak dapat mengambil output pada instance DB pembaca MySQL Aurora. `SHOW ENGINE INNODB STATUS` Hal ini disebabkan perilaku InnoDB default saat `innodb_read_only` diaktifkan.

  Di Aurora MySQL versi 3.05.2 dan yang lebih tinggi, ketika `SHOW ENGINE INNODB STATUS` dijalankan pada instance pembaca, output ditulis ke log kesalahan MySQL, memungkinkan pemecahan masalah yang lebih mudah.

Untuk informasi selengkapnya tentang bekerja dengan log kesalahan MySQL, lihat log kesalahan Aurora MySQL. Untuk informasi selengkapnya`SHOW ENGINE INNODB STATUS`, lihat pernyataan SHOW ENGINE dalam dokumentasi MySQL.

Peningkatan dan migrasi:

- Memperbaiki masalah yang dapat menyebabkan kegagalan upgrade dari Aurora MySQL versi 2 ke Aurora MySQL versi 3 ketika ada kolom yang ditentukan pengguna hadir dalam skema tabel. `FTS_DOC_ID`
- Memperbaiki masalah yang dapat menyebabkan kegagalan upgrade dari Aurora MySQL versi 2 ke Aurora MySQL versi 3 karena masalah sinkronisasi saat memproses ruang meja InnoDB.
- Memperbaiki masalah yang dapat menyebabkan peningkatan versi utama ke Aurora MySQL versi 3 gagal karena adanya entri yatim piatu untuk ruang meja yang sudah dihapus di tabel sistem InnoDB di Aurora MySQL versi 2.

Integrasi perbaikan bug MySQL Community Edition

Rilis ini mencakup semua perbaikan bug komunitas hingga dan termasuk 8.0.32, selain yang berikut ini. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 3.x.

- Memperbaiki masalah saat `records_in_range` melakukan pembacaan disk dalam jumlah berlebihan untuk `INSERT` operasi, yang menyebabkan penurunan kinerja secara bertahap. (Perbaikan Bug Komunitas #34976138)

Pembaruan mesin database Aurora MySQL 2023-11-21 (versi 3.05.1) (Usang)

Versi: 3.05.1

Aurora MySQL 3.05.1 tersedia secara umum. Aurora MySQL versi 3.05 kompatibel dengan MySQL 8.0.32. Untuk informasi selengkapnya, lihat Catatan rilis MySQL 8.0.

Rilis Aurora MySQL yang saat ini didukung adalah 2.07.*, 2.11.*, 2.12.*, 3.01.*, 3.02.*, 3.03.*, 3.04.*, dan 3.05.*.

Anda dapat meningkatkan klaster basis data Aurora MySQL 3.* yang ada ke Aurora MySQL 3.05.1. Anda juga dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 3.05.1.

Jika Anda meningkatkan basis data global Aurora MySQL ke versi 3.05.*, Anda harus meningkatkan klaster DB primer dan sekunder Anda ke versi yang sama persis, termasuk tingkat patch-nya. Untuk informasi selengkapnya tentang peningkatan versi kecil basis data global Aurora, lihat Peningkatan versi kecil.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Memperbaiki masalah keamanan dan CVEs tercantum di bawah ini:

Rilis ini mencakup semua CVEs perbaikan komunitas hingga dan termasuk MySQL 8.0.32.

- CVE-2023-38545

Integrasi perbaikan bug MySQL Community Edition

Rilis ini meliputi semua perbaikan bug komunitas hingga dan termasuk 8.0.32, selain hal-hal di bawah ini. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 3.x.

- Memperbaiki masalah di InnoDB ketika, jika tabel MySQL dalam skema sistem memiliki kolom yang ditambahkan `INSTANT ADD` antara Aurora MySQL versi 3.01 melalui Aurora MySQL versi 3.04, dan setelah Aurora MySQL ditingkatkan ke versi 3.05.0, pada tabel ini akan mengakibatkan server tiba-tiba ditutup. DMLs (Perbaikan Bug Komunitas #35625510)

Pembaruan mesin database Aurora MySQL 2023-10-30 (versi 3.05.0.1) (Usang)

Versi: 3.05.0.1

Aurora MySQL 3.05.0.1 umumnya tersedia di wilayah berikut: AS Timur (Virginia N.), AS Timur (Ohio), AS Barat (California N.), AS Barat (Oregon), (AS-Timur), dan (AS-Barat). AWS GovCloud

AWS GovCloud Ini adalah rilis awal khusus perbaikan keamanan. Semua perbaikan ini akan di-deploy secara lebih luas di semua Wilayah dengan rilis patch berikutnya, 3.05.1. Aurora MySQL versi 3.05 kompatibel dengan MySQL 8.0.32. Untuk informasi selengkapnya tentang perubahan komunitas yang telah terjadi, lihat Catatan Rilis MySQL 8.0.

Rilis Aurora MySQL yang saat ini didukung adalah 2.07.*, 2.11.*, 2.12.*, 3.01.*, 3.02.*, 3.03.*, 3.04.*, dan 3.05.*.

Anda dapat meningkatkan klaster basis data Aurora MySQL 3.* yang ada ke Aurora MySQL 3.05.0.1. Anda juga dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 3.05.0.1.

Jika Anda meningkatkan basis data global Aurora MySQL ke versi 3.05.*, Anda harus meningkatkan klaster DB primer dan sekunder Anda ke versi yang sama persis, termasuk tingkat patch-nya. Untuk informasi selengkapnya tentang peningkatan versi kecil basis data global Aurora, lihat Peningkatan versi kecil.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Memperbaiki masalah keamanan dan CVEs tercantum di bawah ini:

Rilis ini mencakup semua CVEs perbaikan komunitas hingga dan termasuk MySQL 8.0.32.

- CVE-2023-38545

Pembaruan mesin database Aurora MySQL 2023-10-25 (versi 3.05.0) (Usang)

Versi: 3.05.0

Aurora MySQL 3.05.0 tersedia secara umum. Aurora MySQL versi 3.05 kompatibel dengan MySQL 8.0.32. Untuk informasi selengkapnya tentang perubahan komunitas yang telah terjadi, lihat Catatan Rilis MySQL 8.0.

Untuk detail fitur-fitur baru di Aurora MySQL versi 3, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0. Untuk perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2, lihat

Membandingkan Aurora MySQL versi 2 dan Aurora MySQL versi 3. Untuk perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition, lihat Membandingkan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition.

Rilis Aurora MySQL yang didukung saat ini adalah 2.07.9, 2.07.10, 2.11.*, 2.12.*, 3.03.*, 3.04.*, dan 3.05.*.

Anda dapat melakukan upgrade di tempat, memulihkan snapshot, atau memulai upgrade terkelola blue/green menggunakan Amazon RDS Blue/Green Deployment dari klaster Aurora MySQL versi 2 yang didukung saat ini ke dalam klaster Aurora MySQL versi 3.05.0.

Untuk informasi tentang merencanakan peningkatan ke Aurora MySQL versi 3, lihat Perencanaan peningkatan untuk Aurora MySQL versi 3 di Panduan Pengguna Amazon Aurora. Untuk informasi umum tentang peningkatan Aurora MySQL, lihat Meningkatkan klaster DB Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Untuk informasi pemecahan masalah, lihat Memecahkan masalah peningkatan dengan Aurora MySQL versi 3.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Fitur baru:

- Menambahkan dukungan untuk menyimpan data dari klaster basis data Aurora MySQL ke dalam file teks di dalam bucket Amazon S3 yang dienkripsi dengan kunci KMS (SSE-KMS). Untuk informasi selengkapnya, lihat Menyimpan data dari klaster DB Amazon Aurora MySQL ke dalam file teks di bucket Amazon S3.
- Memperkenalkan variabel status global baru `aurora_tmz_version` untuk menunjukkan versi informasi zona waktu (TZ) saat ini yang digunakan oleh mesin. Nilai mengikuti versi database zona waktu IANA dan diformat sebagai "YYYYsuffix“, misalnya, 2022a dan 2023c. Untuk informasi selengkapnya, lihat Variabel status global Aurora MySQL.

Memperbaiki masalah keamanan dan CVEs tercantum di bawah ini:

Perbaikan dan penyempurnaan lain untuk penanganan fine-tune di lingkungan terkelola. Di bawah ini adalah beberapa perbaikan CVE tambahan:

- CVE-2022-37434

Perbaikan ketersediaan:

- Memperbaiki masalah di mana instans basis data MySQL Aurora yang menggunakan kueri paralel mungkin mengalami mulai ulang basis data saat menjalankan kueri paralel dalam jumlah besar secara bersamaan.
- Memperbaiki masalah dengan perselisihan kunci yang disebabkan oleh thread pencatatan log audit yang akhirnya mengarah ke batas waktu aplikasi klien dan penggunaan CPU yang tinggi.
- Memperbaiki masalah yang dapat menyebabkan set GTID yang dieksekusi dipulihkan secara tidak benar pada klaster replika log biner (binlog) dengan binlog yang ditingkatkan aktif ketika sumber binlog mana pun memiliki `gtid_mode` yang diatur ke `ON` atau `ON_PERMISSIVE`. Masalah ini dapat menyebabkan instans penulis klaster replika memulai ulang waktu tambahan selama pemulihan, atau menyebabkan hasil yang salah saat mengkueri set GTID yang dieksekusi.
- Memperbaiki masalah manajemen memori yang dapat menyebabkan instans basis data Aurora MySQL memulai ulang atau melakukan failover karena penurunan memori yang dapat dilepas saat log biner yang ditingkatkan diaktifkan.
- Memperbaiki masalah yang dapat menyebabkan instans basis data memulai ulang saat mencoba membaca halaman basis data milik tabel yang dihapus.
- Memperbaiki masalah yang dapat menyebabkan instans pembaca memulai ulang saat instans penulis meningkatkan volume basis data ke kelipatan 160GB.
- Memperbaiki masalah di mana instans basis data Aurora MySQL dengan fitur log biner yang ditingkatkan diaktifkan mungkin terhenti selama startup instans basis data saat proses pemulihan log biner sedang dijalankan.
- Memperbaiki masalah di mana instans basis data Aurora MySQL mungkin beberapa kali memulai ulang selama startup instans sementara segmen rollback besar diinisialisasi.
- Memperbaiki masalah selama zero downtime patching yang menyebabkan instans memulai ulang yang menyebabkan koneksi basis data ditutup secara tak terduga.
- Memperbaiki masalah yang dapat menyebabkan instans basis data memulai ulang karena deadlatch saat menjalankan pernyataan SHOW STATUS dan PURGE BINARY LOGS secara bersamaan. Log biner pembersihan adalah pernyataan terkelola yang dijalankan untuk menghormati periode retensi binlog yang dikonfigurasi pengguna.

- Memperbaiki masalah yang dapat menyebabkan klaster basis data menjadi tidak tersedia jika instans penulis memulai ulang saat basis data membuat atau menjatuhkan pemicu pada tabel sistem internal.
- Memperbaiki masalah yang dapat menyebabkan instans basis data memulai ulang karena menunggu semaphore yang lama saat menggunakan fitur binlog yang disempurnakan pada klaster dengan replika Aurora.
- Memperbaiki masalah yang dapat menyebabkan instans basis data memulai ulang saat menjalankan kueri yang mereferensikan fungsi agregat.
- Memperbaiki masalah yang, dalam kondisi yang sangat jarang terjadi, dapat menyebabkan instans basis data memulai ulang ketika Aurora Serverless v2 salah mencoba memperbarui cache tabel saat melakukan penskalaan.
- Memperbaiki masalah di mana metode akses pemindaian indeks yang tidak didukung dipertimbangkan untuk ekspresi tabel umum (CTE) saat mewujudkan tabel sementara menengah, yang dapat menyebabkan perilaku yang tidak diinginkan termasuk mulai ulang basis data atau kesalahan hasil kueri. Kami memperbaiki masalah ini dengan menghindari penggunaan metode akses pemindaian indeks yang tidak didukung pada tabel menggunakan mesin TempTable penyimpanan.

Perbaikan umum:

- Memperbaiki masalah yang dapat menyebabkan basis data menjadi tidak tersedia saat binlog yang disempurnakan diaktifkan pada klaster basis data Aurora Serverless v2 yang berjalan di Aurora MySQL 3.04.0.
- Menghapus metadata penyimpanan yang tidak digunakan sebelum menulis ke penyimpanan Aurora saat fitur binlog yang disempurnakan diaktifkan. Perbaikan ini menghindari skenario tertentu ketika basis data memulai ulang atau melakukan failover dapat terjadi karena peningkatan latensi penulisan karena peningkatan byte yang ditransmisikan melalui jaringan.
- Dengan penambahan tabel `malloc_stats` dan `malloc_stats_totals` di dalam `performance_schema`, tiga variabel sistem lanjutan ditambahkan untuk mengontrol perilaku Jemalloc, pengalokasi memori internal:
  - `aurora_jemalloc_background_thread`.
  - `aurora_jemalloc_dirty_decay_ms`.
  - `aurora_jemalloc_tcache_enabled`.

- Memperbaiki masalah di mana tabel skema kinerja khusus Aurora tidak dibuat saat peningkatan versi atau migrasi.
- Menambahkan variabel sistem baru, `aurora_use_vector_instructions`. Ketika parameter ini diaktifkan, Aurora MySQL menggunakan instruksi pemrosesan vektor yang dioptimalkan untuk meningkatkan kinerja pada beban kerja yang berat. I/O Pengaturan ini disetel `ON` secara default di Aurora MySQL 3.05 dan yang lebih tinggi. Untuk detail selengkapnya, lihat Parameter konfigurasi Aurora MySQL.
- Memperbaiki masalah yang dapat menyebabkan `NumBinaryLogFiles` metrik aktif menampilkan hasil yang salah saat binlog yang disempurnakan diaktifkan. CloudWatch
- Batas waktu permintaan untuk operasi Machine Learning Aurora MySQL ke Amazon Sagemaker telah ditingkatkan dari 3 menjadi 30 detik. Ini membantu menyelesaikan masalah di mana pelanggan mungkin melihat peningkatan jumlah percobaan ulang atau kegagalan untuk permintaan ke Amazon Sagemaker dari Machine Learning Aurora MySQL saat menggunakan ukuran batch yang lebih besar.
- Menambahkan dukungan untuk tabel `malloc_stats` dan `malloc_stats_totals` dalam basis data performance_schema.
- Memperbarui kata kunci `FROM` di dalam perintah `LOAD DATA FROM S3` menjadi opsional. Untuk informasi selengkapnya, lihat Memuat data ke dalam klaster DB Amazon Aurora MySQL dari file teks di bucket Amazon S3.
- Menambahkan dukungan untuk parameter `innodb_aurora_instant_alter_column_allowed`, yang mengontrol apakah algoritma `INSTANT` dapat digunakan untuk operasi `ALTER COLUMN`. Untuk informasi selengkapnya lihat Parameter tingkat klaster.
- Memperbaiki masalah yang dapat mencegah koneksi klien baru dibuat ke basis data saat penerusan penulisan diaktifkan.
- Memperbaiki masalah yang dapat menyebabkan modifikasi parameter basis data `table_open_cache` tidak berlaku sampai instans basis data dimulai ulang.
- Memperbaiki masalah yang dapat menyebabkan kesalahan kunci duplikat untuk kolom `AUTO_INCREMENT` yang menggunakan indeks menurun setelah pemulihan snapshot, backtrack, atau operasi klon basis data.
- Memperbaiki masalah yang melibatkan pemindaian indeks di mana hasil yang tidak akurat mungkin dikembalikan saat menjalankan kueri `SELECT` dengan klausa `GROUP BY` dan parameter `aurora_parallel_query` diatur ke `ON`.

- Memperbaiki masalah yang dapat menyebabkan penipisan memori yang tersedia saat menjalankan kueri terhadap tabel `INFORMATION_SCHEMA INNODB_TABLESPACES`.
- Memperbaiki masalah di mana instans pembaca tidak dapat membuka tabel, dengan ERROR 1146. Masalah ini terjadi ketika mengeksekusi jenis Data Definition Language (DDL) online tertentu sementara algoritma `INPLACE` sedang digunakan pada instans penulis.
- Memperbaiki masalah untuk mencegah instans memulai ulang selama penskalaan Aurora Serverless v2 saat proses pemantauan internal secara keliru mengirimkan permintaan penskalaan duplikat.
- Memperbaiki masalah yang dapat menyebabkan restart database saat konsumen log biner (binlog) yang terhubung menggunakan server replikasi binlog duplikat. IDs
- Memperkenalkan cache log relay dalam memori untuk replika log biner yang dikelola Aurora MySQL. Peningkatan ini dapat membantu mencapai peningkatan hingga 40% dalam throughput replikasi log biner. Peningkatan ini diaktifkan secara otomatis saat menggunakan replikasi log biner thread tunggal atau saat menggunakan replikasi multi-thread dengan pemosisian otomatis GTID diaktifkan.

Peningkatan dan migrasi:

- Peningkatan dari MySQL 5.7 ke MySQL 8.0 dengan tabel yang sangat banyak dalam satu basis data menyebabkan server mengonsumsi memori yang berlebihan. Ditemukan bahwa, selama proses pemeriksaan apakah tabel dapat ditingkatkan, kami mengambil semua objek `Table` kamus data di awal, dengan memproses masing-masing objek dan mengambil namanya, kemudian menjalankan Pemeriksaan Kompatibilitas Versi pada daftar. Mengambil semua objek di awal tidak perlu dilakukan dalam kasus ini, dan berkontribusi besar terhadap konsumsi memori. Untuk memperbaiki masalah ini, kami sekarang mengambil satu objek `Table` satu per satu dalam kasus tersebut, dengan melakukan pemeriksaan yang diperlukan, mengambil namanya, dan melepaskan objek tersebut, sebelum melanjutkan ke objek yang berikutnya. (Bug #34526001)
- Meningkatkan kinerja upgrade versi utama dari Aurora MySQL versi 2 ke versi 3 dengan menjalankan pemeriksaan tablespace secara paralel menggunakan semua v yang tersedia pada instance database. CPUs

Integrasi perbaikan bug MySQL Community Edition

Rilis ini meliputi semua perbaikan bug komunitas hingga dan termasuk 8.0.32, selain hal-hal di bawah ini. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 3.x.

- Memperbaiki masalah yang dapat menyebabkan penggunaan CPU yang lebih tinggi karena rotasi sertifikat TLS di latar belakang. (Perbaikan Bug Komunitas #34284186)

Pembaruan mesin database Aurora MySQL 2026-01-02 (versi 3.04.6, kompatibel dengan MySQL 8.0.28)

Versi: 3.04.6

Aurora MySQL 3.04.6 umumnya tersedia. Aurora MySQL versi 3.04 kompatibel dengan MySQL 8.0.28. Untuk informasi selengkapnya tentang perubahan komunitas yang telah terjadi, lihat Catatan Rilis MySQL 8.0.

Untuk detail fitur-fitur baru di Aurora MySQL versi 3, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0. Untuk perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2, lihat Perbandingan Aurora MySQL versi 2 dan Aurora MySQL versi 3. Untuk perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition, lihat Perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition di Panduan Pengguna Amazon Aurora.

Anda dapat melakukan pemutakhiran di tempat yang memanfaatkan zero-downtime-patch, memulihkan snapshot, atau memulai Blue/Green pemutakhiran terkelola blue/green menggunakan Penyebaran Amazon RDS dari klaster Aurora MySQL versi 2 yang saat ini didukung ke dalam klaster Aurora MySQL versi 3.04.6.

Untuk informasi tentang merencanakan peningkatan ke Aurora MySQL versi 3, lihat Merencanakan peningkatan versi utama untuk klaster MySQL Aurora. Untuk informasi umum tentang peningkatan Aurora MySQL, lihat Meningkatkan klaster DB Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Untuk informasi pemecahan masalah, lihat Pemecahan Masalah untuk peningkatan Aurora MySQL di tempat di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Peningkatan ketersediaan

- Memperbaiki masalah yang, dapat menyebabkan mesin restart saat berjalan `KILL <query-id>` setelah menjalankan kueri paralel yang `EXPLAIN FOR CONNECTION <query-id>` sedang berjalan.
- Memperbaiki masalah yang dapat menyebabkan instance penulis menjadi tidak tersedia jika penerusan tulis dinonaktifkan atau instance pembaca dimulai ulang saat menggunakan Global Write Forwarding atau Local Write Forwarding

Perbaikan umum

- Memperbaiki masalah yang menyebabkan instance pembaca tidak menghasilkan log kesalahan saat penerusan penulisan diaktifkan dan parameter “aurora_replica_read_consistency” dimodifikasi.
- Memperbaiki masalah yang dapat menyebabkan beberapa pernyataan SQL tidak masuk log dalam log audit.

Integrasi perbaikan bug MySQL Community Edition

Rilis ini mencakup semua perbaikan bug komunitas hingga dan termasuk 8.0.28. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 3.x.

Pembaruan mesin database Aurora MySQL 2025-05-05 (versi 3.04.4, kompatibel dengan MySQL 8.0.28)

Versi: 3.04.4

Aurora MySQL 3.04.4 umumnya tersedia. Aurora MySQL versi 3.04 kompatibel dengan MySQL 8.0.28. Untuk informasi selengkapnya tentang perubahan komunitas yang telah terjadi, lihat Catatan Rilis MySQL 8.0.

Untuk detail fitur-fitur baru di Aurora MySQL versi 3, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0. Untuk perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2, lihat Perbandingan Aurora MySQL versi 2 dan Aurora MySQL versi 3. Untuk perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition, lihat Perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Versi ini ditetapkan sebagai rilis dukungan jangka panjang (LTS). Untuk informasi selengkapnya, lihat Rilis dukungan jangka panjang (LTS) Aurora MySQL di Panduan Pengguna Amazon Aurora.
> Kami menyarankan agar Anda tidak menyetel `AutoMinorVersionUpgrade` parameter ke `true` (atau mengaktifkan peningkatan versi minor Otomatis di Konsol Manajemen AWS) untuk versi LTS. Melakukannya dapat menyebabkan cluster DB Anda ditingkatkan ke versi target berikutnya untuk kampanye Peningkatan Versi Kecil Otomatis, yang mungkin bukan versi LTS.

Anda dapat melakukan pemutakhiran di tempat yang memanfaatkan zero-downtime-patch, memulihkan snapshot, atau memulai Blue/Green pemutakhiran terkelola blue/green menggunakan Penyebaran Amazon RDS dari klaster Aurora MySQL versi 2 yang saat ini didukung ke dalam klaster Aurora MySQL versi 3.04.4.

Untuk informasi tentang merencanakan peningkatan ke Aurora MySQL versi 3, lihat Merencanakan peningkatan versi utama untuk klaster MySQL Aurora. Untuk informasi umum tentang peningkatan Aurora MySQL, lihat Meningkatkan klaster DB Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Untuk informasi pemecahan masalah, lihat Pemecahan Masalah untuk peningkatan Aurora MySQL di tempat di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan keamanan

Kritis CVEs:

- CVE-2024-11053
- CVE-2024-37371

Peningkatan ketersediaan

- Memperbaiki masalah pada replika di mana gangguan jaringan mungkin tidak membangun kembali koneksi dengan penulis dengan benar.
- Memperbaiki masalah yang dapat menyebabkan restart pada replika log biner (binlog) saat memproses sejumlah besar file log relai selama pemulihan log relai.
- Memperbaiki masalah yang menyebabkan instance pembaca database restart saat menjalankan kueri menggunakan fitur Parallel Query.
- Memperbaiki masalah yang, dalam kondisi langka, dapat menonaktifkan logging biner ketika terjadi kesalahan selama komit transaksi besar.
- Memperbaiki masalah yang dapat menyebabkan replika baca Aurora dimulai ulang jika terjadi pesanan komit transaksi langka tertentu pada instans DB penulis.
- Memperbaiki masalah yang dapat menyebabkan database dimulai ulang saat peristiwa terjadwal dibatalkan selama eksekusi pada instance yang telah diaktifkan Binlog yang Ditingkatkan.
- Memperbaiki masalah ketika instance database menggunakan volume cluster Aurora multi TB, mungkin mengalami peningkatan downtime saat restart karena kegagalan validasi bufferpool InnoDB.

Perbaikan umum

- Hak istimewa berikut telah ditambahkan ke`rds_superuser_role`:`FLUSH_OPTIMIZER_COSTS`,, `FLUSH_STATUS`FLUSH_TABLES, dan`FLUSH_USER_RESOURCES`. Untuk informasi tentang ini`rds_superuser_role`, lihat Akun Pengguna Master Amazon dengan Amazon Aurora di Panduan Pengguna Amazon Aurora. Untuk informasi lebih lanjut tentang hak istimewa dinamis ini, silakan lihat dokumentasi MySQL.
- Memperbaiki masalah saat koneksi yang diawetkan ditangani secara tidak benar selama Zero-downtime patching (ZDP) /Zero-downtime restart (ZDR) yang dapat menyebabkan klien menunggu tanpa batas waktu untuk menyelesaikan kueri.
- Memperbaiki masalah saat baris menjadi tidak dapat dibaca melalui indeks spasial selama pembaruan.

- Memperbaiki masalah saat kueri yang berisi petunjuk pengoptimal yang dibatalkan selama operasi penambalan ulang Zero-downTime atau Zero-downtime mungkin salah ditangani.
- Memperbaiki masalah saat latensi komit tidak diukur saat `innodb_flush_log_at_trx_commit` disetel ke`0`.
- Memperkenalkan optimasi untuk mengurangi penggunaan memori selama pemulihan kamus data logis ketika ada sejumlah besar tabel.
- Memperbaiki masalah yang menyebabkan `SHOW BINARY LOGS` perintah memakan waktu lebih lama untuk dijalankan di cluster tempat Enhanced Binlog diaktifkan atau sebelumnya diaktifkan. Masalah ini juga dapat menyebabkan peningkatan latensi komit jika beberapa `SHOW BINARY LOGS` perintah berjalan secara bersamaan.
- Memperbaiki masalah memori yang terkait dengan peran default dari view definer.
- Memperbaiki masalah yang dapat menyebabkan kegagalan dalam menyelesaikan proses menonaktifkan fitur penerusan tulis.
- Memperbaiki masalah yang dapat mencegah koneksi klien baru dibuat ke database saat penerusan penulisan diaktifkan.
- Memperbaiki masalah yang dapat menyebabkan instance database penulis dimulai ulang saat instance pembaca menggunakan penerusan tulis mengeksekusi pernyataan DHTML yang berisi `timestamp` nilai dan parameter `time_zone` database diatur ke “UTC”.
- Memperbaiki masalah yang menyebabkan tidak tersedianya Replika Baca Aurora atau ketidakkonsistenan definisi tabel dengan kesalahan `'Table does not exist'` pada replika karena kueri baca bersamaan pada operasi replika dan DDL pada penulis.
- Memperbaiki masalah yang dapat menyebabkan set hasil tidak lengkap saat menjalankan kueri yang melibatkan `LEFT-` atau `RIGHT-JOIN` operasi menggunakan algoritma hash-join dengan Parallel Query.
- Dimulai dengan versi Aurora MySQL ini, optimasi sisipan cepat tidak lagi diaktifkan. Untuk informasi selengkapnya, lihat peningkatan kinerja Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Upgrade dan migrasi

- Menghapus peran default yang tidak perlu dibuat selama pemutakhiran dari AMS2 ke AMS3.

Integrasi perbaikan bug MySQL Community Edition

Rilis ini mencakup semua perbaikan bug komunitas hingga dan termasuk 8.0.28. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 3.x.

- Memperbaiki masalah yang menyelesaikan kebuntuan saat`FLUSH STATUS`,`COM_CHANGE_USER`, dan `SHOW PROCESS LIST` dieksekusi secara bersamaan. (Bug #35218030)

Pembaruan mesin database Aurora MySQL 2024-06-26 (versi 3.04.3, kompatibel dengan MySQL 8.0.28)

Versi: 3.04.3

Aurora MySQL 3.04.3 umumnya tersedia. Aurora MySQL versi 3.04 kompatibel dengan MySQL 8.0.28. Untuk informasi selengkapnya tentang perubahan komunitas yang telah terjadi, lihat Catatan Rilis MySQL 8.0.

Untuk detail fitur-fitur baru di Aurora MySQL versi 3, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0. Untuk perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2, lihat Membandingkan Aurora MySQL versi 2 dan Aurora MySQL versi 3. Untuk perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition, lihat Membandingkan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition.

> **Note**
>
> Versi ini ditetapkan sebagai rilis dukungan jangka panjang (LTS). Untuk informasi selengkapnya, lihat Rilis dukungan jangka panjang (LTS) Aurora MySQL di Panduan Pengguna Amazon Aurora.
> Kami menyarankan agar Anda tidak menyetel `AutoMinorVersionUpgrade` parameter ke `true` (atau mengaktifkan peningkatan versi minor Otomatis di Konsol Manajemen AWS) untuk versi LTS. Melakukannya dapat menyebabkan cluster DB Anda ditingkatkan ke versi target berikutnya untuk kampanye Peningkatan Versi Kecil Otomatis, yang mungkin bukan versi LTS.

Rilis Aurora MySQL yang didukung saat ini adalah 2.07.9, 2.7.10, 2.11.*, 2.12.*, 3.03.*, 3.04.*, 3.05.*, 3.06.*, dan 3.07.*.

Anda dapat melakukan upgrade di tempat, memulihkan snapshot, atau memulai upgrade terkelola blue/green menggunakan Amazon RDS Blue/Green Deployment dari klaster Aurora MySQL versi 2 yang tersedia saat ini ke dalam klaster Aurora MySQL versi 3.04.3.

Untuk informasi tentang merencanakan peningkatan ke Aurora MySQL versi 3, lihat Merencanakan peningkatan versi utama untuk cluster DB MySQL Aurora. Untuk informasi umum tentang peningkatan Aurora MySQL, lihat Meningkatkan klaster DB Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Untuk informasi pemecahan masalah, lihat Pemecahan Masalah untuk peningkatan Aurora MySQL di tempat di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Memperbaiki masalah keamanan dan CVEs:

Rilis ini mencakup semua perbaikan CVE komunitas hingga dan termasuk MySQL 8.0.28. Perbaikan CVE berikut disertakan:

- CVE-2024-0853

Perbaikan ketersediaan:

- Memperbaiki masalah yang menyebabkan instance Aurora MySQL DB dimulai ulang saat menjalankan kueri paralel.
- Memperbaiki masalah yang dapat menyebabkan instans DB pembaca dimulai ulang saat membaca tabel yang sedang diubah atau dijatuhkan pada instance DB penulis.
- Memperbaiki masalah yang menyebabkan pelanggaran akses memori yang menyebabkan pelepasan objek mutex yang tidak lagi dimiliki oleh utas.
- Memperbaiki masalah yang dapat menyebabkan instance DB penulis MySQL Aurora memulai ulang saat sesi penerusan tulis ditutup saat menjalankan kueri yang diteruskan.
- Memperbaiki masalah yang menyebabkan instans DB dimulai ulang saat menangani set GTID besar pada instance berkemampuan log biner.

- Memperbaiki masalah saat memproses `INSERT` kueri pada tabel partisi InnoDB yang dapat menyebabkan penurunan memori bebas secara bertahap dalam instans DB.
- Memperbaiki masalah yang, dalam kondisi langka, dapat menyebabkan instance pembaca dimulai ulang saat melakukan `SELECT` kueri pada tabel dengan kendala kunci asing.
- Memperbaiki masalah yang dapat menyebabkan database restart ketika pemulihan kamus data InnoDB membutuhkan waktu lama selama pemulihan database.
- Memperbaiki masalah yang dapat menyebabkan database dimulai ulang ketika kendala cascading `UPDATE` atau kunci `DELETE` asing didefinisikan pada tabel di mana kolom virtual terlibat baik sebagai kolom dalam batasan kunci asing, atau sebagai anggota tabel referensi.
- Memperbaiki masalah Aurora Serverless v2 yang dapat menyebabkan restart basis data saat meningkatkan skala.

Perbaikan umum:

- Memperbaiki masalah yang memberikan nilai yang salah untuk variabel `threads_running` status saat menggunakan Aurora Global Database.
- Memperbaiki masalah yang menyebabkan instans DB dimulai ulang karena informasi pemegang kunci yang tidak akurat `rw_lock` saat menggunakan pembacaan paralel.
- Memperbaiki masalah manajemen memori yang menyebabkan penurunan memori yang dapat dibebaskan dari waktu ke waktu saat menjalankan `SELECT ... INTO OUTFILE ...` kueri.
- Memperbaiki masalah yang dapat menyebabkan instans DB dimulai ulang saat penyimpanan lokal pada instans DB mencapai kapasitas penuh.
- Memperbaiki masalah saat Skema Kinerja tidak diaktifkan saat manajemen otomatis Performance Insights diaktifkan untuk instans DB db.t4g.medium dan db.t4g.large.
- Memperbaiki masalah selama zero-downtime patching (ZDP) yang mencegah instans DB menutup koneksi klien setelah mencapai konfigurasi pelanggan dari salah satu atau. `wait_timeout` `interactive_timeout`
- Memperbaiki masalah di mana `SELECT` kueri pada instance pembaca Aurora mungkin gagal dengan tabel kesalahan tidak ada ketika tabel memiliki setidaknya satu indeks pencarian teks lengkap (FTS) dan `TRUNCATE` pernyataan sedang dijalankan pada instance DB penulis Aurora.

Peningkatan dan migrasi:

- Memperbaiki masalah yang menyebabkan peningkatan atau migrasi gagal saat target versi mesin Aurora MySQL DB adalah 3.04.0 atau lebih tinggi. Hal ini terjadi ketika parameter cluster `lower_case_table_names` DB diatur ke`1`, dan MySQL database collation tidak kompatibel dengan nama tabel huruf kecil.

Integrasi perbaikan bug MySQL Community Edition

Rilis ini mencakup semua perbaikan bug komunitas hingga dan termasuk 8.0.28. Lihat informasi yang lebih lengkap di Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 3.x.

Pembaruan mesin database Aurora MySQL 2024-03-15 (versi 3.04.2, kompatibel dengan MySQL 8.0.28)

Versi: 3.04.2

Aurora MySQL 3.04.2 umumnya tersedia. Aurora MySQL versi 3.04 kompatibel dengan MySQL 8.0.28. Untuk informasi selengkapnya tentang perubahan komunitas yang telah terjadi, lihat Catatan Rilis MySQL 8.0.

Untuk detail fitur-fitur baru di Aurora MySQL versi 3, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0. Untuk perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2, lihat Membandingkan Aurora MySQL versi 2 dan Aurora MySQL versi 3. Untuk perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition, lihat Membandingkan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition.

> **Note**
>
> Versi ini ditetapkan sebagai rilis dukungan jangka panjang (LTS). Untuk informasi selengkapnya, lihat Rilis dukungan jangka panjang (LTS) Aurora MySQL di Panduan Pengguna Amazon Aurora.
> Kami menyarankan agar Anda tidak menyetel `AutoMinorVersionUpgrade` parameter ke `true` (atau mengaktifkan peningkatan versi minor Otomatis di Konsol Manajemen AWS) untuk versi LTS. Melakukannya dapat menyebabkan cluster DB Anda ditingkatkan ke versi target berikutnya untuk kampanye Peningkatan Versi Kecil Otomatis, yang mungkin bukan versi LTS.

Rilis Aurora MySQL yang didukung saat ini adalah 2.07.9, 2.7.10, 2.11.*, 2.12.*, 3.03.*, 3.04.*, 3.05.*, dan 3.06.*.

Anda dapat melakukan upgrade di tempat, memulihkan snapshot, atau memulai upgrade terkelola blue/green menggunakan Amazon RDS Blue/Green Deployment dari klaster Aurora MySQL versi 2 yang tersedia saat ini ke dalam klaster Aurora MySQL versi 3.04.2.

Untuk informasi tentang merencanakan peningkatan ke Aurora MySQL versi 3, lihat Perencanaan peningkatan untuk Aurora MySQL versi 3 di Panduan Pengguna Amazon Aurora. Untuk informasi umum tentang peningkatan Aurora MySQL, lihat Meningkatkan klaster DB Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Untuk informasi pemecahan masalah, lihat Memecahkan masalah peningkatan dengan Aurora MySQL versi 3.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Memperbaiki masalah keamanan dan CVEs:

Perbaikan CVE berikut disertakan dalam rilis ini:

- CVE-2020-11104
- CVE-2020-11105
- CVE-2023-38545
- CVE-2023-38546
- CVE-2023-39975

Perbaikan ketersediaan:

- Memperbaiki masalah saat instans DB replika baca tidak dapat diluncurkan dengan sukses ketika ada beban kerja yang tinggi di instans DB penulis.
- Memperbaiki masalah di mana instance DB penulis MySQL Aurora dapat gagal karena cacat pada komponen yang berkomunikasi dengan penyimpanan Aurora. Cacat terjadi sebagai akibat dari gangguan dalam komunikasi antara instans DB dan penyimpanan yang mendasarinya setelah pembaruan perangkat lunak.

- Memperbaiki masalah yang dapat menyebabkan instance database dimulai ulang saat menjalankan pernyataan SHOW STATUS dan PURGE BINARY LOGS secara bersamaan. `PURGE BINARY LOGS`adalah pernyataan terkelola yang dijalankan untuk menghormati periode retensi binlog yang dikonfigurasi pengguna.
- Memperbaiki masalah yang, selama restart instance database, dapat menyebabkan restart tambahan.
- Memperbaiki masalah dengan perselisihan kunci yang disebabkan oleh thread pencatatan log audit yang dapat mengarah ke batas waktu aplikasi klien dan penggunaan CPU yang tinggi.
- Memperbaiki masalah di mana instance database MySQL Aurora dapat mengalami beberapa restart selama startup instance sementara segmen rollback besar diinisialisasi.
- Memperbaiki masalah yang dapat menyebabkan instans DB dimulai ulang saat menjalankan kueri yang mereferensikan fungsi agregat.

Perbaikan umum:

- Memperbaiki masalah yang dapat menyebabkan kueri paralel gagal karena masalah jaringan sementara saat membaca data dari volume cluster Aurora DB
- Memperbaiki masalah saat pengguna tidak dapat mengganggu kueri apa pun atau mengatur batas waktu sesi untuk `performance_schema` kueri.
- Memperbaiki masalah di mana replikasi log biner (binlog) dikonfigurasi untuk menggunakan sertifikat SSL khusus (mysql.rds_import_binlog_ssl_material) dapat gagal saat instance replikasi sedang menjalani penggantian host.
- Memperbaiki masalah yang terkait dengan manajemen file log audit yang dapat menyebabkan file log tidak dapat diakses untuk diunduh atau dirotasi, dan dalam beberapa kasus meningkatkan penggunaan CPU.
- Pemulihan `AUTO_INCREMENT` kunci yang dioptimalkan untuk mengurangi waktu penyelesaian untuk memulihkan snapshot, melakukan point-in-time pemulihan, dan mengkloning cluster DB dengan sejumlah besar tabel dalam database.
- Memperbaiki masalah di mana pernyataan SQL mengacu pada beberapa `performance_schema` tabel dapat mengembalikan kesalahan karena tabel ini hilang setelah bermigrasi dari MySQL Komunitas ke Aurora MySQL versi 3.04.0 dan 3.04.1.
- Memperbaiki masalah saat instance replika baca kecil dapat mengalami peningkatan kelambatan replikasi setelah memutakhirkan dari versi Aurora MySQL yang lebih rendah dari 2.11.*.

- Memperbaiki masalah yang dapat menyebabkan kesalahan kunci duplikat untuk `AUTO_INCREMENT` kolom yang menggunakan indeks turun setelah pemulihan snapshot, backtrack, atau operasi kloning database.
- Memperbaiki masalah yang dapat menyebabkan modifikasi parameter `table_open_cache` database tidak berlaku sampai instans DB dimulai ulang.
- Memperbaiki masalah saat instans DB pembaca tidak dapat membuka tabel, dengan ERROR 1146. Masalah ini terjadi saat menjalankan jenis pernyataan Data Definition Language (DDL) online tertentu saat `INPLACE` algoritme digunakan pada instance DB penulis.
- Memperbaiki masalah untuk mencegah instans memulai ulang selama penskalaan Aurora Serverless v2 saat proses pemantauan internal secara keliru mengirimkan permintaan penskalaan duplikat.
- Memperbaiki masalah yang dapat menyebabkan restart database saat konsumen log biner (binlog) yang terhubung menggunakan server replikasi binlog duplikat. IDs

Peningkatan dan migrasi:

- Memperbaiki masalah yang dapat menyebabkan peningkatan versi utama ke Aurora MySQL versi 3 gagal karena adanya entri yatim piatu untuk ruang meja yang sudah dihapus di tabel sistem InnoDB di Aurora MySQL versi 2.

Integrasi perbaikan bug MySQL Community Edition

Rilis ini mencakup semua perbaikan bug komunitas hingga dan termasuk 8.0.28, selain yang berikut ini. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 3.x.

- Memperbaiki masalah di mana nilai baris cache dapat dihitung secara tidak benar, menyebabkan kegagalan selama restart database pada instance berbasis Graviton. (Perbaikan Bug Komunitas #35479763)
- Berulang menjalankan rutinitas yang disimpan, memiliki sebagai subquery pernyataan SELECT yang berisi beberapa `AND`/`OR`,, atau `XOR` kondisi, menyebabkan konsumsi berlebihan dan mungkin akhirnya kelelahan memori virtual. (Perbaikan Bug Komunitas #33852530)

Pembaruan mesin database Aurora MySQL 2023-11-13 (versi 3.04.1, kompatibel dengan MySQL 8.0.28)

Versi: 3.04.1

Aurora MySQL 3.04.1 tersedia secara umum. Aurora MySQL versi 3.04 kompatibel dengan MySQL 8.0.28. Untuk informasi selengkapnya tentang perubahan komunitas yang telah terjadi, lihat Catatan Rilis MySQL 8.0.

> **Note**
>
> Versi ini ditetapkan sebagai rilis dukungan jangka panjang (LTS). Untuk informasi selengkapnya, lihat Rilis dukungan jangka panjang (LTS) Aurora MySQL di Panduan Pengguna Amazon Aurora.
> Kami menyarankan agar Anda tidak menyetel `AutoMinorVersionUpgrade` parameter ke `true` (atau mengaktifkan peningkatan versi minor Otomatis di Konsol Manajemen AWS) untuk versi LTS. Melakukannya dapat menyebabkan cluster DB Anda ditingkatkan ke versi target berikutnya untuk kampanye Peningkatan Versi Kecil Otomatis, yang mungkin bukan versi LTS.

Untuk detail fitur-fitur baru di Aurora MySQL versi 3, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0. Untuk perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2, lihat Membandingkan Aurora MySQL versi 2 dan Aurora MySQL versi 3. Untuk perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition, lihat Membandingkan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition.

Rilis Aurora MySQL yang saat ini didukung adalah 2.07.9, 2.7.10, 2.11.*, 2.12.*, 3.01.*, 3.02.*, 3.03.*, 3.04.*, dan 3.05.*.

Anda dapat melakukan upgrade di tempat, memulihkan snapshot, atau memulai upgrade terkelola blue/green menggunakan Amazon RDS Blue/Green Deployment dari klaster Aurora MySQL versi 2 yang tersedia saat ini ke dalam klaster Aurora MySQL versi 3.04.1.

Untuk informasi tentang merencanakan peningkatan ke Aurora MySQL versi 3, lihat Perencanaan peningkatan untuk Aurora MySQL versi 3 di Panduan Pengguna Amazon Aurora. Untuk informasi umum tentang peningkatan Aurora MySQL, lihat Meningkatkan klaster DB Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Untuk informasi pemecahan masalah, lihat Memecahkan masalah peningkatan dengan Aurora MySQL versi 3.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan ketersediaan:

- Memperbaiki masalah di mana instans basis data MySQL Aurora yang menggunakan kueri paralel mungkin mengalami mulai ulang basis data saat menjalankan kueri paralel dalam jumlah besar secara bersamaan.
- Memperbaiki masalah yang dapat menyebabkan set GTID yang dieksekusi dipulihkan secara tidak benar pada klaster replika log biner (binlog) dengan binlog yang ditingkatkan aktif ketika sumber binlog mana pun telah mengatur `gtid_mode` ke `ON` atau `ON_PERMISSIVE`. Masalah ini dapat menyebabkan instans penulis klaster replika memulai ulang waktu tambahan selama pemulihan, atau menyebabkan hasil yang salah saat mengkueri set GTID yang dieksekusi.
- Memperbaiki masalah manajemen memori yang dapat menyebabkan instans basis data Aurora MySQL memulai ulang atau melakukan failover karena penurunan memori yang dapat dilepas saat log biner yang ditingkatkan diaktifkan.
- Memperbaiki masalah yang dapat menyebabkan instans pembaca memulai ulang saat instans penulis meningkatkan volume basis data ke kelipatan 160GB.
- Memperbaiki masalah di mana instans basis data Aurora MySQL dengan fitur log biner yang ditingkatkan diaktifkan mungkin terhenti selama startup instans basis data saat proses pemulihan log biner sedang dijalankan.
- Memperbaiki masalah yang dapat menyebabkan instans basis data memulai ulang karena deadlatch saat menjalankan pernyataan `SHOW STATUS` dan `PURGE BINARY LOGS` secara bersamaan. Log biner pembersihan adalah pernyataan terkelola yang dijalankan untuk menghormati periode retensi binlog yang dikonfigurasi pengguna.
- Memperbaiki masalah yang dapat menyebabkan klaster basis data menjadi tidak tersedia jika instans penulis memulai ulang saat basis data membuat atau menjatuhkan pemicu pada tabel sistem internal.

- Memperbaiki masalah yang dapat menyebabkan instans basis data memulai ulang karena menunggu semaphore yang lama saat menggunakan fitur binlog yang disempurnakan pada klaster dengan replika Aurora.

Perbaikan umum:

- Memperbaiki masalah yang dapat menyebabkan basis data menjadi tidak tersedia saat binlog yang disempurnakan diaktifkan pada klaster basis data Aurora Serverless v2 yang berjalan di Aurora MySQL 3.04.0.
- Menghapus metadata penyimpanan yang tidak digunakan sebelum menulis ke Penyimpanan Aurora saat fitur binlog yang disempurnakan diaktifkan. Perbaikan ini menghindari skenario tertentu ketika basis data memulai ulang atau melakukan failover dapat terjadi karena peningkatan latensi penulisan karena peningkatan byte yang ditransmisikan melalui jaringan.
- Memperbaiki masalah di mana tabel skema kinerja khusus Aurora tidak dibuat saat peningkatan versi atau migrasi.
- Memperbaiki masalah yang dapat menyebabkan `NumBinaryLogFiles` metrik aktif menampilkan hasil yang salah saat binlog yang disempurnakan diaktifkan. CloudWatch

Peningkatan dan migrasi:

- Peningkatan dari MySQL 5.7 ke MySQL 8.0 dengan tabel yang sangat banyak dalam satu basis data menyebabkan server mengonsumsi memori yang berlebihan. Ditemukan bahwa, selama proses pemeriksaan apakah tabel dapat ditingkatkan, kami mengambil semua objek `Table` kamus data di awal, dengan memproses masing-masing objek dan mengambil namanya, kemudian menjalankan `CHECK TABLE ... FOR UPGRADE` pada daftar. Mengambil semua objek di awal tidak perlu dilakukan dalam kasus ini, dan berkontribusi besar terhadap konsumsi memori. Untuk memperbaiki masalah ini, kami sekarang mengambil satu objek `Table` satu per satu dalam kasus tersebut, dengan melakukan pemeriksaan yang diperlukan, mengambil namanya, dan melepaskan objek tersebut, sebelum melanjutkan ke objek yang berikutnya. (Bug #34526001)

Integrasi perbaikan bug MySQL Community Edition

Rilis ini meliputi semua perbaikan bug komunitas hingga dan termasuk 8.0.28, selain hal-hal di bawah ini. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 3.x.

- Memperbaiki masalah yang dapat menyebabkan penggunaan CPU yang lebih tinggi karena rotasi sertifikat TLS di latar belakang (Perbaikan Bug Komunitas #34284186)

Pembaruan mesin database Aurora MySQL 2023-07-31 (versi 3.04.0), kompatibel dengan MySQL 8.0.28)

Versi: 3.04.0

Aurora MySQL 3.04.0 tersedia secara umum. Aurora MySQL versi 3.04 kompatibel dengan MySQL 8.0.28, Aurora MySQL versi 3.03 kompatibel dengan MySQL 8.0.26, dan Aurora MySQL versi 3.02 kompatibel dengan MySQL 8.0.23. Untuk informasi selengkapnya tentang perubahan komunitas yang terjadi dari 8.0.23 ke 8.0.28, lihat Catatan Rilis MySQL 8.0.

> **Note**
>
> Versi ini ditetapkan sebagai rilis dukungan jangka panjang (LTS). Untuk informasi selengkapnya, lihat Rilis dukungan jangka panjang (LTS) Aurora MySQL di Panduan Pengguna Amazon Aurora.
> Kami menyarankan agar Anda tidak menyetel `AutoMinorVersionUpgrade` parameter ke `true` (atau mengaktifkan peningkatan versi minor Otomatis di Konsol Manajemen AWS) untuk versi LTS. Melakukannya dapat menyebabkan cluster DB Anda ditingkatkan ke versi target berikutnya untuk kampanye Peningkatan Versi Kecil Otomatis, yang mungkin bukan versi LTS.

Untuk detail fitur-fitur baru di Aurora MySQL versi 3, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0. Untuk perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2, lihat Membandingkan Aurora MySQL versi 2 dan Aurora MySQL versi 3. Untuk perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition, lihat Membandingkan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition.

Rilis Aurora MySQL yang saat ini didukung adalah 2.07.9, 2.11.1, 2.11.2, 3.01.*, 3.02.*, 3.03.*, dan 3.04.0.

Anda dapat melakukan upgrade di tempat, memulihkan snapshot, atau memulai upgrade terkelola blue/green menggunakan Amazon RDS Blue/Green Deployment dari klaster Aurora MySQL versi 2 yang didukung saat ini ke dalam klaster Aurora MySQL versi 3.04.0.

Untuk informasi tentang merencanakan peningkatan ke Aurora MySQL versi 3, lihat Perencanaan peningkatan untuk Aurora MySQL versi 3 di Panduan Pengguna Amazon Aurora. Untuk informasi umum tentang peningkatan Aurora MySQL, lihat Meningkatkan klaster DB Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Untuk informasi pemecahan masalah, lihat Memecahkan masalah peningkatan dengan Aurora MySQL versi 3.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Log biner (binlog) yang disempurnakan Aurora MySQL saat ini tidak didukung untuk instans basis data Aurora Nirserver v2 pada Aurora MySQL versi 3.04.0. Mengaktifkan fitur ini dapat menyebabkan basis data menjadi tidak tersedia. Jika Anda memerlukan penggunaan log biner yang disempurnakan pada Aurora MySQL versi 3.04.0, sebaiknya gunakan kelas instans basis data non-nirserver atau atur ACU minimum dan maksimum instans basis data Nirserver v2 ke nilai yang sama.
> Informasi selengkapnya tentang pencatatan log biner yang disempurnakan di Aurora MySQL tersedia di Panduan Pengguna Aurora.

Perbaikan

Fitur baru:

- Memperbaiki kinerja kueri menggunakan indeks teks lengkap InnoDB untuk mencari frasa dalam mode bahasa alami. Untuk informasi selengkapnya tentang pencarian teks lengkap di MySQL, lihat Fungsi Pencarian Teks Lengkap.
- Amazon Aurora MySQL mendukung penerusan penulisan lokal (dalam klaster). Sekarang Anda dapat meneruskan operasi penulisan dari instans DB pembaca ke instans DB penulis dalam klaster DB Aurora MySQL. Untuk informasi selengkapnya, lihat Menggunakan penerusan penulisan lokal di klaster DB Aurora MySQL.
- Menambahkan kemampuan untuk mengubah nilai parameter `aurora_replica_read_consistency` untuk fitur Menggunakan penerusan penulisan dalam

basis data global Amazon Aurora dalam sesi yang menonaktifkan `autocommit`. Untuk informasi selengkapnya, lihat Parameter konfigurasi untuk penerusan penulisan.

- Dimulai dengan Aurora MySQL 3.04, untuk fitur penerusan penulisan basis data global, Anda sekarang dapat mengatur nilai parameter `aurora_replica_read_consistency` melalui klaster basis data dan grup parameter instans basis data. Sebelum Aurora MySQL versi 3.04, nilai parameter ini hanya dapat dikonfigurasi pada tingkat sesi.

Memperbaiki masalah keamanan dan CVEs:

- Mengubah SSL/TLS penyedia dari OpenSSL menjadi. AWS-LC Perbaikan ini membawa sejumlah perubahan termasuk, tetapi tidak terbatas pada:
  - Koneksi basis data menggunakan SSL sekarang dapat dipulihkan oleh Zero Downtime Restart dan Zero Downtime Patching saat meningkatkan versi dari Aurora MySQL versi 3.04.0 ke versi yang lebih tinggi.
  - Support untuk TLSv1 .3 yang mencakup dukungan untuk TLS_AES_128_GCM_, TLS_AES_256_GCM_ dan TLS_ _ _ SSL ciphersSHA256. SHA384 CHACHA20 POLY1305 SHA256
  - Penghapusan dukungan untuk cipher DHE-RSA-* yang kurang aman.

  Untuk informasi selengkapnya, lihat Menggunakan klaster DB Aurora MySQL
- Menambahkan hak istimewa dinamis `SHOW_ROUTINE` ke `rds_superuser_role` yang memungkinkan akses ke definisi dan properti dari semua rutinitas yang disimpan, seperti prosedur dan fungsi yang disimpan. Untuk detail selengkapnya, lihat SHOW_ROUTINE.
- Memperbaiki masalah yang dapat menyebabkan log audit melewatkan peristiwa selama rotasi file log audit.
- Mengaktifkan dukungan untuk protokol Keamanan Lapisan Pengangkutan (TLS) 1.3 yang aman dan berkinerja baik sambil mempertahankan kompatibilitas dengan versi TLS 1.2.
- Versi TLS TLSv1 dan TLSv1 .1 tidak digunakan lagi di komunitas MySQL 8.0.26 dan dengan demikian di Aurora MySQL 3.03. Protokol-protokol ini sekarang telah dihapus di MySQL 8.0.28 komunitas dan juga di Aurora MySQL 3.04. Secara default, setiap klien aman yang tidak dapat berkomunikasi melalui TLS 1.2 atau lebih tinggi akan ditolak. Untuk informasi selengkapnya tentang cara terhubung ke instans basis data Anda menggunakan TLS, silakan lihat Keamanan dengan Amazon Aurora MySQL.

Perbaikan CVE berikut disertakan dalam rilis ini:

- CVE-2023-21963
- CVE-2023-21912
- CVE-2023-0215
- CVE-2022-43551
- CVE-2022-37434
- CVE-2022-21635
- CVE-2022-21556
- CVE-2022-21352
- CVE-2021-35630
- CVE-2021-35624

Perbaikan ketersediaan:

- Memperbaiki masalah yang dapat menyebabkan restart database selama pemulihan transaksi yang lama.
- Memperbaiki masalah dalam enkripsi peristiwa aliran aktivitas database yang dapat menyebabkan database dimulai ulang.
- Memperbaiki masalah manajemen memori karena kesalahan berupa memori habis saat pool buffer InnoDB sedang diinisialisasi selama startup atau saat penskalaan di Aurora Nirserver v2. Masalah ini mungkin telah menyebabkan restart instance database atau penurunan kinerja termasuk pengurangan throughput atau peningkatan latensi.
- Memperbaiki masalah yang dapat menyebabkan instance pembaca MySQL Aurora dimulai ulang saat menjalankan kueri yang menggunakan rencana eksekusi kueri paralel MySQL Aurora.
- Memperbaiki masalah yang, dalam situasi tertentu, dapat menyebabkan instance pembaca Aurora dimulai ulang selama estimasi rentang.
- Memperbaiki masalah yang dapat mengganggu pemulihan database selama startup jika restart terjadi saat menjalankan operasi penyisipan berat yang melibatkan kolom penambahan otomatis.
- Memperbaiki masalah pada audit lanjutan Aurora yang menyebabkan pencatatan log berlebihan pesan informasi ke log kesalahan Aurora MySQL saat variabel server `server_audit_events` disetel ke `ALL` atau `QUERY`. Masalah ini dapat menyebabkan instance database dimulai ulang.
- Memperbaiki masalah yang dapat menyebabkan restart database selama rollback `INSERT` pernyataan saat query paralel diaktifkan.

- Memperbaiki masalah yang dapat menyebabkan instance database dimulai ulang saat menjalankan alat `EXPLAIN ANALYZE` pembuatan profil pada kueri yang mengembalikan output `all select tables were optimized away` dalam kolom `EXTRA` informasi. Untuk informasi selengkapnya, lihat dokumentasi MySQL tentang Format Output `EXPLAIN`.
- Memperbaiki masalah yang dapat menyebabkan instance pembaca Region sekunder database global Aurora menggunakan penerusan penulisan global untuk memulai ulang saat pernyataan komit implisit yang diteruskan mengalami kesalahan.
- Memperbaiki masalah yang dapat menyebabkan instance penulis di Wilayah primer database global Aurora dimulai ulang saat `SELECT FOR UPDATE` kueri dijalankan menggunakan penerusan penulisan global dari Wilayah sekunder database global Aurora.

Perbaikan umum:

- Menambahkan prosedur tersimpan baru, `mysql.rds_gtid_purged`, untuk memungkinkan pelanggan mengatur variabel sistem `GTID_PURGED`. Untuk informasi selengkapnya, lihat mysql.rds_gtid_purged.
- Menambahkan dua prosedur tersimpan baru, `mysql.rds_start_replication_until` dan`mysql.rds_start_replication_until_gtid`, yang memungkinkan pelanggan mengonfigurasi lokasi untuk menghentikan replikasi log biner. Untuk informasi selengkapnya tentang cara mengonfigurasi lokasi berhenti untuk replikasi log biner di Aurora MySQL, lihat mysql.rds_start_replication_until.
- Memperbaiki masalah yang mencegah prosedur tersimpan kontrol replikasi Aurora MySQL memodifikasi variabel `sql_log_bin`, saat dipanggil dari sebuah sesi dengan mode commit otomatis dinonaktifkan.
- Menambahkan dukungan replikasi logis untuk pernyataan Data Control Language (DCL) berikut: `GRANT/REVOKE` dan `CREATE/DROP/ALTER/RENAME USER`.
- Memperbaiki masalah untuk mencegah statistik InnoDB menjadi usang, yang terkadang dapat menghasilkan rencana eksekusi kueri yang kurang optimal yang dapat menyebabkan peningkatan waktu eksekusi kueri.
- Menambahkan dua tampilan sistem baru, `information_schema.aurora_global_db_instance_status` dan `information_schema.aurora_global_db_status`. Kedua tampilan ini dapat digunakan untuk menampilkan status dan topologi sumber daya primer dan sekunder dalam klaster basis data global Aurora MySQL. Detail kedua tampilan sistem ini dapat ditemukan di sini, tabel information_schema khusus Aurora MySQL.

- Memperbaiki masalah di mana pengguna tidak dapat mengakses basis data dengan karakter wildcard dalam nama basis data setelah mengeksekusi pernyataan `SET ROLE` dengan karakter wildcard yang ditimpa.
- Memperbaiki masalah di mana peristiwa yang dilaporkan saat memproses rotasi log audit mungkin tidak ditulis ke log audit.
- Memperbaiki masalah saat membuat tabel sementara internal, melalui eksekusi `TRIGGER`, dapat menyebabkan instans basis data penulis memulai ulang.
- Menambahkan variabel sistem baru, `innodb_aurora_max_partitions_for_range`. Dalam beberapa kasus di mana statistik yang dipertahankan tidak tersedia, parameter ini dapat digunakan untuk mempercepat waktu eksekusi estimasi jumlah baris pada tabel yang dipartisi. Informasi selengkapnya dapat ditemukan di dalam dokumentasi, Parameter konfigurasi Aurora MySQL.
- Memperbaiki masalah yang mengizinkan pelanggan untuk mengatur `ROW_FORMAT` sebagai `COMPRESSED` saat membuat tabel yang dipartisi. Tabel akan secara implisit dikonversi ke format `COMPACT` dengan peringatan yang menginformasikan bahwa Aurora MySQL tidak mendukung tabel terkompresi.
- Memperbaiki masalah yang dapat menyebabkan replikasi log biner multithreaded berhenti saat `replica_parallel_type` variabel disetel `LOGICAL_CLOCK` dan variabel diputar. `replica_preserve_commit_order` `ON` Masalah ini dapat terjadi ketika transaksi yang lebih besar dari 500 MB dijalankan pada sumbernya.
- Memperbaiki masalah saat fitur penerusan penulisan basis data global diaktifkan yang dapat menyebabkan perubahan pada konfigurasi `performance_schema` pada instans pembaca di Wilayah sekunder diteruskan secara tidak sengaja ke instans penulis di Wilayah utama.
- Memperbaiki masalah di mana variabel status server `innodb_buffer_pool_reads` mungkin tidak diperbarui setelah halaman data dibaca dari sistem file penyimpanan Aurora.
- Kueri paralel Aurora MySQL tidak didukung saat memilih konfigurasi klaster I/O-Dioptimalkan Aurora. Untuk informasi selengkapnya, lihat Batasan kueri paralel Amazon Aurora MySQL.
- Memperbaiki masalah, saat kueri paralel diaktifkan, yang menyebabkan pengoptimal rencana kueri memilih rencana eksekusi yang tidak efisien untuk kueri `SELECT` tertentu yang mendapat manfaat dari indeks primer atau sekunder.
- Meningkatkan definisi zona waktu ke versi IANA 2023c.
- Memperkenalkan pengoptimalan kinerja manajemen file pada replika binlog untuk membantu mengurangi perselisihan saat menulis ke file log relay.

- Memperbaiki masalah di mana `RPO_LAG_IN_MILLISECONDS` kolom dalam `information_schema.aurora_global_db_status` tabel dan `AuroraGlobalDBRPOLag` CloudWatch metrik selalu ditampilkan nol terlepas dari beban kerja pengguna.
- Memperkenalkan parameter baru `aurora_tmptable_enable_per_table_limit`. Ketika parameter ini diaktifkan, `tmp_table_size` variabel mendefinisikan ukuran maksimum tabel sementara internal dalam memori individu yang dibuat oleh mesin TempTable penyimpanan. Untuk detail tambahan, lihat Mesin penyimpanan untuk tabel sementara internal (implisit).
- Memperbaiki masalah di mana koneksi tambahan dibuat saat fitur penerusan penulisan basis data global diaktifkan. Masalah terjadi ketika transaksi hanya-baca pada instans pembaca salah meneruskan commit implisit kepada penulis.
- Memperbaiki masalah di mana bidang `PROCESSLIST_USER` dan `PROCESSLIST_HOST` di dalam tabel `performance_schema.threads` tidak diisi pada penulis di Wilayah utama untuk koneksi yang menggunakan fitur penerusan penulisan basis data global. Informasi selengkapnya tentang tabel ini dan Skema Kinerja dapat ditemukan di Manual Referensi MySQL, Tabel thread, dan Panduan Pengguna Amazon Aurora Gambaran Umum Skema Kinerja.
- Memperbaiki masalah di mana metrik Cloudwatch `CommitLatency` menampilkan nilai yang salah untuk instans pembaca di Wilayah sekunder saat fitur penerusan penulisan basis data global digunakan. Untuk memantau latensi pernyataan DHTML yang diteruskan pada klaster basis data sekunder, disarankan untuk menggunakan metrik `ForwardingReplicaDMLLatency` dan `ForwardingWriterDMLLatency`. Latensi commit juga dapat diamati menggunakan metrik `CommitLatency` pada instans penulis Wilayah primer. Informasi lebih lanjut tersedia di Panduan Pengguna Aurora, CloudWatch metrik Amazon untuk penerusan tulis.
- Memperbaiki masalah di mana prosedur tersimpan kontrol replikasi Aurora MySQL yang digunakan untuk mengelola dan mengonfigurasi replikasi log biner salah melaporkan kesalahan ketika replikasi log biner multi-thread dikonfigurasi dengan menetapkan nilai variabel `replica_parallel_workers` lebih besar dari 0.
- Memperbaiki masalah yang dapat menyebabkan konsumsi CPU tinggi saat beberapa sesi mencoba mengakses halaman yang tidak ada di memori.

Peningkatan dan migrasi:

- Untuk melakukan peningkatan versi kecil untuk basis data global Aurora dari Aurora MySQL versi 3.01, 3.02, atau 3.03 ke Aurora MySQL versi 3.04 atau yang lebih tinggi, lihat Meningkatkan versi Aurora MySQL dengan memodifikasi versi mesin.

- Memperbaiki masalah yang dapat menyebabkan kegagalan pemeriksaan awal peningkatan versi karena kesalahan inkonsistensi skema yang dilaporkan untuk tabel `mysql.general_log_backup`, `mysql.general_log`, `mysql.slow_log_backup`, dan `mysql.slow_log` saat meningkatkan versi dari Aurora MySQL 2 ke Aurora MySQL 3. Untuk informasi selengkapnya tentang pemecahan masalah peningkatan, lihat Memecahkan masalah peningkatan dengan Aurora MySQL versi 3.
- Memperbaiki masalah yang dapat menyebabkan kegagalan peningkatan versi utama saat melakukan peningkatan ke Aurora MySQL 3 ketika definisi pemicu berisi kata kunci cadangan yang tidak diapit tanda kutip.

Integrasi perbaikan bug MySQL Community Edition

Rilis ini meliputi semua perbaikan bug komunitas hingga dan termasuk 8.0.28, selain hal-hal di bawah ini. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 3.x.

- Memperbaiki masalah di mana blok buffer yang berisi halaman tabel sementara intrinsik direlokasi selama penjelajahan halaman, sehingga menyebabkan kegagalan pernyataan (Bug# 33715694)
- InnoDB: Mencegah operasi DDL online mengakses out-of-bounds memori (Bug # 34750489, Bug # 108925)
- Memperbaiki masalah yang terkadang dapat menghasilkan hasil kueri yang salah saat memproses pernyataan SQL kompleks yang terdiri dari beberapa Ekspresi Tabel Umum bersarang (CTEs) (Bug # 34572040, Bug # 34634469, Bug # 33856374)

Pembaruan mesin database Aurora MySQL 2023-12-08 (versi 3.03.3) (Usang)

Versi: 3.03.3

Aurora MySQL 3.03.3 tersedia secara umum. Aurora MySQL versi 3.03 kompatibel dengan MySQL 8.0.26. Untuk informasi selengkapnya tentang perubahan komunitas yang terjadi dari 8.0.23 ke 8.0.28, lihat Catatan Rilis MySQL 8.0.

Untuk detail fitur-fitur baru di Aurora MySQL versi 3, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0. Untuk perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2, lihat Membandingkan Aurora MySQL versi 2 dan Aurora MySQL versi 3. Untuk perbandingan Aurora

MySQL versi 3 dan MySQL 8.0 Community Edition, lihat Membandingkan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition.

Rilis Aurora MySQL yang saat ini didukung adalah 2.07.9, 2.07.10, 2.11.*, 2.12.*, 3.01.*, 3.02.*, 3.03.*, 3.04.*, dan 3.05.*.

Anda dapat melakukan upgrade di tempat, memulihkan snapshot, atau memulai upgrade terkelola blue/green menggunakan Amazon RDS Blue/Green Deployment dari klaster Aurora MySQL versi 2 yang tersedia saat ini ke dalam klaster Aurora MySQL versi 3.03.3.

Untuk informasi tentang merencanakan peningkatan ke Aurora MySQL versi 3, lihat Perencanaan peningkatan untuk Aurora MySQL versi 3 di Panduan Pengguna Amazon Aurora. Untuk informasi umum tentang peningkatan Aurora MySQL, lihat Meningkatkan klaster DB Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Untuk informasi pemecahan masalah, lihat Memecahkan masalah peningkatan dengan Aurora MySQL versi 3.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Memperbaiki masalah keamanan dan CVEs tercantum di bawah ini:

Perbaikan dan penyempurnaan lain untuk penanganan fine-tune di lingkungan terkelola. Di bawah ini adalah beberapa perbaikan CVE tambahan:

- CVE-2023-38545

Perbaikan ketersediaan:

- Memperbaiki masalah di mana instans basis data MySQL Aurora yang menggunakan kueri paralel mungkin mengalami mulai ulang basis data saat menjalankan kueri paralel dalam jumlah besar secara bersamaan.
- Memperbaiki masalah yang dapat menyebabkan set GTID yang dieksekusi dipulihkan secara tidak benar pada klaster replika log biner (binlog) dengan binlog yang ditingkatkan aktif ketika sumber binlog mana pun memiliki `gtid_mode` yang diatur ke `ON` atau `ON_PERMISSIVE`. Masalah ini dapat

menyebabkan instans penulis klaster replika memulai ulang waktu tambahan selama pemulihan, atau menyebabkan hasil yang salah saat mengkueri set GTID yang dieksekusi.

- Memperbaiki masalah manajemen memori yang dapat menyebabkan instans basis data Aurora MySQL memulai ulang atau melakukan failover karena penurunan memori yang dapat dilepas saat log biner yang ditingkatkan diaktifkan.
- Memperbaiki masalah yang dapat menyebabkan instans pembaca memulai ulang saat instans penulis meningkatkan volume basis data ke kelipatan 160GB.
- Memperbaiki masalah di mana instans basis data Aurora MySQL dengan fitur log biner yang ditingkatkan diaktifkan mungkin terhenti selama startup instans basis data saat proses pemulihan log biner sedang dijalankan.
- Memperbaiki masalah selama zero downtime patching yang menyebabkan instans memulai ulang yang menyebabkan koneksi basis data ditutup secara tak terduga.
- Memperbaiki masalah yang dapat menyebabkan instans basis data memulai ulang karena deadlatch saat menjalankan pernyataan `SHOW STATUS` dan `PURGE BINARY LOGS` secara bersamaan. Log biner pembersihan adalah pernyataan terkelola yang dijalankan untuk menghormati periode retensi binlog yang dikonfigurasi pengguna.
- Memperbaiki masalah yang dapat menyebabkan instans basis data memulai ulang karena menunggu semaphore yang lama saat menggunakan fitur binlog yang disempurnakan pada klaster dengan replika Aurora.

Perbaikan umum:

- Menghapus metadata penyimpanan yang tidak digunakan sebelum menulis ke penyimpanan Aurora saat fitur binlog yang disempurnakan diaktifkan. Perbaikan ini menghindari skenario tertentu ketika basis data memulai ulang atau melakukan failover dapat terjadi karena peningkatan latensi penulisan karena peningkatan byte yang ditransmisikan melalui jaringan.
- Memperbaiki masalah yang dapat menyebabkan `NumBinaryLogFiles` metrik aktif menampilkan hasil yang salah saat binlog yang disempurnakan diaktifkan. CloudWatch
- Memperbaiki masalah yang dapat menyebabkan modifikasi parameter basis data `table_open_cache` tidak berlaku sampai instans basis data dimulai ulang.
- Memperbaiki masalah yang dapat menyebabkan restart database saat konsumen log biner (binlog) yang terhubung menggunakan server replikasi binlog duplikat. IDs

Integrasi perbaikan bug MySQL Community Edition

Rilis ini meliputi semua perbaikan bug komunitas hingga dan termasuk 8.0.26, selain hal-hal di bawah ini. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 3.x.

- Memperbaiki masalah yang dapat menyebabkan penggunaan CPU yang lebih tinggi karena rotasi sertifikat TLS di latar belakang (Perbaikan Bug Komunitas #34284186)

Pembaruan mesin database Aurora MySQL 2023-08-29 (versi 3.03.2) (Usang)

Versi: 3.03.2

Aurora MySQL 3.03.2 tersedia secara umum. Aurora MySQL versi 3.04 kompatibel dengan MySQL 8.0.28, Aurora MySQL versi 3.03 kompatibel dengan MySQL 8.0.26, dan Aurora MySQL versi 3.02 kompatibel dengan MySQL 8.0.23. Untuk informasi selengkapnya tentang perubahan komunitas yang terjadi dari 8.0.23 ke 8.0.28, lihat Catatan Rilis MySQL 8.0.

Untuk detail fitur-fitur baru di Aurora MySQL versi 3, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0. Untuk perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2, lihat Membandingkan Aurora MySQL versi 2 dan Aurora MySQL versi 3. Untuk perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition, lihat Membandingkan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition.

Rilis Aurora MySQL yang saat ini didukung adalah 2.07.9, 2.07.10, 2.11.*, 3.01.*, 3.02.*, 3.03.*, dan 3.04.*.

Anda dapat melakukan upgrade di tempat, memulihkan snapshot, atau memulai upgrade terkelola blue/green menggunakan Amazon RDS Blue/Green Deployment dari klaster Aurora MySQL versi 2 yang tersedia saat ini ke dalam klaster Aurora MySQL versi 3.03.2.

Untuk informasi tentang merencanakan peningkatan ke Aurora MySQL versi 3, lihat Perencanaan peningkatan untuk Aurora MySQL versi 3 di Panduan Pengguna Amazon Aurora. Untuk informasi umum tentang peningkatan Aurora MySQL, lihat Meningkatkan klaster DB Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Untuk informasi pemecahan masalah, lihat Memecahkan masalah peningkatan dengan Aurora MySQL versi 3.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Memperbaiki masalah keamanan dan CVEs:

- Memperbaiki masalah yang mungkin menyebabkan log audit melewatkan peristiwa selama rotasi file log audit.

Perbaikan CVE berikut disertakan dalam rilis ini:

- CVE-2023-21963
- CVE-2023-21912
- CVE-2023-0215
- CVE-2022-43551
- CVE-2022-37434

Perbaikan ketersediaan:

- Memperbaiki masalah yang dapat menyebabkan restart database selama pemulihan transaksi yang lama.
- Memperbaiki masalah yang dapat menyebabkan kluster database menjadi tidak tersedia saat instance penulis dimulai ulang saat database membuat atau menjatuhkan pemicu pada tabel sistem internal.
- Memperbaiki masalah yang dapat menyebabkan instance database dimulai ulang saat menjalankan kueri yang mereferensikan fungsi agregat.
- Memperbaiki masalah yang dapat menyebabkan restart database selama rollback `INSERT` pernyataan saat query paralel diaktifkan.
- Penyisipan cepat diaktifkan hanya untuk tabel InnoDB biasa di Aurora MySQL versi 3.03.2 dan lebih tinggi. Pengoptimalan ini tidak berfungsi untuk tabel sementara InnoDB. Untuk informasi selengkapnya tentang pengoptimalan sisipan cepat, lihat penyempurnaan kinerja Amazon Aurora MySQL.

Perbaikan umum:

- Memperbaiki masalah di mana instans pembaca tidak dapat membuka tabel, dengan `ERROR 1146`. Masalah ini terjadi ketika mengeksekusi jenis Data Definition Language (DDL) online tertentu sementara algoritma `INPLACE` sedang digunakan pada instans penulis.
- Memperkenalkan pengoptimalan kinerja manajemen file pada replika binlog untuk membantu mengurangi perselisihan saat menulis ke file log relay.
- Memperbaiki masalah saat kueri paralel diaktifkan yang menyebabkan pengoptimal rencana kueri memilih rencana eksekusi yang tidak efisien untuk kueri `SELECT` tertentu yang mendapat manfaat dari indeks primer atau sekunder.
- Menambahkan dukungan replikasi logis untuk pernyataan Data Control Language (DCL) berikut: `GRANT/REVOKE` dan `CREATE/DROP/ALTER/RENAME USER`.
- Kueri paralel untuk Amazon Aurora MySQL tidak didukung saat memilih konfigurasi klaster I/O-Dioptimalkan Aurora. Lihat Batasan kueri paralel untuk Aurora MySQL untuk informasi selengkapnya.

Peningkatan dan migrasi:

- Untuk melakukan peningkatan versi kecil untuk basis data global Aurora dari Aurora MySQL versi 3.01 atau 3.02 ke Aurora MySQL versi 3.03 atau yang lebih tinggi, lihat Meningkatkan versi Aurora MySQL dengan memodifikasi versi mesin.
- Memperbaiki masalah yang dapat menyebabkan kegagalan peningkatan versi utama saat melakukan peningkatan ke Aurora MySQL versi 3 ketika definisi pemicu berisi kata kunci cadangan yang tidak diapit tanda kutip.

Integrasi perbaikan bug MySQL Community Edition

Rilis ini meliputi semua perbaikan bug komunitas hingga dan termasuk 8.0.26, selain hal-hal di bawah ini. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 3.x.

- Memperbaiki masalah yang terkadang dapat menghasilkan hasil kueri yang salah saat memproses pernyataan SQL kompleks yang terdiri dari beberapa Ekspresi Tabel Umum bersarang ()CTEs. (Bug #34572040, Bug #34634469, Bug #33856374)
- InnoDB: Kondisi balapan (race condition) antara thread yang mencoba melepaskan dan memulai statistik untuk tabel yang sama menimbulkan kegagalan pernyataan. (Bug #33135425)

- InnoDB: Mencegah operasi DDL online mengakses memori. out-of-bounds (Bug #34750489, Bug #108925)

Pembaruan mesin database Aurora MySQL 2023-05-11 (versi 3.03.1) (Usang)

Versi: 3.03.1

Aurora MySQL 3.03.1 tersedia secara umum. Aurora MySQL versi 3.03 kompatibel dengan MySQL 8.0.26, dan Aurora MySQL versi 3.02 kompatibel dengan MySQL 8.0.23. Untuk informasi selengkapnya tentang perubahan komunitas yang terjadi dari 8.0.23 ke 8.0.26, lihat Catatan Rilis MySQL 8.0.

Untuk detail fitur-fitur baru di Aurora MySQL versi 3, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0. Untuk perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2, lihat Membandingkan Aurora MySQL versi 2 dan Aurora MySQL versi 3. Untuk perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition, lihat Membandingkan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition.

Rilis Aurora MySQL yang saat ini didukung adalah 2.07.9, 2.11.1, 2.11.2, 3.01.*, 3.02.*, dan 3.03.*.

Anda dapat melakukan peningkatan in-place atau memulihkan snapshot dari klaster Aurora MySQL versi 2 yang saat ini didukung ke Aurora MySQL 3.03.1.

Untuk informasi tentang merencanakan peningkatan ke Aurora MySQL versi 3, lihat Perencanaan peningkatan untuk Aurora MySQL versi 3 di Panduan Pengguna Amazon Aurora. Untuk informasi umum tentang peningkatan Aurora MySQL, lihat Meningkatkan klaster DB Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Untuk informasi pemecahan masalah, lihat Memecahkan masalah peningkatan dengan Aurora MySQL versi 3.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Fitur baru:

- Konfigurasi I/O penyimpanan Aurora Optimized tersedia mulai dari versi 3.03.1. Untuk informasi selengkapnya, lihat Konfigurasi penyimpanan klaster DB Amazon Aurora.
- Menambahkan variabel sistem baru, `innodb_aurora_max_partitions_for_range`. Dalam beberapa kasus di mana statistik yang dipertahankan tidak tersedia, parameter ini dapat digunakan untuk mempercepat waktu eksekusi estimasi jumlah baris pada tabel yang dipartisi. Informasi selengkapnya dapat ditemukan di dalam dokumentasi, Parameter konfigurasi Aurora MySQL.

Perbaikan ketersediaan:

- Memperbaiki masalah yang dapat menyebabkan instans basis data memulai ulang karena salah akses ke memori yang tidak valid ketika koneksi langsung ditutup setelah melakukan transaksi.
- Memperbaiki masalah pada Audit Lanjutan Aurora yang menyebabkan pencatatan log berlebihan pesan informasi ke log kesalahan Aurora MySQL saat variabel server `server_audit_events` disetel ke ALL atau QUERY. Masalah ini dapat menyebabkan instans basis data memulai ulang.
- Memperbaiki masalah yang, dalam situasi tertentu, dapat menyebabkan instans pembaca Aurora memulai ulang saat mencoba membaca halaman yang sudah tidak dapat diakses selama estimasi rentang.
- Memperbaiki masalah yang dapat menyebabkan instans pembaca MySQL Aurora memulai ulang saat menjalankan kueri yang menggunakan rencana eksekusi kueri paralel Aurora.
- Memperbaiki masalah di mana instans basis data yang menggunakan replikasi log biner mungkin mengalami peningkatan penggunaan CPU dan kegagalan koneksi ketika beberapa konsumen replikasi log biner dilampirkan.
- Memperbaiki masalah di mana metode akses pemindaian indeks yang tidak didukung dipertimbangkan untuk ekspresi tabel umum (CTE) saat mewujudkan tabel sementara menengah, yang dapat menyebabkan perilaku yang tidak diinginkan termasuk mulai ulang basis data atau kesalahan hasil kueri. Kami memperbaiki masalah ini dengan menghindari penggunaan metode akses pemindaian indeks yang tidak didukung pada tabel menggunakan mesin TempTable penyimpanan.
- Penyisipan cepat tidak diaktifkan dalam versi MySQL Aurora ini, karena masalah yang dapat menyebabkan ketidakkonsistenan saat menjalankan kueri seperti,, dan. `INSERT INTO SELECT FROM` Untuk informasi selengkapnya tentang pengoptimalan sisipan cepat, lihat penyempurnaan kinerja Amazon Aurora MySQL.

Perbaikan umum:

- Memperbaiki masalah yang dapat menyebabkan waktu eksekusi yang lebih lama dari yang diharapkan untuk pernyataan SHOW BINARY LOGS. Hal ini dapat menyebabkan penurunan throughput commit basis data.
- Memperbaiki masalah yang dapat menyebabkan kegagalan pada ekspor paralel untuk tabel pengguna yang memiliki kolom yang ditambahkan menggunakan fungsi ADD COLUMN instan.
- Memperbaiki masalah di mana peristiwa yang dilaporkan saat memproses rotasi log audit mungkin tidak ditulis ke log audit.
- Memperbaiki masalah yang dapat menyebabkan penipisan memori yang tersedia saat menjalankan kueri terhadap tabel INFORMATION_SCHEMA INNODB_TABLESPACES.
- Memperbaiki masalah yang mengizinkan pelanggan untuk mengatur ROW_FORMAT sebagai COMPRESSED saat membuat tabel yang dipartisi. Tabel akan secara implisit dikonversi ke format COMPACT dengan peringatan yang menginformasikan bahwa Aurora MySQL tidak mendukung tabel terkompresi.

Peningkatan dan migrasi:

- Untuk melakukan peningkatan versi kecil untuk basis data global Aurora dari Aurora MySQL versi 3.01 atau 3.02 ke Aurora MySQL versi 3.03 atau yang lebih tinggi, lihat Meningkatkan versi Aurora MySQL dengan memodifikasi versi mesin.
- Memperbaiki masalah yang dapat menyebabkan kegagalan pemeriksaan awal peningkatan versi karena kesalahan inkonsistensi skema yang dilaporkan untuk tabel `mysql.general_log_backup`, `mysql.general_log`, `mysql.slow_log_backup`, dan `mysql.slow_log` saat meningkatkan versi dari Aurora MySQL 2 ke Aurora MySQL 3. Untuk informasi selengkapnya tentang pemecahan masalah peningkatan, lihat Memecahkan masalah peningkatan dengan Aurora MySQL versi 3.

Integrasi perbaikan bug MySQL Community Edition

Rilis ini meliputi semua perbaikan bug komunitas hingga dan termasuk 8.0.26, selain hal-hal di bawah ini. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 3.x.

- Memperbaiki masalah di mana blok buffer yang berisi halaman tabel sementara intrinsik direlokasi selama penjelajahan halaman, sehingga menyebabkan kegagalan pernyataan. (Bug #33715694)

Pembaruan mesin database Aurora MySQL 2023-03-01 (versi 3.03.0) (Usang)

Versi: 3.03.0

Aurora MySQL 3.03.0 tersedia secara umum. Aurora MySQL versi 3.03 kompatibel dengan MySQL 8.0.26, dan Aurora MySQL versi 3.02 kompatibel dengan MySQL 8.0.23. Untuk informasi selengkapnya tentang perubahan komunitas yang terjadi dari 8.0.23 ke 8.0.26, lihat Catatan Rilis MySQL 8.0.

Untuk detail fitur-fitur baru di Aurora MySQL versi 3, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0. Untuk perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2, lihat Membandingkan Aurora MySQL versi 2 dan Aurora MySQL versi 3. Untuk perbandingan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition, lihat Membandingkan Aurora MySQL versi 3 dan MySQL 8.0 Community Edition.

Rilis Aurora MySQL yang didukung saat ini adalah 2.07.*, 2.11.*, 3.01.*, 3.02.*, dan 3.03.*.

Anda dapat melakukan peningkatan in-place atau memulihkan snapshot dari klaster Aurora MySQL versi 2 yang saat ini didukung ke Aurora MySQL 3.03.0.

Untuk informasi tentang merencanakan peningkatan ke Aurora MySQL versi 3, lihat Perencanaan peningkatan untuk Aurora MySQL versi 3 di Panduan Pengguna Amazon Aurora. Untuk informasi umum tentang peningkatan Aurora MySQL, lihat Meningkatkan klaster DB Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Untuk informasi pemecahan masalah, lihat Memecahkan masalah peningkatan dengan Aurora MySQL versi 3.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Memperbaiki masalah keamanan dan CVEs tercantum di bawah ini:

Perbaikan dan penyempurnaan lain untuk penanganan fine-tune di lingkungan terkelola. Di bawah ini adalah beberapa perbaikan CVE tambahan:

- CVE-2022-32221
- CVE-2022-21451
- CVE-2022-21444

Perbaikan ketersediaan:

- Memperbaiki masalah di mana kelas instans DB yang lebih besar mungkin mengalami masalah selama mulai ulang karena inisialisasi pool buffer memakan waktu lebih lama dari yang diharapkan.
- Memperbaiki masalah di mana instans DB mungkin memulai ulang selama proses pemulihan basis data saat log biner diaktifkan.
- Memperbaiki masalah yang dapat menyebabkan kegagalan koneksi pada instans pembaca saat menjalankan pernyataan Data Control Language (DCL), misalnya `GRANT` dan `REVOKE`, atau saat membuat koneksi baru pada instans penulis.
- Memperbaiki masalah di mana kueri paralel salah digunakan untuk operasi Data Manipulation Language (DML), seperti pernyataan `DELETE` dan `UPDATE`, yang saat ini tidak didukung, yang menyebabkan instans DB memulai ulang. Untuk informasi selengkapnya tentang operasi yang didukung dalam kueri paralel, lihat Keterbatasan kueri paralel Aurora MySQL.
- Memperbaiki masalah yang, dalam kasus yang sangat jarang, dapat menyebabkan replika Aurora memulai ulang selama eksekusi serentak operasi pembaruan besar atau beban kerja Data Definition Language (DDL) pada instans penulis dan operasi baca pada set tabel yang sama di replika Aurora.
- Memperbaiki masalah pada operasi penurunan skala instans pembaca Aurora Nirserver v2 yang dapat menyebabkan instans pembaca memulai ulang, dan dalam beberapa kasus yang sangat jarang, menyebabkan inkonsistensi data.
- Memperbaiki masalah yang dapat menyebabkan instans DB memulai ulang karena salah mengakses lokasi memori yang tidak valid saat koneksi ke instans DB ditutup.
- Memperbaiki masalah yang, dalam kondisi yang sangat jarang, dapat menyebabkan instans DB memulai ulang saat memproses kueri dengan klausa `GROUP BY` yang memotong kolom desimal menjadi nol angka desimal.
- Memperbaiki masalah yang dapat menyebabkan instans DB memulai ulang karena salah mengakses catatan saat menjalankan kueri rentang menggunakan indeks spasial.
- Memperbaiki masalah yang dapat menyebabkan mulai ulang instans DB pada instans replika Aurora MySQL saat tabel sementara internal melampaui nilai mmap atau memori default atau yang dikonfigurasi oleh pelanggan.

- Memperbaiki masalah di mana rotasi log Audit Lanjutan dapat menyebabkan masalah manajemen memori.
- Penyisipan cepat tidak diaktifkan dalam versi MySQL Aurora ini, karena masalah yang dapat menyebabkan ketidakkonsistenan saat menjalankan kueri seperti,, dan. `INSERT INTO SELECT FROM` Untuk informasi selengkapnya tentang pengoptimalan sisipan cepat, lihat penyempurnaan kinerja Amazon Aurora MySQL.

Perbaikan umum:

- Memperbaiki latensi kueri baca sesi penerusan tulis basis data global menggunakan pengaturan konsistensi baca GLOBAL.
- Memperbaiki masalah di mana nilai parameter `wait_timeout` tidak dihormati setelah sesi klien mengeksekusi perintah `reset_connection` atau `change_user`.
- Memperbaiki masalah di mana aplikasi mungkin mengalami peningkatan latensi saat menghubungkan ke instans DB saat instans tersebut mengalami peningkatan mendadak dalam koneksi masuk. Dua CloudWatch metrik baru, `AuroraSlowHandshakeCount` dan`AuroraSlowConnectionHandleCount`, diperkenalkan untuk membantu memecahkan masalah penundaan pembuatan koneksi untuk instans Aurora MySQL DB. Informasi lebih lanjut tentang metrik ini dapat ditemukan di dokumentasi definisi metrik CloudWatch Aurora, metrik Amazon untuk CloudWatch Amazon Aurora.
- Parameter `temptable_use_mmap` telah dihentikan, dan dukungan untuk parameter tersebut direncanakan akan dihapus dalam rilis MySQL mendatang. Untuk informasi selengkapnya, lihat Mesin penyimpanan untuk tabel sementara internal (implisit).
- Memperbaiki masalah yang dapat menyebabkan waktu eksekusi yang lebih lama dari yang diharapkan untuk pernyataan `SHOW BINARY LOGS`. Hal ini dapat menyebabkan penurunan throughput commit basis data.

Peningkatan dan migrasi:

- Untuk melakukan peningkatan versi kecil untuk basis data global Aurora dari Aurora MySQL versi 3.01 atau 3.02 ke Aurora MySQL versi 3.03 atau yang lebih tinggi, lihat Meningkatkan versi Aurora MySQL dengan memodifikasi versi mesin.
- Memperbaiki masalah yang dapat menyebabkan kegagalan peningkatan versi utama dari Aurora MySQL versi 2 ke Aurora MySQL versi 3 ketika terdapat tabel dalam jumlah besar (lebih dari 750 ribu) di dalam klaster.

- Memperbaiki masalah yang dapat menyebabkan kegagalan peningkatan versi utama dari Aurora MySQL versi 2 ke Aurora MySQL versi 3 karena migrasi tabel `mysql.innodb_table_stats` dan `mysql.innodb_index_stats` membutuhkan waktu yang lebih lama dari yang diharapkan. Masalah ini terutama memengaruhi klaster DB dengan jutaan tabel.
- Memperbaiki masalah yang dapat menyebabkan kegagalan peningkatan dari Aurora MySQL versi 2 ke Aurora MySQL versi 3 karena kesalahan inkonsistensi skema. Kesalahan ini dilaporkan oleh pemeriksa awal peningkatan untuk tabel `mysql.general_log_template` dan`mysql.slow_log_template`. Untuk informasi selengkapnya tentang pemecahan masalah peningkatan, lihat Memecahkan masalah peningkatan dengan Aurora MySQL versi 3.
- Memperbaiki masalah yang dapat menyebabkan kegagalan peningkatan dari Aurora MySQL versi 2 ke Aurora MySQL versi 3 karena kesalahan `schemaInconsistencyCheck`. Kesalahan ini disebabkan oleh inkonsistensi skema dalam tabel `mysql.table_migration_index_info`, seperti yang dilaporkan oleh `upgrade-prechecks.log`. Untuk informasi selengkapnya tentang pemecahan masalah peningkatan ke Aurora MySQL versi 3, lihat Memecahkan masalah peningkatan dengan Aurora MySQL versi 3.

Integrasi perbaikan bug MySQL Community Edition

Rilis ini meliputi semua perbaikan bug komunitas hingga dan termasuk 8.0.26, selain hal-hal di bawah ini. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 3.x.

- Memperbaiki masalah di mana beberapa jenis kolom, termasuk `JSON` dan `TEXT`, terkadang kehabisan buffer urutan jika ukurannya minimal tidak 15 kali lipat dari baris terbesar dalam urutan. Sekarang buffer urutan hanya perlu 15 kali lebih besar dari kunci urutan terbesar. (Bug #103325, Bug #105532, Bug #32738705, Bug #33501541)
- Memperbaiki masalah di mana InnoDB tidak selalu menangani beberapa nama resmi untuk partisi tabel dengan benar. (Bug #32208630)
- Memperbaiki masalah yang, dalam kondisi tertentu, dapat mengembalikan hasil yang salah karena perhitungan properti nullability yang tidak akurat saat menjalankan kueri dengan kondisi `OR`. (Bug #34060289)
- Memperbaiki masalah yang, dalam kondisi tertentu, dapat mengembalikan hasil yang salah ketika dua kondisi berikut terpenuhi:
  - tabel turunan digabungkan ke dalam blok kueri luar
  - kueri mencakup sambungan kiri dan subkueri `IN`

(Bug #34060289)

- Memperbaiki masalah di mana nilai `AUTO_INCREMENT` yang salah dihasilkan ketika nilai kolom integer maksimum terlampaui. Kesalahan tersebut disebabkan nilai kolom maksimum yang tidak dipertimbangkan. Nilai `AUTO_INCREMENT` yang valid sebelumnya seharusnya dikembalikan dalam kasus ini, sehingga menyebabkan kesalahan kunci duplikat. (Bug #87926, Bug #26906787)
- Memperbaiki masalah yang tidak memungkinkan pencabutan hak istimewa `DROP` pada Skema Kinerja. (Bug #33578113)
- Memperbaiki masalah di mana prosedur tersimpan yang berisi pernyataan `IF` menggunakan `EXISTS`, yang bertindak pada satu atau beberapa tabel yang dihapus dan dibuat ulang di antara eksekusi, tidak dieksekusi dengan benar untuk invokasi berikutnya setelah invokasi pertama. (Bug #32855634).
- Memperbaiki masalah di mana kueri yang mereferensikan tampilan di subkueri dan blok kueri luar dapat menyebabkan mulai ulang yang tidak terduga. (Bug#32324234)

Pembaruan mesin database Aurora MySQL 2023-04-17 (versi 3.02.3) (Usang)

Versi: 3.02.3

Aurora MySQL 3.02.3 tersedia secara umum. Aurora MySQL versi 3.02 kompatibel dengan MySQL 8.0.23, dan Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7.

Untuk detail fitur baru di Aurora MySQL versi 3, dan perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2 atau komunitas MySQL 8.0, lihat Membandingkan Aurora MySQL versi 2 dan Aurora MySQL versi 3 di Amazon Aurora Panduan Pengguna.

Rilis Aurora MySQL yang saat ini didukung adalah 2.07.*, 2.11.1, 2.11.2, 3.01.*, 3.02.*, dan 3.03.*.

Anda dapat melakukan peningkatan in-place atau memulihkan snapshot dari klaster Aurora MySQL versi 2 yang saat ini didukung ke Aurora MySQL 3.02.3.

Untuk informasi tentang merencanakan peningkatan ke Aurora MySQL versi 3, lihat Perencanaan peningkatan untuk Aurora MySQL versi 3 di Panduan Pengguna Amazon Aurora. Untuk prosedur peningkatan itu sendiri, lihat Meningkatkan ke Aurora MySQL versi 3 di Panduan Pengguna Amazon Aurora. Untuk informasi umum tentang peningkatan Aurora MySQL, lihat Meningkatkan klaster DB Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Untuk informasi pemecahan masalah, lihat Memecahkan masalah peningkatan dengan Aurora MySQL versi 3.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan ketersediaan:

- Memperbaiki masalah yang dapat menyebabkan instans basis data memulai ulang karena salah akses ke memori yang tidak valid ketika koneksi langsung ditutup setelah melakukan transaksi.
- Penyisipan cepat tidak diaktifkan dalam versi MySQL Aurora ini, karena masalah yang dapat menyebabkan ketidakkonsistenan saat menjalankan kueri seperti,, dan. `INSERT INTO SELECT FROM` Untuk informasi selengkapnya tentang pengoptimalan sisipan cepat, lihat penyempurnaan kinerja Amazon Aurora MySQL.

Perbaikan umum:

- Memperbaiki masalah di mana metode akses pemindaian indeks yang tidak didukung dipertimbangkan untuk ekspresi tabel umum (CTE) saat mewujudkan tabel sementara menengah, yang dapat menyebabkan perilaku yang tidak diinginkan termasuk mulai ulang basis data atau kesalahan hasil kueri. Masalah ini telah diperbaiki dengan menghindari penggunaan metode akses pemindaian indeks yang tidak didukung pada tabel menggunakan mesin TempTable penyimpanan.
- Memperbaiki masalah yang, dalam kasus yang sangat jarang, dapat menyebabkan instans pembaca Aurora MySQL memulai ulang saat mengakses tabel yang memiliki pembaruan besar atau operasi Data Definition Language (DDL) yang berjalan secara bersamaan pada instans penulis Aurora MySQL.
- Memperbaiki masalah yang, dalam situasi tertentu, dapat menyebabkan instans pembaca Aurora MySQL memulai ulang saat mencoba membaca halaman yang sudah tidak dapat diakses selama estimasi rentang.
- Memperbaiki masalah di mana instans basis data yang menggunakan replikasi log biner mungkin mengalami peningkatan penggunaan CPU dan kegagalan koneksi ketika beberapa konsumen replikasi log biner dilampirkan.
- Memperbaiki masalah yang dapat menyebabkan instans pembaca MySQL Aurora memulai ulang saat menjalankan kueri yang menggunakan rencana eksekusi kueri paralel Aurora.

Pembaruan mesin database Aurora MySQL 2022-11-18 (versi 3.02.2) (Usang)

Versi: 3.02.2

Aurora MySQL 3.02.2 tersedia secara umum. Aurora MySQL versi 3.02 kompatibel dengan MySQL 8.0.23, Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7, dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Untuk detail fitur baru di Aurora MySQL versi 3 dan perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2 atau komunitas MySQL 8.0, lihat Membandingkan Aurora MySQL versi 2 dan Aurora MySQL versi 3 di Panduan Pengguna Amazon Aurora.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Anda dapat memulihkan snapshot dari klaster Aurora MySQL versi 2 yang saat ini didukung ke Aurora MySQL 3.02.2.

Untuk informasi tentang merencanakan peningkatan ke Aurora MySQL versi 3, lihat Perencanaan peningkatan untuk Aurora MySQL versi 3 di Panduan Pengguna Amazon Aurora. Untuk prosedur peningkatan itu sendiri, lihat Meningkatkan ke Aurora MySQL versi 3 di Panduan Pengguna Amazon Aurora. Untuk informasi umum tentang peningkatan Aurora MySQL, lihat Meningkatkan klaster DB Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Untuk informasi pemecahan masalah, lihat Memecahkan masalah peningkatan dengan Aurora MySQL versi 3.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Aurora MySQL versi 3.02.2 tersedia dan kompatibel secara umum dengan MySQL 8.0.23 komunitas.

Memperbaiki masalah keamanan dan CVEs tercantum di bawah ini:

Perbaikan dan penyempurnaan lain untuk penanganan fine-tune di lingkungan terkelola. Di bawah ini adalah beberapa perbaikan CVE tambahan:

- CVE-2022-21451
- CVE-2021-36222
- CVE-2021-22926

- CVE-2022-21444

Perbaikan ketersediaan:

- Memperbaiki masalah yang dapat menyebabkan instans basis data memulai ulang karena salah mengakses memori yang tidak valid saat koneksi ke instans basis data ditutup secara eksplisit atau implisit.
- Memperbaiki masalah yang dapat menyebabkan startup basis data terganggu berulang kali pada kelas instans yang lebih besar karena inisialisasi pool buffer memakan waktu lebih lama dari yang diharapkan.
- Memperbaiki masalah yang, dalam kondisi yang sangat jarang terjadi, dapat menyebabkan instans basis data memulai ulang ketika Aurora Serverless v2 salah mencoba memperbarui cache tabel saat melakukan penskalaan.
- Memperbaiki masalah yang, dalam kondisi yang sangat jarang, dapat menyebabkan basis data memulai ulang saat memproses kueri dengan klausa GROUP BY yang memotong kolom desimal menjadi nol angka desimal.
- Penyisipan cepat tidak diaktifkan dalam versi MySQL Aurora ini, karena masalah yang dapat menyebabkan ketidakkonsistenan saat menjalankan kueri seperti,, dan. `INSERT INTO SELECT FROM` Untuk informasi selengkapnya tentang pengoptimalan sisipan cepat, lihat penyempurnaan kinerja Amazon Aurora MySQL.

Perbaikan umum:

- Memperbaiki masalah yang dapat menyebabkan kegagalan peningkatan dari Aurora MySQL versi 2 (kompatibel dengan MySQL 5.7) ke Aurora MySQL versi 3 (kompatibel dengan MySQL 8.0) karena inkonsistensi metadata dalam tabel mysql.host.
- Menambahkan perbaikan kinerja untuk mengurangi waktu peningkatan versi dari Aurora MySQL versi 2 (kompatibel dengan MySQL 5.7) ke Aurora MySQL versi 3 (kompatibel dengan MySQL 8.0). Dengan memparalelkan langkah-langkah peningkatan tertentu, waktu menjadi makin singkat saat menggunakan kelas instans yang lebih besar, seperti db.r6g.16xlarge atau db.r5.24xlarge.

- Menambahkan dukungan untuk menampilkan semua kesalahan saat meningkatkan dari Aurora MySQL versi 2 (kompatibel dengan MySQL 5.7) ke Aurora MySQL versi 3 (kompatibel dengan MySQL 8.0) ketika versi sebelumnya dibatasi untuk menampilkan hanya 50 kesalahan.
- Memperbaiki masalah yang, dalam kondisi yang sangat jarang, dapat menyebabkan penghitung kenaikan otomatis menjadi salah setelah peningkatan versi utama dari Aurora MySQL versi 2 (kompatibel dengan MySQL 5.7) ke Aurora MySQL versi 3 (kompatibel dengan MySQL 8.0).
- Memperbaiki masalah yang dapat menyebabkan kegagalan peningkatan versi utama dari Aurora MySQL versi 2 ke Aurora MySQL versi 3 karena memigrasi tabel `mysql.innodb_table_stats` dan `mysql.innodb_index_stats` membutuhkan waktu lebih lama dari yang diharapkan. Masalah ini terutama memengaruhi klaster basis data dengan tabel dalam jumlah besar (>1,5 juta).
- Memperbaiki masalah yang dapat menyebabkan kegagalan peningkatan versi utama dari Aurora MySQL versi 2 ke Aurora MySQL versi 3 karena cacat pada alur kerja peningkatan mesin AMS 8.0, yang menyebabkan catatan log terakumulasi pada volume klaster penyimpanan Aurora dan menghentikan operasi tulis normal. Masalah ini terutama memengaruhi klaster basis data dengan tabel dalam jumlah besar, kira-kira >750 ribu.
- Memperbaiki masalah yang mencegah instance idle Aurora MySQL Serverless v2 diturunkan menjadi 0,5 ACUs karena utas pembersihan MySQL tidak aktif secara tidak benar.
- Memperbaiki masalah di mana aplikasi mungkin mengalami peningkatan latensi saat menghubungkan ke instans basis data saat instans tersebut mengalami peningkatan mendadak dalam koneksi masuk.
- Memperkenalkan dua CloudWatch metrik Amazon baru untuk membantu memecahkan masalah penundaan pembuatan koneksi untuk instance database MySQL Aurora. Informasi lebih lanjut tentang AuroraSlowHandshakeCount dan AuroraSlowConnectionHandleCount metrik dapat ditemukan dalam definisi metrik CloudWatch Aurora.

Integrasi perbaikan bug MySQL Community Edition

Rilis ini meliputi semua perbaikan bug komunitas hingga dan termasuk 8.0.23, selain hal-hal di bawah ini. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 3.x.

- Memperbaiki masalah yang, dalam kondisi tertentu, dapat mengembalikan hasil yang salah karena perhitungan properti nullability yang tidak akurat saat menjalankan kueri dengan kondisi OR. (Bug #34060289)

- Memperbaiki masalah yang, dalam kondisi tertentu, dapat mengembalikan hasil yang salah ketika dua kondisi berikut terpenuhi:
  - Tabel turunan digabungkan ke dalam blok kueri luar.
  - Kueri mencakup sambungan kiri dan subkueri IN. (Bug #34060289)
- Memperbaiki masalah yang tidak memungkinkan pencabutan hak istimewa DROP pada Skema Kinerja. (Bug #33578113)
- Memperbaiki masalah di mana prosedur tersimpan yang berisi pernyataan IF menggunakan EXISTS, yang bertindak pada satu atau beberapa tabel yang dihapus dan dibuat ulang di antara eksekusi, tidak dieksekusi dengan benar untuk invokasi berikutnya setelah invokasi pertama. (MySQL Bug #32855634).
- Nilai AUTO_INCREMENT yang salah dihasilkan ketika nilai kolom integer maksimum terlampaui. Kesalahan tersebut disebabkan nilai kolom maksimum yang tidak dipertimbangkan. Nilai AUTO_INCREMENT yang valid sebelumnya seharusnya dikembalikan dalam kasus ini, sehingga menyebabkan kesalahan kunci duplikat. (Bug #87926, Bug #26906787)
- Memperbaiki masalah yang dapat menyebabkan kegagalan saat memutakhirkan kluster basis data Aurora MySQL versi 1 (Kompatibel dengan MySQL 5.6) yang berisi tabel buatan pengguna dengan tabel tertentu. IDs Penugasan tabel ini IDs dapat mengakibatkan tabel kamus data yang bertentangan IDs saat meningkatkan dari Aurora MySQL versi 2 (Kompatibel dengan MySQL 5.7) ke Aurora MySQL versi 3 (Kompatibel dengan MySQL 8.0) (Bug #33919635)

Pembaruan mesin database Aurora MySQL 2022-09-07 (versi 3.02.1) (Usang)

Versi: 3.02.1

Aurora MySQL 3.02.1 tersedia secara umum. Aurora MySQL versi 3.02 kompatibel dengan MySQL 8.0.23, Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7, dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Untuk detail fitur baru di Aurora MySQL versi 3 dan perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2 atau komunitas MySQL 8.0, lihat Membandingkan Aurora MySQL versi 2 dan Aurora MySQL versi 3 di Panduan Pengguna Amazon Aurora.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Untuk informasi tentang merencanakan peningkatan ke Aurora MySQL versi 3, lihat Perencanaan peningkatan untuk Aurora MySQL versi 3 di Panduan Pengguna Amazon Aurora. Untuk prosedur peningkatan itu sendiri, lihat Meningkatkan ke Aurora MySQL versi 3 di Panduan Pengguna Amazon Aurora. Untuk informasi umum tentang peningkatan Aurora MySQL, lihat Meningkatkan klaster DB Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Untuk informasi pemecahan masalah, lihat Memecahkan masalah peningkatan dengan Aurora MySQL versi 3.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Aurora MySQL versi 3.02.1 tersedia dan kompatibel secara umum dengan MySQL 8.0.23 komunitas.

Memperbaiki masalah keamanan dan CVEs tercantum di bawah ini:

Perbaikan dan penyempurnaan lain untuk penanganan fine-tune di lingkungan terkelola. Di bawah ini adalah beberapa perbaikan CVE tambahan:

- CVE-2022-0778

Perbaikan ketersediaan:

- Memperbaiki masalah yang dapat menyebabkan kegagalan koneksi dan latensi tinggi ketika beberapa replika log biner (binlog) MySQL dilampirkan ke simpul penulis Aurora atau ketika ada banyak kueri berjalan lama secara bersamaan dengan lonjakan permintaan koneksi baru.
- Memperbaiki masalah yang menyebabkan mulai ulang basis data saat audit lanjutan untuk peristiwa `CONNECT` diaktifkan.
- Memperbaiki masalah yang dapat menyebabkan mulai ulang basis data pada instans replika baca Aurora MySQL saat tabel sementara internal menghabiskan ukuran yang dialokasikan dalam memori dan file mmap yang ditetapkan sebagai nilai yang dikonfigurasi pelanggan atau default.
- Memperbaiki masalah yang dapat menyebabkan replika baca berulang kali selama operasi DDL bersamaan pada prosedur tersimpan.
- Penyisipan cepat tidak diaktifkan dalam versi MySQL Aurora ini, karena masalah yang dapat menyebabkan ketidakkonsistenan saat menjalankan kueri seperti,, dan. `INSERT INTO SELECT`

`FROM` Untuk informasi selengkapnya tentang pengoptimalan sisipan cepat, lihat penyempurnaan kinerja Amazon Aurora MySQL.

Perbaikan umum:

- Menambahkan dukungan untuk instans R6i.

Informasi Tambahan:

- Aurora MySQL versi 3.02.1 tidak berisi dukungan untuk peningkatan versi utama langsung dari Aurora MySQL versi 2 (kompatibel dengan MySQL 5.7). Untuk melakukan peningkatan versi utama ke versi ini, pertama-tama lakukan peningkatan versi utama ke Aurora MySQL versi 3.02.0, kemudian lakukan peningkatan versi kecil secara in-place ke Aurora MySQL versi 3.02.1.

Pembaruan mesin database Aurora MySQL 2022-04-20 (versi 3.02.0) (Usang)

Versi: 3.02.0

Aurora MySQL 3.02.0 tersedia secara umum. Aurora MySQL versi 3.02 kompatibel dengan MySQL 8.0.23, Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7, dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Untuk detail fitur baru di Aurora MySQL versi 3 dan perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2 atau komunitas MySQL 8.0, lihat Membandingkan Aurora MySQL versi 2 dan Aurora MySQL versi 3 di Panduan Pengguna Amazon Aurora.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Anda dapat memulihkan snapshot dari klaster Aurora MySQL versi 2 yang saat ini didukung ke Aurora MySQL 3.02.0.

Untuk informasi tentang merencanakan peningkatan ke Aurora MySQL versi 3, lihat Perencanaan peningkatan untuk Aurora MySQL versi 3 di Panduan Pengguna Amazon Aurora. Untuk prosedur peningkatan itu sendiri, lihat Meningkatkan ke Aurora MySQL versi 3 di Panduan Pengguna Amazon Aurora. Untuk informasi umum tentang peningkatan Aurora MySQL, lihat Meningkatkan klaster DB Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Untuk informasi pemecahan masalah, lihat Memecahkan masalah peningkatan dengan Aurora MySQL versi 3.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Aurora MySQL versi 3.02.0 tersedia dan kompatibel secara umum dengan MySQL 8.0.23 komunitas.

Memperbaiki masalah keamanan dan CVEs tercantum di bawah ini:

Perbaikan dan penyempurnaan lain untuk penanganan fine-tune di lingkungan terkelola. Di bawah ini adalah beberapa perbaikan CVE tambahan:

- CVE-2021-22946

Fitur baru:

- Amazon Aurora Serverless v2 tersedia secara umum. Untuk informasi selengkapnya, lihat gambaran umum Amazon Aurora Nirserver, blog, dan dokumentasi Menggunakan Aurora Nirserver v2. Mulai sekarang dengan membuat basis data Aurora Serverless v2 hanya dengan menggunakan beberapa langkah di Konsol Manajemen AWS.

Perbaikan ketersediaan:

- Memperbaiki masalah yang dapat menyebabkan server berpotensi masuk ke mulai ulang tanpa henti dan menyebabkan hilangnya ketersediaan saat menghapus catatan atau menghapus tabel yang berisi kolom dengan panjang dua atau lebih variabel (tipe VARCHAR, VARBINARY, BLOB, dan TEXT). Untuk detail selengkapnya tentang jenis kolom, lihat innodb-row-format.
- Memperbaiki masalah di mana batas waktu koneksi yang ada dan koneksi baru tidak dapat dibuat pada klaster dengan Log Biner diaktifkan dan memiliki setidaknya satu konsumen Log Biner yang dilampirkan yang menghasilkan perselisihan sumber daya antara aplikasi dan konsumen.
- Memori yang dapat dibebaskan ditunjukkan oleh `FreeableMemory` CloudWatch metrik. Untuk informasi selengkapnya, lihat CloudWatch metrik Amazon untuk Amazon Aurora.

  - Memperbaiki masalah yang dapat menyebabkan instans basis data memulai ulang atau melakukan failover karena penurunan memori yang dapat dilepas saat replikasi log biner diaktifkan.
  - Memperbaiki masalah yang dapat menyebabkan instans basis data memulai ulang atau melakukan failover karena penurunan memori yang dapat dilepas saat mengatur variabel sesi.
  - Memperbaiki masalah yang dapat menyebabkan instans basis data memulai ulang atau melakukan failover karena penurunan memori yang dapat dilepas saat proses basis data membuka file yang ada.
- Memperbaiki masalah yang, dalam kondisi yang sangat jarang, dapat menyebabkan kesalahan entri duplikat saat memasukkan baris baru ke dalam tabel yang berisi kolom `AUTO_INCREMENT` pada klaster yang dipulihkan dari snapshot.
- Penyisipan cepat tidak diaktifkan dalam versi MySQL Aurora ini, karena masalah yang dapat menyebabkan ketidakkonsistenan saat menjalankan kueri seperti,, dan. `INSERT INTO SELECT FROM` Untuk informasi selengkapnya tentang pengoptimalan sisipan cepat, lihat penyempurnaan kinerja Amazon Aurora MySQL.

Perbaikan umum:

- Memperbaiki masalah di mana status volume tidak ditampilkan saat menggunakan perintah `SHOW VOLUME STATUS`. Untuk informasi selengkapnya, lihat AuroraMySql.Managing. VolumeStatus.
- Memperbaiki masalah yang dapat menyebabkan kegagalan panggilan ke mysql_rds_import_binlog_ssl_material dengan KESALAHAN server MySQL 3512.
- Memperbaiki masalah kesalahan pelaporan lag replika Aurora untuk instans pembaca Aurora yang dihapus.

Peningkatan/Migrasi:

- Memperbaiki masalah yang dapat menyebabkan kegagalan migrasi basis data MySQL 8.0.x ke Aurora MySQL versi 3 karena masalah dalam menyalin file ibdata dan ruang tabel ke penyimpanan Aurora.
- Memperbaiki masalah yang dapat menyebabkan kegagalan peningkatan klaster dari Aurora MySQL versi 2 ke Aurora MySQL versi 3 ketika tabel basis data berisi data dalam jumlah besar.
- Memperbaiki masalah yang dapat menyebabkan kegagalan saat memulihkan klaster dari Aurora MySQL versi 2 ke Aurora MySQL versi 3 karena kegagalan dalam membuat informasi kamus data serial (SDI) untuk tabel.

- Memperbaiki masalah yang dapat menyebabkan kegagalan peningkatan dari Aurora MySQL versi 2 ke Aurora MySQL versi 3 karena kesalahan inkonsistensi skema yang dilaporkan oleh pemeriksaan awal peningkatan untuk tabel sistem RDS.
- Memperbaiki masalah yang dapat menyebabkan kegagalan saat bermigrasi atau memulihkan dari RDS untuk MySQL 8.0 atau Aurora MySQL versi 2 ke basis data Aurora MySQL versi 3 karena sintaks tidak valid dalam prosedur tersimpan yang dikelola RDS.
- Memperbaiki masalah yang dapat menyebabkan kegagalan peningkatan dari Aurora MySQL 2 ke Aurora MySQL 3 karena kesalahan inkonsistensi skema yang dilaporkan oleh pemeriksaan awal peningkatan untuk tabel general log dan slow log.

Integrasi perbaikan bug MySQL Community Edition

Rilis ini meliputi semua perbaikan bug komunitas hingga dan termasuk 8.0.23, selain hal-hal di bawah ini. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 3.x.

- Memperbaiki penanganan yang tidak tepat pada tabel sementara yang digunakan untuk kursor di dalam prosedur tersimpan yang dapat mengakibatkan perilaku server yang tidak terduga, mysqld-8-0-24-bug. (Bug #32416811)

Pembaruan mesin database Aurora MySQL 2022-04-15 (versi 3.01.1) (Usang)

Versi: 3.01.1

Aurora MySQL 3.01.1 tersedia secara umum. Aurora MySQL versi 3.01 kompatibel dengan MySQL 8.0.23, Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7, dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Untuk detail fitur baru di Aurora MySQL versi 3 dan perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2 atau komunitas MySQL 8.0, lihat Membandingkan Aurora MySQL versi 2 dan Aurora MySQL versi 3 di Panduan Pengguna Amazon Aurora.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Anda dapat memulihkan snapshot dari klaster Aurora MySQL versi 2 yang saat ini didukung ke Aurora MySQL 3.01.1.

Untuk informasi tentang merencanakan peningkatan ke Aurora MySQL versi 3, lihat Perencanaan peningkatan untuk Aurora MySQL versi 3 di Panduan Pengguna Amazon Aurora. Untuk prosedur peningkatan itu sendiri, lihat Meningkatkan ke Aurora MySQL versi 3 di Panduan Pengguna Amazon Aurora. Untuk informasi umum tentang peningkatan Aurora MySQL, lihat Meningkatkan klaster DB Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Untuk informasi pemecahan masalah, lihat Memecahkan masalah peningkatan dengan Aurora MySQL versi 3.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Aurora MySQL versi 3.01.1 tersedia dan kompatibel secara umum dengan MySQL 8.0.23 komunitas.

Aurora MySQL versi 3.01.1 direkomendasikan untuk peningkatan dan migrasi ke basis data Aurora yang kompatibel dengan MySQL 8.0.

Memperbaiki masalah keamanan dan CVEs tercantum di bawah ini:

Perbaikan dan penyempurnaan lain untuk penanganan fine-tune di lingkungan terkelola. Di bawah ini adalah beberapa perbaikan CVE tambahan:

- CVE-2021-36222
- CVE-2021-22946
- CVE-2021-22926

Perbaikan ketersediaan:

- Memori yang dapat dibebaskan ditunjukkan oleh `FreeableMemory` CloudWatch metrik. Untuk informasi selengkapnya, lihat CloudWatch metrik Amazon untuk Amazon Aurora.
  - Memperbaiki masalah yang dapat menyebabkan instans basis data memulai ulang atau melakukan failover karena penurunan memori yang dapat dilepas saat replikasi log biner diaktifkan.

  - Memperbaiki masalah yang dapat menyebabkan instans basis data memulai ulang atau melakukan failover karena penurunan memori yang dapat dilepas saat mengatur variabel sesi.
  - Memperbaiki masalah yang dapat menyebabkan instans basis data memulai ulang atau melakukan failover karena penurunan memori yang dapat dilepas saat proses basis data membuka file yang ada.
- Memperbaiki masalah yang, dalam kondisi yang sangat jarang, dapat menyebabkan kesalahan entri duplikat saat memasukkan baris baru ke dalam tabel yang berisi kolom `AUTO_INCREMENT` pada klaster yang dipulihkan dari snapshot.
- Penyisipan cepat tidak diaktifkan dalam versi MySQL Aurora ini, karena masalah yang dapat menyebabkan ketidakkonsistenan saat menjalankan kueri seperti,, dan. `INSERT INTO SELECT FROM` Untuk informasi selengkapnya tentang pengoptimalan sisipan cepat, lihat penyempurnaan kinerja Amazon Aurora MySQL.

Perbaikan umum:

- Memperbaiki masalah di mana status volume tidak ditampilkan saat menggunakan perintah `SHOW VOLUME STATUS`. Untuk informasi selengkapnya, lihat AuroraMySql.Managing. VolumeStatus.
- Memperbaiki masalah yang dapat menyebabkan kegagalan panggilan ke mysql_rds_import_binlog_ssl_material dengan KESALAHAN server MySQL 3512.
- Memperbaiki masalah kesalahan pelaporan lag replika Aurora untuk instans pembaca Aurora yang dihapus.

Peningkatan/Migrasi:

- Memperbaiki masalah yang dapat menyebabkan kegagalan migrasi basis data MySQL 8.0.x ke Aurora MySQL versi 3 karena masalah dalam menyalin file ibdata dan ruang tabel ke penyimpanan Aurora.
- Memperbaiki masalah yang dapat menyebabkan kegagalan peningkatan klaster dari Aurora MySQL versi 2 ke Aurora MySQL versi 3 ketika tabel basis data berisi data dalam jumlah besar.
- Memperbaiki masalah yang dapat menyebabkan kegagalan saat memulihkan klaster dari Aurora MySQL versi 2 ke Aurora MySQL versi 3 karena kegagalan dalam membuat informasi kamus data serial (SDI) untuk tabel.
- Memperbaiki masalah yang dapat menyebabkan kegagalan peningkatan dari Aurora MySQL versi 2 ke Aurora MySQL versi 3 karena kesalahan inkonsistensi skema yang dilaporkan oleh pemeriksaan awal peningkatan untuk tabel sistem RDS.

- Memperbaiki masalah yang dapat menyebabkan kegagalan saat bermigrasi atau memulihkan dari RDS untuk MySQL 8.0 atau Aurora MySQL versi 2 ke basis data Aurora MySQL versi 3 karena sintaks tidak valid dalam prosedur tersimpan yang dikelola RDS.
- Memperbaiki masalah yang dapat menyebabkan kegagalan peningkatan dari Aurora MySQL 2 ke Aurora MySQL 3 karena kesalahan inkonsistensi skema yang dilaporkan oleh pemeriksaan awal peningkatan untuk tabel general log dan slow log.

Integrasi perbaikan bug MySQL Community Edition

Rilis ini meliputi semua perbaikan bug komunitas hingga dan termasuk 8.0.23, selain hal-hal di bawah ini. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 3.x.

- Memperbaiki penanganan yang tidak tepat pada tabel sementara yang digunakan untuk kursor di dalam prosedur tersimpan yang dapat mengakibatkan perilaku server yang tidak terduga, mysqld-8-0-24-bug. (Bug #32416811)

Pembaruan mesin database Aurora MySQL 2021-11-18 (versi 3.01.0) (Usang)

Versi: 3.01.0

Aurora MySQL 3.01.0 tersedia secara umum. Aurora MySQL versi 3.01 kompatibel dengan MySQL 8.0.23, Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7, dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Untuk detail fitur baru di Aurora MySQL versi 3 dan perbedaan antara Aurora MySQL versi 3 dan Aurora MySQL versi 2 atau komunitas MySQL 8.0, lihat Membandingkan Aurora MySQL versi 2 dan Aurora MySQL versi 3 di Panduan Pengguna Amazon Aurora.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Anda dapat memulihkan snapshot dari klaster Aurora MySQL versi 2 yang saat ini didukung ke Aurora MySQL 3.01.0.

Untuk informasi tentang merencanakan peningkatan ke Aurora MySQL versi 3, lihat Perencanaan peningkatan untuk Aurora MySQL versi 3 di Panduan Pengguna Amazon Aurora. Untuk prosedur

peningkatan itu sendiri, lihat Meningkatkan ke Aurora MySQL versi 3 di Panduan Pengguna Amazon Aurora. Untuk informasi umum tentang peningkatan Aurora MySQL, lihat Meningkatkan klaster DB Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Untuk informasi pemecahan masalah, lihat Memecahkan masalah peningkatan dengan Aurora MySQL versi 3.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Aurora MySQL versi 3.01.0 umumnya kompatibel dengan MySQL 8.0.23 komunitas. Versi ini mencakup perbaikan keamanan untuk masalah Kerentanan dan Eksposur Umum (CVE) pada MySQL 8.0.23 komunitas.

Aurora MySQL versi 3.01.0 berisi semua perbaikan bug khusus Aurora hingga Aurora MySQL versi 2.10.0.

Untuk detail tentang fitur-fitur baru di Aurora MySQL versi 3, lihat Fitur dari MySQL 8.0 komunitas dan Pengoptimalan kueri paralel baru di Panduan Pengguna Amazon Aurora.

Perbaikan ketersediaan:

- Penyisipan cepat tidak diaktifkan dalam versi MySQL Aurora ini, karena masalah yang dapat menyebabkan ketidakkonsistenan saat menjalankan kueri seperti,, dan. `INSERT INTO SELECT FROM` Untuk informasi selengkapnya tentang pengoptimalan sisipan cepat, lihat penyempurnaan kinerja Amazon Aurora MySQL.

Pembaruan mesin basis data untuk Amazon Aurora MySQL versi 2

> **Important**
>
> Aurora MySQL versi 2 mencapai akhir dukungan standar pada 31 Oktober 2024. Untuk informasi selengkapnya, lihat Mempersiapkan dukungan standar versi 2 Edisi yang kompatibel dengan Amazon Aurora MySQL versi 2.

Berikut ini adalah pembaruan mesin basis data untuk Amazon Aurora MySQL versi 2.

- Pembaruan mesin database Aurora MySQL 2025-04-09 (versi 2.12.5, kompatibel dengan MySQL 5.7.44) - Versi Dukungan Diperpanjang RDS
- Pembaruan mesin database Aurora MySQL 2024-10-23 (versi 2.12.4, kompatibel dengan MySQL 5.7.44) - Versi Dukungan Diperpanjang RDS
- Pembaruan mesin database Aurora MySQL 2024-07-09 (versi 2.12.3, kompatibel dengan MySQL 5.7.44) - Versi Dukungan Diperpanjang RDS
- Pembaruan mesin database Aurora MySQL 2024-03-19 (versi 2.12.2, kompatibel dengan MySQL 5.7.44) - Versi Dukungan Diperpanjang RDS
- Pembaruan mesin database Aurora MySQL 2023-12-28 (versi 2.12.1, kompatibel dengan MySQL 5.7.40) - Versi Dukungan Diperpanjang RDS
- Pembaruan mesin database Aurora MySQL 2023-10-25 (versi 2.12.0.1, kompatibel dengan MySQL 5.7.40) - Versi Dukungan Diperpanjang RDS (Beta)
- Pembaruan mesin database Aurora MySQL 2023-07-25 (versi 2.12.0, kompatibel dengan MySQL 5.7.40) - Versi Dukungan Diperpanjang RDS
- Pembaruan mesin database Aurora MySQL 2024-07-19 (versi 2.11.6, kompatibel dengan MySQL 5.7.12) - Versi Dukungan Diperpanjang RDS
- Pembaruan mesin database Aurora MySQL 2024-03-26 (versi 2.11.5, kompatibel dengan MySQL 5.7.12) - Versi Dukungan Diperpanjang RDS (Default)
- Pembaruan mesin database Aurora MySQL 2023-10-17 (versi 2.11.4, kompatibel dengan MySQL 5.7.12) - Versi Dukungan Diperpanjang RDS
- Pembaruan mesin database Aurora MySQL 2023-06-09 (versi 2.11.3, kompatibel dengan MySQL 5.7.12) - Versi Dukungan Diperpanjang RDS

- Pembaruan mesin database Aurora MySQL 2023-03-24 (versi 2.11.2, kompatibel dengan MySQL 5.7.12) - Versi Dukungan Diperpanjang RDS
- Pembaruan mesin database Aurora MySQL 2023-02-14 (versi 2.11.1, kompatibel dengan MySQL 5.7.12) - Versi Dukungan Diperpanjang RDS
- Pembaruan mesin database Aurora MySQL 2022-10-25 (versi 2.11.0, kompatibel dengan MySQL 5.7.12) - Versi Dukungan Diperpanjang RDS
- Pembaruan mesin basis data Aurora MySQL 2022-11-01 (versi 2.10.3) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2022-01-26 (versi 2.10.2) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2021-10-21 (versi 2.10.1) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2021-05-25 (versi 2.10.0) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2021-11-12 (versi 2.09.3) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2021-02-26 (versi 2.09.2) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2020-12-11 (versi 2.09.1) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2020-09-17 (versi 2.09.0) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2020-01-06 (versi 2.08.4) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2020-11-12 (versi 2.08.3) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2020-08-28 (versi 2.08.2) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2020-06-18 (versi 2.08.1) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2020-06-02 (versi 2.08.0) (Dihentikan)
- Pembaruan mesin database Aurora MySQL 2023-08-15 (versi 2.07.10) (Usang)
- Pembaruan mesin database Aurora MySQL 2023-05-04 (versi 2.07.9) (Usang)
- Pembaruan mesin basis data Aurora MySQL 2022-06-16 (versi 2.07.8) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2021-11-24 (versi 2.07.7) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2021-09-02 (versi 2.07.6) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2021-07-06 (versi 2.07.5) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2021-03-04 (versi 2.07.4) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2020-11-10 (versi 2.07.3) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2020-04-17 (versi 2.07.2) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2019-12-23 (versi 2.07.1) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2019-11-25 (versi 2.07.0) (Dihentikan)

- Pembaruan mesin basis data Aurora MySQL 2019-11-22 (versi 2.06.0) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2019-11-11 (versi 2.05.0) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2020-08-14 (versi 2.04.9) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2019-11-20 (versi 2.04.8) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2019-11-14 (versi 2.04.7) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2019-09-19 (versi 2.04.6) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2019-07-08 (versi 2.04.5) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2019-05-29 (versi 2.04.4) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2019-05-09 (versi 2.04.3) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2019-05-02 (versi 2.04.2) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2019-03-25 (versi 2.04.1) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2019-03-25 (versi 2.04.0) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2019-02-07 (versi 2.03.4) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2019-01-18 (versi 2.03.3) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2019-01-09 (versi 2.03.2) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2018-10-24 (versi 2.03.1) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2018-10-11 (versi 2.03) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2018-10-08 (versi 2.02.5) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2018-09-21 (versi 2.02.4) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2018-08-23 (versi 2.02.3) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2018-06-04 (versi 2.02.2) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2018-05-03 (versi 2.02) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2018-03-13 (versi 2.01.1) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2018-02-06 (versi 2.01) (Dihentikan)

Pembaruan mesin database Aurora MySQL 2025-04-09 (versi 2.12.5, kompatibel dengan MySQL 5.7.44) - Versi Dukungan Diperpanjang RDS

Versi: 2.12.5

Aurora MySQL 2.12.5 umumnya tersedia. Versi Aurora MySQL 2.12 kompatibel hingga MySQL 5.7.44. Untuk informasi selengkapnya tentang perubahan komunitas, lihat Perubahan di MySQL 5.7.44 (2022-10-11, Ketersediaan Umum).

Rilis Aurora MySQL yang didukung saat ini adalah 2.11.*, 2.12.*, 3.04.*, 3.05.*, 3.06.*, 3.07.*, dan 3.08*.

Anda dapat meningkatkan cluster database Aurora MySQL 2.* yang ada ke Aurora MySQL 2.12.5. Anda juga dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.12.5.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Memperbaiki masalah keamanan dan CVEs:

Rilis ini mencakup semua perbaikan CVE komunitas hingga dan termasuk MySQL 5.7.44. Perbaikan CVE berikut disertakan:

- CVE-2025-21543
- CVE-2025-21500
- CVE-2025-21491
- CVE-2025-21490
- CVE-2025-21540
- CVE-2025-21559

- CVE-2025-21555
- CVE-2025-21497
- CVE-2025-21520
- CVE-2025-21501
- CVE-2024-37371
- CVE-2024-11053
- CVE-2024-21201
- CVE-2024-21241
- CVE-2024-21230
- CVE-2023-44487

Perbaikan ketersediaan:

- Memperbaiki masalah pada replika di mana gangguan jaringan mungkin tidak membangun kembali koneksi dengan penulis dengan benar.

Pembaruan mesin database Aurora MySQL 2024-10-23 (versi 2.12.4, kompatibel dengan MySQL 5.7.44) - Versi Dukungan Diperpanjang RDS

Versi: 2.12.4

Aurora MySQL 2.12.4 umumnya tersedia. Versi Aurora MySQL 2.12 kompatibel hingga MySQL 5.7.44. Untuk informasi selengkapnya tentang perubahan komunitas, lihat Perubahan di MySQL 5.7.44 (2022-10-11, Ketersediaan Umum).

Rilis Aurora MySQL yang didukung saat ini adalah 2.11.*, 2.12.*, 3.04.*, 3.05.*, 3.06.*, dan 3.07.*.

Anda dapat meningkatkan cluster database Aurora MySQL 2.* yang ada ke Aurora MySQL 2.12.4. Anda juga dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.12.4.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Memperbaiki masalah keamanan dan CVEs:

Rilis ini mencakup semua perbaikan CVE komunitas hingga dan termasuk MySQL 5.7.44. Perbaikan CVE berikut disertakan:

- CVE-2023-44487
- CVE-2024-21142
- CVE-2024-21177
- CVE-2024-25062

Integrasi perbaikan bug MySQL Community Edition

Rilis ini mencakup semua perbaikan bug komunitas hingga dan termasuk 5.7.44. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 2.x.

Fitur yang tidak didukung di Aurora MySQL versi 2

Fitur-fitur berikut saat ini tidak didukung di Aurora MySQL versi 2 (kompatibel dengan MySQL 5.7).

- Pindai batching

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Pernyataan SQL `CREATE TABLESPACE`
- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Protokol X

Pembaruan mesin database Aurora MySQL 2024-07-09 (versi 2.12.3, kompatibel dengan MySQL 5.7.44) - Versi Dukungan Diperpanjang RDS

Versi: 2.12.3

Aurora MySQL 2.12.3 umumnya tersedia. Versi Aurora MySQL 2.12 kompatibel hingga MySQL 5.7.44. Untuk informasi selengkapnya tentang perubahan komunitas, lihat Perubahan di MySQL 5.7.44 (2022-10-11, Ketersediaan Umum).

Rilis Aurora MySQL yang didukung saat ini adalah 2.11.*, 2.12.*, 3.03.*, 3.04.*, 3.05.*, 3.06.*, dan 3.07.*.

Anda dapat meningkatkan cluster database Aurora MySQL 2.* yang ada ke Aurora MySQL 2.12.3. Anda juga dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.12.3.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Memperbaiki masalah keamanan dan CVEs:

- Memperbaiki masalah keamanan untuk prosedur tersimpan MySQL.

Rilis ini mencakup semua perbaikan CVE komunitas hingga dan termasuk MySQL 5.7.44. Perbaikan CVE berikut disertakan:

- CVE-2023-21912
- CVE-2023-44487
- CVE-2024-0853
- CVE-2024-20993
- CVE-2024-20998
- CVE-2024-21008
- CVE-2024-21009
- CVE-2024-21013
- CVE-2024-21047
- CVE-2024-21054
- CVE-2024-21055
- CVE-2024-21057
- CVE-2024-21062
- CVE-2024-21069
- CVE-2024-21096
- CVE-2024-21097

Perbaikan ketersediaan:

- Memperbaiki masalah yang menyebabkan instance Aurora MySQL DB dimulai ulang saat menjalankan kueri paralel.
- Memperbaiki masalah yang dapat menyebabkan server database dimulai ulang karena akses sumber daya koneksi secara bersamaan selama penskalaan tanpa batas, nol downtime restart (ZDR), dan nol downtime patching (ZDP).
- Memperbaiki masalah yang dapat menyebabkan instans DB pembaca dimulai ulang saat membebaskan memori yang digunakan untuk aplikasi log.
- Memperbaiki masalah dalam proses latar belakang yang menyebabkan kueri berjalan lama atau gagal saat operasi latar belakang menurunkan indeks sementara.
- Memperbaiki masalah dengan rutinitas startup yang dapat menyebabkan instance DB penulis dimulai ulang karena inkonsistensi metadata.
- Menambahkan indikator untuk kemajuan pemulihan transaksi. Ini menghindari potensi tidak tersedianya, dalam situasi yang jarang terjadi, ketika pemulihan transaksi membutuhkan waktu lama untuk diselesaikan.
- Memperbaiki masalah yang dapat menyebabkan instans DB pembaca dimulai ulang saat membaca tabel yang sedang diubah atau dijatuhkan pada instance DB penulis.
- Memperbaiki masalah di mana nilai `thread_stack` parameter rendah menyebabkan database restart berulang kali. `thread_stack`Nilai minimum yang diizinkan telah ditingkatkan dari 131.072 menjadi 136.192 untuk memastikan booting berhasil dan mencegah masalah startup.
- Memperbaiki masalah yang menyebabkan instans DB pembaca dimulai ulang saat menjalankan kueri paralel.
- Memperbaiki masalah yang dapat menyebabkan replika baca Aurora dimulai ulang jika terjadi pesanan komit transaksi langka tertentu pada instans DB penulis.
- Memperbaiki masalah yang, dalam kasus yang jarang terjadi, dapat menyebabkan instans DB dimulai ulang ketika transaksi hanya-baca memperoleh kunci bersama.
- Memperbaiki masalah yang dapat menyebabkan instans DB pembaca yang menggunakan penerusan tulis untuk memulai ulang saat pernyataan komit implisit yang diteruskan mengalami kesalahan.

Perbaikan umum:

- Memperbaiki masalah yang dapat menyebabkan pernyataan SQL mengalami kesalahan atau peringatan pelanggaran kunci primer yang tidak terduga pada beberapa baris saat melakukan `INSERT` pernyataan bersamaan pada tabel yang memiliki kolom kunci `AUTO_INCREMENT` utama

dan kolom kunci unik, dan ketika `INSERT` pernyataan memiliki pelanggaran kunci unik pada baris yang berbeda.

- Memperbaiki masalah yang dapat menyebabkan hasil kueri yang salah ketika ZDR salah mengembalikan variabel sesi yang ditetapkan sebagai petunjuk dalam kueri.
- Memperbaiki masalah dalam kueri paralel yang menyebabkan set hasil yang tidak lengkap dikembalikan saat menggunakan fungsi bawaan `LPAD` dan `RPAD` string.
- Memperbaiki masalah yang menyebabkan hilangnya indeks kunci asing pada instance DB pembaca saat `ALTER TABLE RENAME COLUMN` pernyataan dijalankan pada instance DB penulis terhadap tabel dengan kunci asing.
- Memperbaiki masalah yang dapat menyebabkan kegagalan dalam menyelesaikan proses menonaktifkan penerusan tulis.
- Memperbaiki masalah selama Aurora Serverless v1 penskalaan yang menyebabkan instans DB dimulai ulang karena akses yang salah ke struktur data internal saat menemukan titik penskalaan.
- Memperbaiki masalah saat Skema Kinerja tidak diaktifkan saat manajemen otomatis Performance Insights diaktifkan untuk instans DB db.t4g.medium dan db.t4g.large.
- Batas waktu permintaan untuk operasi pembelajaran mesin Aurora ke SageMaker Amazon AI telah ditingkatkan dari 3 menjadi 30 detik. Ini membantu menyelesaikan masalah di mana pelanggan mungkin melihat peningkatan jumlah percobaan ulang atau kegagalan untuk permintaan ke Amazon SageMaker AI dari pembelajaran mesin Aurora saat menggunakan ukuran batch yang lebih besar.
- Memperbaiki masalah di mana lambat`INSERT`,`DELETE`, dan `UPDATE` kueri yang dijalankan oleh Penjadwal Acara MySQL tidak direkam dalam log kueri lambat kecuali didahului oleh kueri lambat. `SELECT`

Integrasi perbaikan bug MySQL Community Edition

Rilis ini mencakup semua perbaikan bug komunitas hingga dan termasuk 5.7.44. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 2.x.

- Memperbaiki masalah saat tabel sementara terikat pada pemicu saat menjalankan pernyataan dapat menyebabkan restart mesin DB yang tidak terduga.
- Memperbaiki cacat yang dapat menyebabkan server keluar ketika tabel tunggal `UPDATE` dan `DELETE` pernyataan menggunakan ekspresi yang diindeks dijalankan sebagai pernyataan yang disiapkan. (Bug #29257254)

Fitur yang tidak didukung di Aurora MySQL versi 2

Fitur-fitur berikut saat ini tidak didukung di Aurora MySQL versi 2 (kompatibel dengan MySQL 5.7).

- Pindai batching

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Pernyataan SQL `CREATE TABLESPACE`
- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Protokol X

Pembaruan mesin database Aurora MySQL 2024-03-19 (versi 2.12.2, kompatibel dengan MySQL 5.7.44) - Versi Dukungan Diperpanjang RDS

Versi: 2.12.2

Aurora MySQL 2.12.2 umumnya tersedia. Versi Aurora MySQL 2.12 kompatibel hingga MySQL 5.7.44. Untuk informasi selengkapnya tentang perubahan komunitas, lihat Perubahan di MySQL 5.7.44 (2022-10-11, Ketersediaan Umum).

Rilis Aurora MySQL yang didukung saat ini adalah 2.11.*, 2.12.*, 3.03.*, 3.04.*, 3.05.* dan 3.06.*.

Anda dapat meningkatkan cluster database Aurora MySQL 2.* yang ada ke Aurora MySQL 2.12.2. Anda juga dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.12.2.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Memperbaiki masalah keamanan dan CVEs tercantum di bawah ini:

Rilis ini mencakup semua perbaikan CVE komunitas hingga dan termasuk MySQL 5.7.44. Perbaikan CVE berikut disertakan:

- CVE-2024-20963
- CVE-2023-39975
- CVE-2023-38545

Masalah keamanan:

- Menambahkan perbaikan yang memastikan replika log biner default untuk digunakan SSL/TLS jika sumber mendukung koneksi terenkripsi, terlepas dari pengaturannya. `MASTER_SSL`

Perbaikan ketersediaan:

- Memperbaiki masalah yang dapat mencegah instance replika baca berhasil diluncurkan jika ada beban kerja yang tinggi pada instance penulis.
- Memperbaiki masalah yang dapat menyebabkan instance penulis database Aurora MySQL gagal karena cacat pada komponen yang berkomunikasi dengan penyimpanan Aurora. Cacat terjadi sebagai akibat dari gangguan dalam komunikasi antara instance database dan penyimpanan yang mendasarinya setelah pembaruan perangkat lunak dari instance penyimpanan Aurora.
- Memperbaiki masalah yang, dalam kondisi langka, dapat menyebabkan instance pembaca dimulai ulang.
- Memperbaiki masalah di mana pengguna yang memiliki hak istimewa dapat mengubah batas sumber daya yang terkait dengan pengguna, rdsadmin. Ketika disetel secara tidak benar, batas sumber daya ini dapat menghambat kemampuan agen pemantauan RDS untuk memantau kesehatan instance database yang menyebabkan tidak tersedianya database.

Peningkatan dan migrasi:

- Memperbaiki masalah yang terjadi saat mencoba memulai replikasi log biner untuk klaster MySQL Aurora yang telah bermigrasi dari Amazon RDS MySQL 5.7 dan berisi prosedur tersimpan yang tidak didukung.
- Menonaktifkan penjadwal acara database selama upgrade versi utama ke Aurora MySQL versi 3. Pembaruan ini membantu menghindari perubahan apa pun pada database oleh eksekusi acara saat pemutakhiran versi utama sedang berlangsung.

Integrasi perbaikan bug MySQL Community Edition

Rilis ini mencakup semua perbaikan bug komunitas hingga dan termasuk 5.7.44. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 2.x.

Fitur yang tidak didukung di Aurora MySQL versi 2

Fitur-fitur berikut saat ini tidak didukung di Aurora MySQL versi 2 (kompatibel dengan MySQL 5.7).

- Pindai batching

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Pernyataan SQL `CREATE TABLESPACE`
- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Protokol X

Pembaruan mesin database Aurora MySQL 2023-12-28 (versi 2.12.1, kompatibel dengan MySQL 5.7.40) - Versi Dukungan Diperpanjang RDS

Versi: 2.12.1

Aurora MySQL 2.12.1 tersedia secara umum. Aurora MySQL 2.12 kompatibel hingga dengan MySQL 5.7.40. Untuk informasi selengkapnya tentang perubahan komunitas, lihat Perubahan di MySQL 5.7.40 (2022-10-11, Ketersediaan Umum).

Rilis Aurora MySQL yang didukung saat ini adalah 2.11.*, 2.12.*, 3.01.*, 3.02.*, 3.03.*, 3.04.*, dan 3.05.*.

Anda dapat meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.12.1. Anda juga dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.12.1.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Memperbaiki masalah keamanan dan CVEs tercantum di bawah ini:

Rilis ini mencakup semua CVEs perbaikan komunitas hingga dan termasuk MySQL 5.7.44.

Perbaikan dan penyempurnaan lain untuk penanganan fine-tune di lingkungan terkelola. Di bawah ini adalah beberapa perbaikan CVE tambahan:

- CVE-2023-38546
- CVE-2023-38545
- CVE-2023-22053
- CVE-2023-22028
- CVE-2023-22026
- CVE-2023-22015
- CVE-2022-24407
- CVE-2020-11105
- CVE-2020-11104
- Pemrosesan tetap token karakter tunggal oleh plugin pengurai Pencarian Teks Lengkap (FTS) (Bug #35432973)

- Memperbaiki masalah di mana peristiwa yang dilaporkan saat memproses rotasi log audit mungkin tidak ditulis ke log audit

Fitur baru:

- Menambahkan dukungan untuk replikasi log biner multi-threaded (binlog), di mana thread SQL pada replika binlog akan menerapkan peristiwa log biner secara paralel jika memungkinkan. Pelajari lebih lanjut tentang opsi konfigurasi untuk membantu menyempurnakan replikasi multithreaded Anda di Panduan Pengguna Aurora.

Perbaikan ketersediaan:

- Memperbaiki masalah di mana instans basis data MySQL Aurora yang menggunakan kueri paralel mungkin mengalami mulai ulang basis data saat menjalankan kueri paralel dalam jumlah besar secara bersamaan.
- Memperbaiki masalah dengan perselisihan kunci yang disebabkan oleh thread pencatatan log audit yang dapat mengarah ke batas waktu aplikasi klien dan penggunaan CPU yang tinggi.
- Memperbaiki masalah yang dapat menyebabkan instans basis data memulai ulang saat mencoba membaca halaman basis data milik tabel yang dihapus.
- Memperbaiki masalah yang dapat menyebabkan instans pembaca memulai ulang saat instans penulis meningkatkan volume basis data ke kelipatan 160GB.
- Memperbaiki masalah di pengelola kunci yang dapat menyebabkan restart atau failover, saat menangani komit dua fase dengan tingkat isolasi yang diatur ke `READ_COMMITED` atau `READ_UNCOMMITED` dan transaksi XA digunakan atau log biner (binlog) diaktifkan.
- Memperbaiki masalah yang dapat menyebabkan klaster basis data menjadi tidak tersedia jika instans penulis memulai ulang saat basis data membuat atau menjatuhkan pemicu pada tabel sistem internal.
- Memperbaiki masalah yang dapat menyebabkan instans basis data di-restart ketika jumlah koneksi basis data mendekati nilai yang ditetapkan oleh parameter `max_connections`.
- Memperbaiki masalah yang dapat menyebabkan instans pembaca Aurora memulai ulang saat menjalankan kueri Data Manipulation Language (DML) pada tabel yang berisi indeks teks lengkap.
- Penyisipan cepat tidak diaktifkan dalam versi MySQL Aurora ini, karena masalah yang dapat menyebabkan ketidakkonsistenan saat menjalankan kueri seperti,, dan. `INSERT INTO SELECT FROM` Untuk informasi selengkapnya tentang pengoptimalan sisipan cepat, lihat penyempurnaan kinerja Amazon Aurora MySQL.

Perbaikan umum:

- Memperbaiki masalah yang dapat menyebabkan kueri paralel gagal karena masalah jaringan sementara saat membaca data dari volume klaster Aurora.
- Memperbaiki masalah yang terkait dengan manajemen file log audit yang dapat menyebabkan file log tidak dapat diakses untuk diunduh atau dirotasi, dan dalam beberapa kasus meningkatkan pemanfaatan CPU.
- Memperbaiki masalah di mana instans replika baca kecil mungkin mengalami peningkatan lag replikasi setelah peningkatan dari versi di bawah 2.11.*
- Memperbaiki masalah yang dapat menyebabkan pesan log berlebihan saat berkonsultasi dengan tabel pemberian akses `procs_priv` untuk verifikasi permintaan yang melibatkan rutinitas tersimpan.
- Memperbaiki masalah manajemen memori yang dapat menyebabkan instans basis data menggunakan memori yang berlebihan saat menjalankan kueri menggunakan pengoptimalan sambungan hash.
- Memperbaiki masalah yang dapat menghasilkan nilai variabel `Threads_running` yang salah dalam tabel status global `information_schema` dan `performance_schema` saat menggunakan penerusan tulis.
- Memperbaiki masalah yang menyebabkan mulai ulang basis data saat menjalankan pernyataan `SELECT` dengan tabel yang dipartisi (dibuat dalam versi MySQL yang mendukung penangan partisi `ha_partition` lama) dan kueri paralel dipilih oleh perencana kueri.
- Memperbaiki masalah yang dapat mencegah koneksi klien baru dibuat ke basis data saat penerusan penulisan diaktifkan.
- Mengurangi lag replikasi log biner (binlog) ketika replika binlog Aurora MySQL menjalankan peristiwa `QUERY` yang ditulis ke file binlog sumber tanpa basis data default yang ditentukan oleh perintah `USE`.
- Memperbaiki masalah yang dapat menyebabkan `CommitLatency` CloudWatch metrik salah dilaporkan saat `innodb_flush_log_at_trx_commit` parameter tidak disetel ke 1.
- Memperbaiki masalah yang dapat menyebabkan koneksi basis data ditutup sebelum terbentuk. Masalah ini lebih cenderung memengaruhi instans basis data yang membuka dan menutup koneksi dengan laju tinggi.
- Memperbaiki masalah yang dapat menyebabkan restart database saat konsumen log biner (binlog) yang terhubung menggunakan server replikasi binlog duplikat. IDs

- Memperbaiki masalah yang dapat menyebabkan replikasi log biner multithreaded berhenti saat `replica_parallel_type` variabel disetel `LOGICAL_CLOCK` dan variabel diputar. `replica_preserve_commit_order` `ON` Masalah ini dapat terjadi ketika transaksi yang lebih besar dari 500 MB dijalankan pada sumbernya.

Integrasi perbaikan bug MySQL Community Edition

Rilis ini meliputi semua perbaikan bug komunitas hingga dan termasuk 5.7.40, selain hal-hal di bawah ini. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 2.x.

- Memperbaiki masalah yang dapat menyebabkan koneksi jarak jauh yang ada dan baru terhenti saat dijalankan bersamaan dengan pernyataan `SHOW PROCESSLIST` (Bug Komunitas #34857411)
- Replikasi: Beberapa peristiwa log biner tidak selalu ditangani dengan benar (Bug #34617506)

Fitur yang tidak didukung di Aurora MySQL versi 2

Fitur-fitur berikut saat ini tidak didukung di Aurora MySQL versi 2 (kompatibel dengan MySQL 5.7).

- Pindai batching

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Pernyataan SQL `CREATE TABLESPACE`
- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber

- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Protokol X

Pembaruan mesin database Aurora MySQL 2023-10-25 (versi 2.12.0.1, kompatibel dengan MySQL 5.7.40) - Versi Dukungan Diperpanjang RDS (Beta)

Versi: 2.12.0.1

Aurora MySQL 2.12.0.1 umumnya tersedia di wilayah berikut: AS Timur (Virginia N.), AS Timur (Ohio), AS Barat (California N.), AS Barat (Oregon), (AS-Timur), dan (AS-Barat). AWS GovCloud AWS GovCloud Ini adalah rilis awal khusus perbaikan keamanan. Semua perbaikan ini akan di-deploy secara lebih luas di semua Wilayah dengan rilis patch berikutnya, 2.12.1. Aurora MySQL versi 2.12 kompatibel dengan MySQL 5.7.40.

Rilis Aurora MySQL yang didukung saat ini adalah 2.07.*, 2.11.*, 2.12.*, 3.01.*, 3.02.*, 3.03.*, dan 3.04.*.

Anda dapat meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.12.0.1. Anda juga dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.12.0.1.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Memperbaiki masalah keamanan dan CVEs tercantum di bawah ini:

Rilis ini mencakup semua CVEs perbaikan komunitas hingga dan termasuk MySQL 5.7.40.

- CVE-2023-38545

Perbaikan ketersediaan:

- Penyisipan cepat tidak diaktifkan dalam versi MySQL Aurora ini, karena masalah yang dapat menyebabkan ketidakkonsistenan saat menjalankan kueri seperti,, dan. `INSERT INTO SELECT FROM` Untuk informasi selengkapnya tentang pengoptimalan sisipan cepat, lihat penyempurnaan kinerja Amazon Aurora MySQL.

Pembaruan mesin database Aurora MySQL 2023-07-25 (versi 2.12.0, kompatibel dengan MySQL 5.7.40) - Versi Dukungan Diperpanjang RDS

Versi: 2.12.0

Aurora MySQL 2.12.0 tersedia secara umum. Aurora MySQL 2.12 kompatibel hingga dengan MySQL 5.7.40. Untuk informasi selengkapnya tentang perubahan komunitas, lihat Perubahan di MySQL 5.7.40 (2022-10-11, Ketersediaan Umum).

Rilis Aurora MySQL yang didukung saat ini adalah 2.11.*, 2.12.*, 3.01.*, 3.02.* dan 3.03.*.

Anda dapat meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.12.0. Anda juga dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.12.0.

Jika Anda memiliki pertanyaan atau masalah,AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Memperbaiki masalah keamanan dan CVEs tercantum di bawah ini:

Rilis ini mencakup semua CVEs perbaikan komunitas hingga dan termasuk MySQL 5.7.40.

- Cipher SSL default yang digunakan oleh Aurora MySQL telah diperbarui untuk mengecualikan nilai CBC3 DES--SHA yang kurang aman dari parameter database SSL_CIPHER. Jika Anda mengalami masalah koneksi SSL karena penghapusan cipher DES- CBC3 -SHA, silakan gunakan cipher aman yang berlaku dari daftar berikut, Mengonfigurasi cipher suite untuk koneksi ke cluster DB MySQL Aurora. Informasi selengkapnya tentang Konfigurasi Cipher Koneksi klien MySQL dapat ditemukan di dokumentasi MySQL.
- CVE-2023-21963
- CVE-2023-21912
- CVE-2023-21840
- CVE-2023-0215
- CVE-2022-43551
- CVE-2022-37434
- CVE-2022-32221
- CVE-2021-36222
- CVE-2021-22926
- CVE-2021-2169

Perbaikan ketersediaan:

- Memperbaiki masalah dalam enkripsi peristiwa aliran aktivitas basis data yang dapat menyebabkan mulai ulang basis data
- Memperbaiki dua masalah yang dapat menyebabkan basis data gagal memulai ulang jika ini terjadi saat menjalankan kueri Data Definition Language (DDL)
- Memperbaiki masalah di mana lonjakan koneksi dapat menyebabkan peningkatan latensi kueri atau mulai ulang instans basis data
- Memperbaiki masalah yang, dalam kasus yang sangat jarang, dapat menyebabkan replika Aurora memulai ulang selama eksekusi serentak operasi pembaruan besar atau beban kerja Data Definition Language (DDL) pada instans penulis dan operasi baca pada set tabel yang sama di replika Aurora
- Memperbaiki masalah di mana lonjakan koneksi dapat menyebabkan proses pembuatan koneksi memakan waktu lebih lama untuk diselesaikan atau gagal dengan kesalahan batas waktu

- Memperbaiki masalah di mana rotasi log Audit Lanjutan dapat mengurangi memori yang dapat dilepas, yang dapat menyebabkan instans basis data memulai ulang
- Memperbaiki masalah yang dapat menyebabkan instans pembaca MySQL Aurora memulai ulang saat menjalankan kueri yang menggunakan rencana eksekusi kueri paralel Aurora
- Memperbaiki masalah yang dapat menyebabkan instans penulis memulai ulang saat menjalankan kueri `OPTIMIZE TABLE` pada tabel dengan indeks Pencarian Teks Lengkap (FTS)
- Memperbaiki masalah yang dapat menyebabkan instance penulis di AWS Wilayah utama database global Aurora dimulai ulang saat `SELECT FOR UPDATE` kueri dijalankan menggunakan penerusan penulisan global dari Wilayah sekunder database global Aurora
- Memperbaiki masalah yang dapat menyebabkan instance pembaca AWS Region sekunder database global Aurora menggunakan penerusan tulis global untuk memulai ulang ketika pernyataan komit implisit yang diteruskan mengalami kesalahan
- Penyisipan cepat tidak diaktifkan dalam versi MySQL Aurora ini, karena masalah yang dapat menyebabkan ketidakkonsistenan saat menjalankan kueri seperti,, dan. `INSERT INTO SELECT FROM` Untuk informasi selengkapnya tentang pengoptimalan sisipan cepat, lihat penyempurnaan kinerja Amazon Aurora MySQL.

Perbaikan umum:

- Memperkenalkan pengoptimalan kinerja manajemen file pada replika binlog untuk membantu mengurangi perselisihan saat menulis ke file log relay
- Memperbaiki masalah yang dapat menyebabkan kesalahan pelaporan penghitung `buffer_pool_read_requests` dalam metrik `information_schema`
- Memperbaiki masalah yang dapat menyebabkan penyimpanan lokal terisi penuh saat menjalankan operasi `LOAD FROM S3` atau `SELECT INTO S3`. Masalah ini juga dapat menyebabkan penggunaan CPU yang lebih tinggi, mulai ulang basis data karena memori rendah, dan peningkatan latensi untuk kueri-kueri ini.
- Memperbaiki masalah di mana instans basis data yang menggunakan replikasi log biner mungkin mengalami peningkatan penggunaan CPU dan kegagalan koneksi ketika beberapa konsumen replikasi log biner dilampirkan
- Memperbaiki masalah di mana variabel status server SSL tidak diisi
- Memperbaiki masalah di mana pernyataan Data Manipulation Language (DHTML) yang mengeksekusi penulisan duplikat dapat menyebabkan pencatatan log kesalahan yang berlebihan dan peningkatan latensi kueri

- Meningkatkan definisi zona waktu ke versi IANA 2023c
- Menambahkan dukungan untuk mengaktifkan dan menonaktifkan pencatatan log biner tingkat sesi. Lihat Prosedur Tersimpan - Mereplikasi di Panduan Pengguna Amazon Aurora
- Menambahkan dukungan untuk pengaturan format log biner tingkat sesi. Lihat Prosedur Tersimpan - Mereplikasi di Panduan Pengguna Amazon Aurora
- Memperbaiki masalah di mana jika parameter `aurora_disable_hash_join` diatur ke `1` atau `ON`, pengoptimal mungkin tetap dapat menggunakan sambungan hash
- Memperbaiki masalah yang melibatkan pemindaian indeks di mana hasil yang tidak akurat mungkin dikembalikan saat menjalankan kueri `SELECT` dengan klausa `GROUP BY` dan parameter `aurora_parallel_query` diatur ke `ON`
- Memperbaiki masalah yang, dalam kasus yang sangat jarang, dapat menyebabkan instans pembaca Amazon Aurora memulai ulang saat mengakses tabel yang memiliki pembaruan besar atau operasi Data Definition Language (DDL) yang berjalan secara bersamaan pada instans penulis
- Memperbaiki masalah yang dapat menyebabkan kesalahan pelaporan penghitung `buffer_pool_read_requests` dalam metrik `information_schema`
- Memperbaiki masalah yang dapat menyebabkan replika binlog memulai ulang jika variabel sistem server uuid sumber hilang atau memiliki nilai yang tidak valid
- Memperbaiki masalah untuk mencegah statistik InnoDB menjadi usang, yang terkadang dapat menghasilkan rencana eksekusi kueri yang kurang optimal yang dapat menyebabkan peningkatan waktu eksekusi kueri
- Memperbaiki masalah dimana `AuroraGlobalDBRPOLag` CloudWatch metrik selalu ditampilkan nol terlepas dari beban kerja pengguna

Peningkatan dan migrasi:

- Untuk melakukan peningkatan versi kecil untuk basis data global Aurora dari Aurora MySQL versi 2.07 atau 2.11 ke Aurora MySQL versi 2.12 atau yang lebih tinggi, lihat Meningkatkan versi Aurora MySQL dengan memodifikasi versi mesin.

Integrasi perbaikan bug MySQL Community Edition

Rilis ini meliputi semua perbaikan bug komunitas hingga dan termasuk 5.7.40, selain hal-hal di bawah ini. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 2.x.

- Memperbaiki masalah yang dapat menyebabkan penggunaan CPU yang lebih tinggi karena rotasi sertifikat TLS di latar belakang (Perbaikan Bug Komunitas #34284186)

Fitur yang tidak didukung di Aurora MySQL versi 2

Fitur-fitur berikut saat ini tidak didukung di Aurora MySQL versi 2 (kompatibel dengan MySQL 5.7).

- Pindai batching.

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Pernyataan SQL `CREATE TABLESPACE`
- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Protokol X

Pembaruan mesin database Aurora MySQL 2024-07-19 (versi 2.11.6, kompatibel dengan MySQL 5.7.12) - Versi Dukungan Diperpanjang RDS

Versi: 2.11.6

Aurora MySQL 2.11.6 umumnya tersedia. Aurora MySQL versi 2.11 kompatibel dengan MySQL 5.7.12. Untuk informasi selengkapnya tentang perubahan komunitas, lihat Perubahan di MySQL 5.7.12 (2016-04-11, Ketersediaan Umum).

Rilis Aurora MySQL yang didukung saat ini adalah 2.07.9, 2.07.10, 2.11.*, 2.12.*, 3.01.*, 3.02.*, 3.03.*, 3.04.*, 3.05.*, 3.06.*, dan 3.07.*.

Anda dapat meningkatkan cluster database Aurora MySQL 2.* yang ada ke Aurora MySQL 2.11.6. Anda juga dapat memulihkan snapshot dari rilis Aurora MySQL versi 2 yang didukung saat ini ke Aurora MySQL 2.11.6.

Jika Anda meningkatkan basis data global Aurora MySQL ke versi 2.11.*, Anda harus meningkatkan klaster DB primer dan sekunder Anda ke versi yang sama persis, termasuk tingkat patch-nya. Untuk informasi selengkapnya tentang peningkatan versi kecil basis data global Aurora, lihat Peningkatan versi kecil.

Segera setelah peningkatan versi mesin secara in-place ke Aurora MySQL 2.11.* dilakukan, peningkatan sistem operasi diterapkan secara otomatis ke semua instans yang terpengaruh pada kelas instans DB db.r4, db.r5, db.t2, dan db.t3, jika instans-instans tersebut menjalankan versi sistem operasi lama. Dalam klaster DB Multi-AZ, semua instans pembaca menerapkan peningkatan sistem operasi terlebih dahulu. Ketika peningkatan sistem operasi pada instans pembaca pertama selesai, failover terjadi dan instans penulis sebelumnya ditingkatkan.

> **Note**
>
> Peningkatan sistem operasi tidak diterapkan secara otomatis ke basis data global Aurora selama peningkatan versi utama.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Memperbaiki masalah keamanan dan CVEs:

- Memperbaiki masalah keamanan dalam prosedur tersimpan MySQL.

Rilis ini mencakup semua perbaikan CVE komunitas hingga dan termasuk MySQL 5.7.12. Perbaikan CVE berikut disertakan dalam rilis ini:

- CVE-2023-21912
- CVE-2023-38545
- CVE-2023-39975
- CVE-2023-44487
- CVE-2024-0853
- CVE-2024-20963
- CVE-2024-20993
- CVE-2024-20998
- CVE-2024-21008
- CVE-2024-21009
- CVE-2024-21013
- CVE-2024-21047
- CVE-2024-21054
- CVE-2024-21055

- CVE-2024-21057
- CVE-2024-21062
- CVE-2024-21069
- CVE-2024-21096
- CVE-2024-21097

Perbaikan ketersediaan:

- Memperbaiki masalah yang dapat menyebabkan server database restart karena akses sumber daya koneksi secara bersamaan selama penskalaan mulus, restart zero-downtime (ZDR), dan zero-downtime patching (ZDP).
- Memperbaiki masalah yang dapat menyebabkan instans DB pembaca dimulai ulang saat membebaskan memori yang digunakan untuk aplikasi log.
- Memperbaiki masalah yang menyebabkan instans DB pembaca dimulai ulang saat menjalankan kueri paralel.
- Memperbaiki masalah yang dapat menyebabkan instans DB pembaca yang menggunakan penerusan tulis untuk memulai ulang saat pernyataan komit implisit yang diteruskan mengalami kesalahan.

Perbaikan umum:

- Memperbaiki masalah yang dapat menyebabkan pernyataan SQL mengalami kesalahan atau peringatan pelanggaran kunci primer yang tidak terduga pada beberapa baris saat melakukan `INSERT` pernyataan bersamaan pada tabel yang memiliki kolom kunci `AUTO_INCREMENT` utama dan kolom kunci unik, dan ketika `INSERT` pernyataan memiliki pelanggaran kunci unik pada baris yang berbeda.
- Memperbaiki masalah yang dapat menyebabkan hasil kueri salah saat ZDR salah mengembalikan variabel sesi yang ditetapkan sebagai petunjuk dalam kueri.
- Memperbaiki masalah selama Aurora Serverless v1 penskalaan yang menyebabkan instans DB dimulai ulang karena akses yang salah ke struktur data internal saat menemukan titik penskalaan.
- Memperbaiki masalah di mana lambat`INSERT`,`DELETE`, dan `UPDATE` kueri yang dijalankan oleh Penjadwal Acara MySQL tidak direkam dalam log kueri lambat kecuali didahului oleh kueri lambat. `SELECT`

Integrasi perbaikan bug MySQL Community Edition

Rilis ini mencakup semua perbaikan bug komunitas hingga dan termasuk 5.7.12. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 2.x.

Fitur yang tidak didukung di Aurora MySQL versi 2

Fitur-fitur berikut saat ini tidak didukung di Aurora MySQL versi 2 (kompatibel dengan MySQL 5.7).

- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin database Aurora MySQL 2024-03-26 (versi 2.11.5, kompatibel dengan MySQL 5.7.12) - Versi Dukungan Diperpanjang RDS (Default)

Versi: 2.11.5

Aurora MySQL 2.11.5 umumnya tersedia. Aurora MySQL versi 2.11 kompatibel dengan MySQL 5.7.12. Untuk informasi selengkapnya tentang perubahan komunitas, lihat Perubahan di MySQL 5.7.12 (2016-04-11, Ketersediaan Umum).

Versi 2.11.5 adalah versi default saat ini untuk Aurora MySQL versi 2 saat Anda membuat cluster DB. Untuk informasi selengkapnya, lihat Versi Amazon Aurora default.

Rilis Aurora MySQL yang didukung saat ini adalah 2.07.9, 2.07.10, 2.11.*, 2.12.*, 3.01.*, 3.02.*, 3.03.*, 3.04.*, 3.05.*, dan 3.06.*.

Anda dapat meningkatkan cluster database Aurora MySQL 2.* yang ada ke Aurora MySQL 2.11.5. Anda juga dapat memulihkan snapshot dari rilis Aurora MySQL versi 2 yang didukung saat ini ke Aurora MySQL 2.11.5.

Jika Anda meningkatkan basis data global Aurora MySQL ke versi 2.11.*, Anda harus meningkatkan klaster DB primer dan sekunder Anda ke versi yang sama persis, termasuk tingkat patch-nya. Untuk informasi selengkapnya tentang peningkatan versi kecil basis data global Aurora, lihat Peningkatan versi kecil.

Segera setelah peningkatan versi mesin secara in-place ke Aurora MySQL 2.11.* dilakukan, peningkatan sistem operasi diterapkan secara otomatis ke semua instans yang terpengaruh pada kelas instans DB db.r4, db.r5, db.t2, dan db.t3, jika instans-instans tersebut menjalankan versi sistem operasi lama. Dalam klaster DB Multi-AZ, semua instans pembaca menerapkan peningkatan sistem operasi terlebih dahulu. Ketika peningkatan sistem operasi pada instans pembaca pertama selesai, failover terjadi dan instans penulis sebelumnya ditingkatkan.

> **Note**
>
> Peningkatan sistem operasi tidak diterapkan secara otomatis ke basis data global Aurora selama peningkatan versi utama.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Memperbaiki masalah keamanan dan CVEs:

Perbaikan CVE berikut disertakan dalam rilis ini:

- CVE-2020-11104
- CVE-2020-11105
- CVE-2023-22015
- CVE-2023-22026
- CVE-2023-22028
- CVE-2023-22084
- CVE-2023-38545
- CVE-2023-38546
- CVE-2024-20963

Perbaikan ketersediaan:

- Memperbaiki masalah di mana instance DB penulis MySQL Aurora dapat gagal karena cacat pada komponen yang berkomunikasi dengan penyimpanan Aurora. Cacat terjadi sebagai akibat dari gangguan dalam komunikasi antara instans DB dan penyimpanan yang mendasarinya setelah pembaruan perangkat lunak.
- Memperbaiki masalah yang, dalam kondisi langka, dapat menyebabkan instans DB pembaca dimulai ulang.

- Memperbaiki masalah dengan perselisihan kunci yang disebabkan oleh thread pencatatan log audit yang dapat mengarah ke batas waktu aplikasi klien dan penggunaan CPU yang tinggi.

Perbaikan umum:

- Memperbaiki masalah yang dapat menyebabkan kueri paralel gagal karena masalah jaringan sementara saat membaca data dari volume cluster Aurora DB.
- Memperbaiki masalah yang terkait dengan manajemen file log audit yang dapat menyebabkan file log tidak dapat diakses untuk diunduh atau dirotasi, dan dalam beberapa kasus meningkatkan penggunaan CPU.
- Memperbaiki masalah yang dapat menghasilkan nilai `Threads_running` variabel yang salah dalam `information_schema` dan tabel status `performance_schema` global saat menggunakan penerusan tulis.

Peningkatan dan migrasi:

- Memperbaiki masalah yang mencegah inisiasi replikasi log biner pada cluster DB MySQL Aurora yang dimigrasi dari RDS untuk MySQL 5.7.
- Menonaktifkan penjadwal acara database selama upgrade versi utama ke Aurora MySQL versi 3. Ini membantu menghindari perubahan apa pun pada database dengan eksekusi acara saat pemutakhiran versi utama sedang berlangsung.

Integrasi perbaikan bug MySQL Community Edition

Rilis ini meliputi semua perbaikan bug komunitas hingga dan termasuk 5.7.12, selain hal-hal di bawah ini. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 2.x.

Fitur yang tidak didukung di Aurora MySQL versi 2

Fitur-fitur berikut saat ini tidak didukung di Aurora MySQL versi 2 (kompatibel dengan MySQL 5.7).

- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin database Aurora MySQL 2023-10-17 (versi 2.11.4, kompatibel dengan MySQL 5.7.12) - Versi Dukungan Diperpanjang RDS

Versi: 2.11.4

Aurora MySQL 2.11.4 tersedia secara umum. Aurora MySQL versi 2.11 kompatibel dengan MySQL 5.7.12. Untuk informasi selengkapnya tentang perubahan komunitas, lihat Perubahan di MySQL 5.7.12 (2016-04-11, Ketersediaan Umum).

Rilis Aurora MySQL yang didukung saat ini adalah 2.07.09, 207.10, 2.11.*, 2.12.*, 3.01.*, 3.02.*, 3.03.*, dan 3.04.*.

Anda dapat meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.11.4. Anda juga dapat memulihkan snapshot dari rilis Aurora MySQL yang didukung saat ini ke Aurora MySQL 2.11.4.

Jika Anda meningkatkan basis data global Aurora MySQL ke versi 2.11.*, Anda harus meningkatkan klaster DB primer dan sekunder Anda ke versi yang sama persis, termasuk tingkat patch-nya. Untuk informasi selengkapnya tentang peningkatan versi kecil basis data global Aurora, lihat Peningkatan versi kecil.

Segera setelah peningkatan versi mesin secara in-place ke Aurora MySQL 2.11.* dilakukan, peningkatan sistem operasi diterapkan secara otomatis ke semua instans yang terpengaruh pada kelas instans DB db.r4, db.r5, db.t2, dan db.t3, jika instans-instans tersebut menjalankan versi sistem operasi lama. Dalam klaster DB Multi-AZ, semua instans pembaca menerapkan peningkatan sistem operasi terlebih dahulu. Ketika peningkatan sistem operasi pada instans pembaca pertama selesai, failover terjadi dan instans penulis sebelumnya ditingkatkan.

> **Note**
>
> Peningkatan sistem operasi tidak diterapkan secara otomatis ke basis data global Aurora selama peningkatan versi utama.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Memperbaiki masalah keamanan dan CVEs tercantum di bawah ini:

- Memperbaiki masalah di mana peristiwa yang dilaporkan saat memproses rotasi log audit mungkin tidak ditulis ke log audit.
- CVE-2022-24407

Perbaikan ketersediaan:

- Memperbaiki masalah di mana instans basis data MySQL Aurora yang menggunakan kueri paralel mungkin mengalami mulai ulang basis data saat menjalankan kueri paralel dalam jumlah besar secara bersamaan.
- Memperbaiki masalah yang dapat menyebabkan instance database dimulai ulang saat menjalankan beban kerja baca I/O intensif.
- Memperbaiki masalah yang dapat menyebabkan instans basis data memulai ulang saat mencoba membaca halaman basis data milik tabel yang dihapus.
- Memperbaiki masalah yang dapat menyebabkan instans pembaca memulai ulang saat instans penulis meningkatkan volume basis data ke kelipatan 160GB.
- Memperbaiki masalah yang dapat menyebabkan klaster basis data menjadi tidak tersedia jika instans penulis memulai ulang saat basis data membuat atau menjatuhkan pemicu pada tabel sistem internal.
- Memperbaiki masalah yang dapat menyebabkan instans pembaca memulai ulang saat menjalankan kueri Data Manipulation Language (DML) pada tabel yang berisi indeks teks lengkap.
- Memperbaiki masalah yang dapat menyebabkan instans pembaca memulai ulang saat menjalankan kueri yang menggunakan rencana eksekusi kueri paralel Aurora.
- Memperbaiki masalah yang dapat menyebabkan instans penulis memulai ulang saat menjalankan kueri `OPTIMIZE TABLE` pada tabel dengan indeks Pencarian Teks Lengkap (FTS).
- Penyisipan cepat tidak diaktifkan dalam versi MySQL Aurora ini, karena masalah yang dapat menyebabkan ketidakkonsistenan saat menjalankan kueri seperti,, dan. `INSERT INTO SELECT FROM` Untuk informasi selengkapnya tentang pengoptimalan sisipan cepat, lihat penyempurnaan kinerja Amazon Aurora MySQL.

Perbaikan umum:

- Memperbaiki masalah di mana instans replika baca kecil mungkin mengalami peningkatan lag replikasi setelah peningkatan dari versi di bawah 2.11.*.
- Memperbaiki masalah yang dapat menyebabkan pesan log berlebihan saat berkonsultasi dengan tabel pemberian akses procs_priv untuk verifikasi permintaan yang melibatkan rutinitas tersimpan.
- Memperbaiki masalah manajemen memori yang dapat menyebabkan instans basis data menggunakan memori yang berlebihan saat menjalankan kueri menggunakan pengoptimalan sambungan hash.

- Memperbaiki masalah yang menyebabkan mulai ulang basis data saat menjalankan pernyataan `SELECT` dengan tabel yang dipartisi (dibuat dalam versi MySQL yang mendukung penangan partisi `ha_partition` lama) dan kueri paralel dipilih oleh perencana kueri.
- Memperbaiki masalah yang dapat mencegah koneksi klien baru dibuat ke basis data saat penerusan penulisan diaktifkan.
- Mengurangi lag replikasi log biner (binlog) ketika replika binlog Aurora MySQL menjalankan peristiwa `QUERY` yang ditulis ke file binlog sumber tanpa basis data default yang ditentukan oleh perintah `USE`.
- Memperbaiki masalah yang melibatkan pemindaian indeks di mana hasil yang tidak akurat mungkin dikembalikan saat menjalankan kueri `SELECT` dengan klausa `GROUP BY` dan parameter `aurora_parallel_query` diatur ke `ON`.
- Menambahkan dukungan untuk mengaktifkan dan menonaktifkan pencatatan log biner tingkat sesi. Lihat Prosedur Tersimpan - Mereplikasi di Panduan Pengguna Amazon Aurora.
- Memperbaiki masalah yang dapat menyebabkan replika binlog memulai ulang jika variabel sistem server_uuid sumber hilang atau memiliki nilai yang tidak valid.
- Menambahkan dukungan untuk pengaturan format log biner tingkat sesi. Lihat Prosedur Tersimpan - Mereplikasi di Panduan Pengguna Amazon Aurora.
- Memperbaiki masalah yang dapat menyebabkan `CommitLatency` CloudWatch metrik salah dilaporkan saat `innodb_flush_log_at_trx_commit` parameter tidak disetel ke 1.
- Memperbaiki masalah untuk mencegah statistik InnoDB menjadi usang, yang terkadang dapat menghasilkan rencana eksekusi kueri yang kurang optimal yang dapat menyebabkan peningkatan waktu eksekusi kueri.
- Memperbaiki masalah yang dapat menyebabkan restart database saat konsumen log biner (binlog) yang terhubung menggunakan server replikasi binlog duplikat. IDs

Integrasi perbaikan bug MySQL Community Edition

Rilis ini meliputi semua perbaikan bug komunitas hingga dan termasuk 5.7.12, selain hal-hal di bawah ini. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 2.x.

- Replikasi: Beberapa peristiwa log biner tidak selalu ditangani dengan benar. (Bug #34617506)
- Memperbaiki masalah yang dapat menyebabkan penggunaan CPU yang lebih tinggi karena rotasi sertifikat TLS di latar belakang (Perbaikan Bug Komunitas #34284186).

- Dalam pernyataan yang disiapkan, beberapa jenis subkueri dapat menyebabkan keluar server. (Bug #33100586)

Fitur yang tidak didukung di Aurora MySQL versi 2

Fitur-fitur berikut saat ini tidak didukung di Aurora MySQL versi 2 (kompatibel dengan MySQL 5.7).

- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin database Aurora MySQL 2023-06-09 (versi 2.11.3, kompatibel dengan MySQL 5.7.12) - Versi Dukungan Diperpanjang RDS

Versi: 2.11.3

Aurora MySQL 2.11.3 tersedia secara umum. Aurora MySQL versi 2.11 kompatibel dengan MySQL 5.7.12. Untuk informasi selengkapnya tentang perubahan komunitas, lihat Perubahan di MySQL 5.7.12 (2016-04-11, Ketersediaan Umum).

Rilis Aurora MySQL yang didukung saat ini adalah 2.07.*, 2.11.*, 3.01.*, 3.02.*, dan 3.03.*.

Anda dapat meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.11.3. Anda juga dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.11.3.

Jika Anda meningkatkan basis data global Aurora MySQL ke versi 2.11.*, Anda harus meningkatkan klaster DB primer dan sekunder Anda ke versi yang sama persis, termasuk tingkat patch-nya. Untuk informasi selengkapnya tentang peningkatan versi kecil basis data global Aurora, lihat Peningkatan versi kecil.

Segera setelah peningkatan versi mesin secara in-place ke Aurora MySQL 2.11.* dilakukan, peningkatan sistem operasi diterapkan secara otomatis ke semua instans yang terpengaruh pada kelas instans DB db.r4, db.r5, db.t2, dan db.t3, jika instans-instans tersebut menjalankan versi sistem operasi lama. Dalam klaster DB Multi-AZ, semua instans pembaca menerapkan peningkatan sistem operasi terlebih dahulu. Ketika peningkatan sistem operasi pada instans pembaca pertama selesai, failover terjadi dan instans penulis sebelumnya ditingkatkan.

> **Note**
>
> Peningkatan sistem operasi tidak diterapkan secara otomatis ke basis data global Aurora selama peningkatan versi utama.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Memperbaiki masalah keamanan dan CVEs tercantum di bawah ini:

- Memperbarui cipher SSL default yang digunakan oleh Aurora MySQL untuk mengecualikan nilai `DES-CBC3-SHA` yang kurang aman dari parameter basis data SSL_CIPHER. Jika Anda mengalami masalah koneksi SSL karena penghapusan sandi, gunakan `DES-CBC3-SHA` sandi aman yang berlaku dari daftar ini,. ConfiguringCipherSuites Informasi selengkapnya tentang Konfigurasi Cipher Koneksi klien MySQL dapat ditemukan di dokumentasi MySQL.
- CVE-2023-21963
- CVE-2023-21912
- CVE-2023-0215
- CVE-2022-43551
- CVE-2022-37434

Perbaikan ketersediaan:

- Memperbaiki masalah di enkripsi peristiwa aliran aktivitas basis data (DAS) yang dapat menyebabkan mulai ulang basis data.
- Memperbaiki dua masalah yang dapat menyebabkan basis data gagal memulai ulang jika ini terjadi saat menjalankan kueri Data Definition Language (DDL).
- Penyisipan cepat tidak diaktifkan dalam versi MySQL Aurora ini, karena masalah yang dapat menyebabkan ketidakkonsistenan saat menjalankan kueri seperti,, dan. `INSERT INTO SELECT FROM` Untuk informasi selengkapnya tentang pengoptimalan sisipan cepat, lihat penyempurnaan kinerja Amazon Aurora MySQL.

Perbaikan umum:

- Memperkenalkan pengoptimalan kinerja manajemen file pada replika binlog untuk membantu mengurangi perselisihan saat menulis ke file log relay.
- Memperbaiki masalah di mana jika parameter `aurora_disable_hash_join` diatur ke 1 atau ON, pengoptimal mungkin tetap dapat menggunakan sambungan hash.
- Memperbaiki masalah yang dapat menyebabkan kesalahan pelaporan penghitung `buffer_pool_read_requests` dalam metrik `information_schema`.

- Memperbaiki masalah yang dapat menyebabkan penyimpanan lokal terisi penuh saat menjalankan operasi `LOAD FROM S3` atau `SELECT INTO S3`. Masalah ini juga dapat menyebabkan penggunaan CPU yang lebih tinggi, mulai ulang basis data karena memori rendah, dan peningkatan latensi untuk kueri-kueri ini.

Fitur yang tidak didukung di Aurora MySQL versi 2

Fitur-fitur berikut saat ini tidak didukung di Aurora MySQL versi 2 (kompatibel dengan MySQL 5.7).

- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin database Aurora MySQL 2023-03-24 (versi 2.11.2, kompatibel dengan MySQL 5.7.12) - Versi Dukungan Diperpanjang RDS

Versi: 2.11.2

Aurora MySQL 2.11.2 tersedia secara umum. Aurora MySQL versi 2.11 kompatibel dengan MySQL 5.7.12. Untuk informasi selengkapnya tentang perubahan komunitas, lihat Perubahan di MySQL 5.7.12 (2016-04-11, Ketersediaan Umum).

Rilis Aurora MySQL yang didukung saat ini adalah 2.07.*, 2.11.*, 3.01.*, 3.02.*, dan 3.03.*.

Anda dapat meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.11.2. Anda juga dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.11.2.

Jika Anda meningkatkan basis data global Aurora MySQL ke versi 2.11.*, Anda harus meningkatkan klaster DB primer dan sekunder Anda ke versi yang sama persis, termasuk tingkat patch-nya. Untuk informasi selengkapnya tentang peningkatan versi kecil basis data global Aurora, lihat Peningkatan versi kecil.

Segera setelah peningkatan versi mesin secara in-place ke Aurora MySQL 2.11.* dilakukan, peningkatan sistem operasi diterapkan secara otomatis ke semua instans yang terpengaruh pada kelas instans DB db.r4, db.r5, db.t2, dan db.t3, jika instans-instans tersebut menjalankan versi sistem operasi lama. Dalam klaster DB Multi-AZ, semua instans pembaca menerapkan peningkatan sistem operasi terlebih dahulu. Ketika peningkatan sistem operasi pada instans pembaca pertama selesai, failover terjadi dan instans penulis sebelumnya ditingkatkan.

> **Note**
>
> Peningkatan sistem operasi tidak diterapkan secara otomatis ke basis data global Aurora selama peningkatan versi utama.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan umum:

- Memperbaiki masalah di mana instans basis data yang menggunakan replikasi log biner mungkin mengalami peningkatan penggunaan CPU dan kegagalan koneksi ketika beberapa konsumen replikasi log biner dilampirkan.
- Memperbaiki masalah yang dapat menyebabkan ketidaksinkronan instans pembaca di Wilayah sekunder basis data global setelah peningkatan ke Aurora MySQL versi 2.11 jika penulis basis data utama menggunakan Aurora MySQL versi 2.10.

Perbaikan ketersediaan:

- Penyisipan cepat tidak diaktifkan dalam versi MySQL Aurora ini, karena masalah yang dapat menyebabkan ketidakkonsistenan saat menjalankan kueri seperti,, dan. `INSERT INTO SELECT FROM` Untuk informasi selengkapnya tentang pengoptimalan sisipan cepat, lihat penyempurnaan kinerja Amazon Aurora MySQL.

Fitur yang tidak didukung di Aurora MySQL versi 2

Fitur-fitur berikut saat ini tidak didukung di Aurora MySQL versi 2 (kompatibel dengan MySQL 5.7).

- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan

implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin database Aurora MySQL 2023-02-14 (versi 2.11.1, kompatibel dengan MySQL 5.7.12) - Versi Dukungan Diperpanjang RDS

Versi: 2.11.1

Aurora MySQL 2.11.1 tersedia secara umum. Aurora MySQL versi 2.11 kompatibel dengan MySQL 5.7.12. Untuk informasi selengkapnya tentang perubahan komunitas, lihat Perubahan di MySQL 5.7.12 (2016-04-11, Ketersediaan Umum).

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.07.*, 2.09.*, 2.10.*, 2.11.*, 3.01.*, dan 3.02.*.

Anda dapat meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.11.1. Untuk klaster yang menjalankan Aurora MySQL versi 1, Anda dapat meningkatkan klaster Aurora MySQL 1.23 atau yang lebih tinggi yang sudah ada langsung ke 2.11.1. Anda juga dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.11.1.

Jika Anda meningkatkan basis data global Aurora MySQL ke versi 2.11.* dan Anda mengaktifkan penerusan tulis, Anda harus meningkatkan klaster DB primer dan sekunder Anda ke versi yang sama

persis, termasuk tingkat patch-nya, untuk melanjutkan penggunaan penerusan tulis. Untuk informasi selengkapnya tentang peningkatan versi kecil basis data global Aurora, lihat Peningkatan versi kecil.

Segera setelah peningkatan versi mesin secara in-place ke Aurora MySQL 2.11.* dilakukan, peningkatan sistem operasi diterapkan secara otomatis ke semua instans yang terpengaruh pada kelas instans DB db.r4, db.r5, db.t2, dan db.t3, jika instans-instans tersebut menjalankan versi sistem operasi lama. Dalam klaster DB Multi-AZ, semua instans pembaca menerapkan peningkatan sistem operasi terlebih dahulu. Ketika peningkatan sistem operasi pada instans pembaca pertama selesai, failover terjadi dan instans penulis sebelumnya ditingkatkan.

> **Note**
>
> Peningkatan sistem operasi tidak diterapkan secara otomatis ke basis data global Aurora selama peningkatan versi utama.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Memperbaiki masalah keamanan dan CVEs tercantum di bawah ini:

Perbaikan dan penyempurnaan lain untuk penanganan fine-tune di lingkungan terkelola. Di bawah ini adalah beberapa perbaikan CVE tambahan:

- CVE-2022-32221
- CVE-2021-36222
- CVE-2021-22926
- CVE-2021-2169

Perbaikan ketersediaan:

- Memperbaiki masalah di mana lonjakan koneksi dapat menyebabkan peningkatan latensi kueri atau mulai ulang instans basis data.
- Memperbaiki masalah yang, dalam kasus yang sangat jarang, dapat menyebabkan replika Aurora memulai ulang selama eksekusi serentak operasi pembaruan besar atau beban kerja Data Definition Language (DDL) pada instans penulis dan operasi baca pada set tabel yang sama di replika Aurora.
- Memperbaiki masalah di mana lonjakan koneksi dapat menyebabkan proses pembuatan koneksi memakan waktu lebih lama untuk diselesaikan atau gagal dengan kesalahan batas waktu.
- Memperbaiki masalah di mana rotasi log Audit Lanjutan dapat mengurangi memori yang dapat dilepas, yang dapat menyebabkan instans basis data memulai ulang.
- Penyisipan cepat tidak diaktifkan dalam versi MySQL Aurora ini, karena masalah yang dapat menyebabkan ketidakkonsistenan saat menjalankan kueri seperti,, dan. `INSERT INTO SELECT FROM` Untuk informasi selengkapnya tentang pengoptimalan sisipan cepat, lihat penyempurnaan kinerja Amazon Aurora MySQL.

Perbaikan umum:

- Memperbaiki masalah di mana variabel status server SSL tidak diisi.
- Memperbaiki masalah di mana pernyataan Data Manipulation Language (DHTML) yang mengeksekusi penulisan duplikat dapat menyebabkan pencatatan log kesalahan yang berlebihan dan peningkatan latensi kueri.

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL versi 2 (kompatibel dengan MySQL 5.7).

- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.

- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin database Aurora MySQL 2022-10-25 (versi 2.11.0, kompatibel dengan MySQL 5.7.12) - Versi Dukungan Diperpanjang RDS

Versi: 2.11.0

Aurora MySQL 2.11.0 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7.12. Untuk informasi selengkapnya tentang perubahan komunitas, lihat Perubahan di MySQL 5.7.12 (2016-04-11, Ketersediaan Umum).

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 2.11.*, 3.01.*, dan 3.02.*.

Anda dapat meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.11.0. Untuk klaster yang menjalankan Aurora MySQL versi 1, Anda dapat meningkatkan klaster Aurora MySQL 1.23 atau yang lebih tinggi yang sudah ada langsung ke 2.11.0. Anda juga dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.11.0.

Jika Anda meningkatkan basis data global Aurora MySQL ke versi 2.11.* dan Anda mengaktifkan penerusan tulis, Anda harus meningkatkan klaster DB primer dan sekunder Anda ke versi yang sama persis, termasuk tingkat patch-nya, untuk melanjutkan penggunaan penerusan tulis. Untuk informasi selengkapnya tentang peningkatan versi kecil basis data global Aurora, lihat Peningkatan versi kecil.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Memperbaiki masalah keamanan dan CVEs:

- Mengubah SSL/TLS penyedia dari OpenSSL menjadi. AWS-LC

  Ini membawa sejumlah perubahan termasuk, namun tidak terbatas pada, penghapusan dukungan untuk cipher DHE-RSA-* yang kurang aman.

  Untuk informasi selengkapnya, lihat Menggunakan klaster DB Aurora MySQL

Perbaikan CVE berikut disertakan dalam rilis ini:

- CVE-2022-21460
- CVE-2022-21451
- CVE-2022-21444
- CVE-2022-21417
- CVE-2022-21304
- CVE-2022-21303
- CVE-2022-21245
- CVE-2021-36222
- CVE-2021-28196
- CVE-2021-23841
- CVE-2021-22926
- CVE-2021-3449
- CVE-2021-2307
- CVE-2021-2226
- CVE-2021-2202
- CVE-2021-2194
- CVE-2021-2179
- CVE-2021-2178
- CVE-2021-2174
- CVE-2021-2171
- CVE-2021-2169
- CVE-2021-2166
- CVE-2021-2160
- CVE-2021-2154

Fitur baru:

- Dengan rilis Aurora MySQL versi 2.11, peningkatan sistem operasi baru menjadi tersedia. Kami menyarankan Anda menerapkan pembaruan OS yang tertunda ini ke semua instans basis data Aurora MySQL Anda setelah peningkatan ke versi 2.11. Untuk informasi selengkapnya, lihat Bekerja dengan pembaruan sistem operasi.

- Opsi konfigurasi dinamis baru, innodb_deadlock_detect, dapat digunakan untuk menonaktifkan deteksi deadlock. Pada sistem konkurensi tinggi, deteksi deadlock dapat menyebabkan perlambatan ketika banyak thread yang menunggu kunci yang sama. Kadang-kadang, mungkin lebih efisien untuk menonaktifkan deteksi deadlock dan mengandalkan pengaturan innodb_lock_wait_timeout untuk rollback transaksi ketika terjadi deadlock. (Bug #23477773) Informasi selengkapnya tentang deteksi deadlock Innodb dapat ditemukan di dokumentasi MySQL.
- Fungsi `UUID_TO_BIN`, `BIN_TO_UUID`, dan `IS_UUID` dari MySQL 8.0 telah ditambahkan. Informasi selengkapnya tentang penggunaan fungsi-fungsi ini dapat ditemukan di Fungsi Lain-lain MySQL.
- Menambahkan dukungan untuk petunjuk pengoptimal yang memungkinkan pengguna untuk mengaktifkan atau menonaktifkan kueri paralel Aurora MySQL berdasarkan tabel atau kueri.
  - Bekerja dengan kueri paralel untuk Amazon Aurora MySQL
  - Petunjuk Aurora MySQL
- Menghapus dukungan tipe instans R3.
- Menambahkan dukungan untuk instans R6i.

Perbaikan ketersediaan:

- Memperbaiki masalah yang dapat mencegah replikasi logis lintas wilayah dalam klaster basis data karena kesalahan penulisan nama dan lokasi file binlog ke log kesalahan. Masalah ini dapat terjadi ketika mesin dimulai ulang setelah menjalankan pernyataan DDL.
- Memperbaiki masalah yang, dalam kondisi yang sangat jarang, dapat menyebabkan instans pembaca Aurora memulai ulang saat menjalankan pernyataan Daftar Kontrol Akses (ACL) seperti GRANT dan FLUSH pada instans penulis. Masalah ini kemungkinan besar memengaruhi instans pembaca dengan pengguna dan operasi ACL dalam jumlah besar (misalnya, perubahan izin).
- Memperbaiki masalah yang, dalam kondisi yang sangat jarang, dapat menyebabkan instans penulis memulai ulang atau melakukan failover ketika transaksi mengakses baris yang dihapus oleh transaksi lain.
- Memperbaiki kinerja pencarian frasa Teks Penuh untuk secara signifikan mengurangi waktu yang dibutuhkan untuk mencari frasa dalam suatu tabel dengan indeks teks penuh.
- Memperbaiki masalah di mana, setelah instans penulis memulai ulang, instans tersebut terhenti dalam pemulihan yang lambat dan kemudian memulai ulang lagi. Masalah ini terjadi ketika terdapat baris dalam jumlah besar yang tidak di-commit dalam basis data pada saat mulai ulang awal.

- Memperbaiki masalah yang, dalam kasus yang sangat jarang, menyebabkan server basis data memulai ulang karena menunggu semaphore yang lama ketika thread pendeteksi deadlock macet.
- Memperbaiki masalah yang, dalam kasus yang jarang terjadi, dapat menyebabkan database dimulai ulang karena menunggu semaphore yang lama ketika I/O utas menemui jalan buntu.
- Penyisipan cepat tidak diaktifkan dalam versi MySQL Aurora ini, karena masalah yang dapat menyebabkan ketidakkonsistenan saat menjalankan kueri seperti,, dan. `INSERT INTO SELECT FROM` Untuk informasi selengkapnya tentang pengoptimalan sisipan cepat, lihat penyempurnaan kinerja Amazon Aurora MySQL.

Perbaikan umum:

- Memperbaiki masalah yang dapat menyebabkan server basis data memulai ulang ketika semua kondisi berikut berlaku:
  - ALLOW_INVALID_DATES dinonaktifkan dalam SQL MODE.
  - Server basis data memproses pernyataan INSERT, UPDATE, DELETE, atau SELECT dengan nilai tipe DATETIME yang tidak valid sehingga bulannya tidak antara 1 dan 12.
- Memperbaiki masalah di mana periode retensi log biner tidak dipatuhi saat log-bin disetel ke OFF, yang menyebabkan penggunaan penyimpanan yang lebih tinggi dari yang diharapkan. Setelah perbaikan ini, log biner akan dibersihkan berdasarkan periode retensi Anda. Informasi selengkapnya tentang cara mengonfigurasi periode retensi log biner Anda dapat ditemukan di Panduan Pengguna MySQL Aurora.
- Memperbaiki masalah yang dapat menyebabkan memori yang dapat dilepas pada instans basis data menjadi berkurang ketika pernyataan SQL Data Control Language (DCL) tertentu seperti GRANT, FLUSH PRIVILEGES dll. dijalankan pada instans tersebut. Sering menggunakan pernyataan tersebut dapat menyebabkan memori yang dapat dibebaskan terus berkurang dan dapat menyebabkan instance database dimulai ulang karena out-of-memory masalah. Penggunaan pernyataan tersebut pada instans penulis juga dapat menyebabkan memori yang dapat dilepas pada instans pembaca berkurang.
- Memperkenalkan ukuran buffer baca yang lebih besar untuk pembacaan yang dilakukan dari log relai untuk meminimalkan jumlah I/O operasi baca, yang mengurangi pertengkaran antara thread I/O dan SQL.
- Memperbaiki masalah yang dapat menyebabkan kegagalan prosedur tersimpan mysql.rds_rotate_slow_log dengan pesan kesalahan “Table 'mysql.slow_log_backup' doesn’t exist”.

- Memperbaiki masalah di mana pembatalan cache kueri yang berlebihan menyebabkan penggunaan CPU dan latensi yang lebih tinggi dari yang diharapkan pada replika baca karena replika baca harus membaca data dari disk, bukan dari cache kueri.
- Memperbaiki masalah yang memungkinkan pengguna menjalankan perintah INSTALL PLUGIN dan UNINSTALL PLUGIN pada instans pembaca, yang dapat menyebabkan deadlock pada LOCK_plugin, LOCK_system_variables_hash, LOCK_global_system_variables. Pernyataan-pernyataan ini sekarang hanya dapat dieksekusi pada instans penulis dalam klaster basis data.
- Memperbaiki masalah ketika klaster mungkin mengalami latensi commit yang lebih tinggi dari yang diharapkan saat log biner diaktifkan. Hal ini memengaruhi semua transaksi yang menghasilkan peristiwa binlog besar (berukuran lebih dari 500MB).
- Memperbaiki masalah yang dapat menyebabkan metrik trx_active_transactions pada tabel INFORMATION_SCHEMA.INNODB_METRICS memiliki nilai yang salah.
- Memperbaiki masalah yang dapat menghentikan replikasi logis karena file binlog menjadi tidak konsisten saat menjalankan rollback ke savepoint untuk transaksi besar.
- Hash kredenal bertopeng di log umum, slow-query-log, dan log audit secara default menggunakan rahasia topeng yang konsisten. Ini dapat dikonfigurasi melalui parameter aurora_mask_password_hashes_type.
- Memperbaiki masalah di mana durasi Zero-Downtime-Restart (ZDR) dilaporkan secara tidak benar dalam peristiwa yang diamati pelanggan.
- Memperbaiki masalah yang dapat menyebabkan kegagalan panggilan ke mysql_rds_import_binlog_ssl_material dengan KESALAHAN 1457 server MySQL.
- Memperbaiki masalah di mana inisialisasi thread dump dapat mengalami deadlock dengan thread untuk membersihkan log biner. Masalah ini dapat menghentikan file binlog aktif dari berotasi sehingga terus bertambah besar atau menyebabkan masalah dengan koneksi replika binlog baru.
- Memperbaiki masalah di mana cache kueri dapat mengembalikan hasil usang pada instans replika Aurora.

Integrasi perbaikan bug MySQL Community Edition

Rilis ini meliputi semua perbaikan bug komunitas hingga dan termasuk 5.7, selain hal-hal di bawah ini. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 2.x.

- Memperbaiki masalah di mana kode untuk membaca informasi kumpulan karakter dari tabel peristiwa pernyataan Skema Kinerja (misalnya, events_statements_current) tidak mencegah

penulisan serentak ke informasi kumpulan karakter tersebut. Akibatnya, kumpulan karakter teks kueri SQL bisa menjadi tidak valid, yang dapat mengakibatkan server keluar. Dengan perbaikan ini, kumpulan karakter yang tidak valid menyebabkan pemotongan kolom SQL_TEXT dan mencegah server keluar. (Bug #23540008)

- InnoDB: Backport perbaikan untuk Community Bug #25189192, Bug #84038. Memperbaiki masalah di mana operasi RENAME TABLE yang memindahkan tabel ke skema yang berbeda, InnoDB gagal memperbarui tabel kamus data INNODB_SYS_DATAFILES. Hal ini mengakibatkan kesalahan saat mulai ulang yang menunjukkan bahwa file data ruang tabel tidak dapat ditemukan.
- InnoDB: Memperbaiki masalah di mana server menghapus indeks kunci asing yang didefinisikan secara internal ketika menambahkan indeks baru dan mencoba menggunakan indeks sekunder yang ditentukan pada kolom yang dihasilkan secara virtual sebagai indeks kunci asing, sehingga menyebabkan server keluar. InnoDB sekarang mengizinkan batasan kunci asing untuk mereferensikan indeks sekunder yang didefinisikan pada kolom yang dihasilkan secara virtual. (Bug #23533396)
- Memperbaiki masalah di mana dua sesi secara bersamaan mengeksekusi operasi INSERT... ON DUPLICATE KEY UPDATE menghasilkan deadlock. Selama rollback sebagian tuple, sesi lain dapat memperbaruinya. Perbaikan untuk bug ini mengembalikan perbaikan untuk Bug #11758237, Bug #17604730, dan Bug #20040791. (Bug #25966845)
- Backport perbaikan untuk Bug Komunitas #27407480: Memperbaiki masalah di mana hak istimewa EXECUTE dan ALTER RUTINE tidak diberikan dengan benar kepada pembuat rutin bahkan dengan automatic_sp_privileges diaktifkan.
- Backport perbaikan untuk Bug Komunitas #24671968: Memperbaiki masalah di mana kueri dapat memberikan hasil yang salah jika klausa WHERE berisi subkueri dependen, tabel memiliki indeks sekunder pada kolom dalam daftar tertentu yang diikuti oleh kolom dalam subkueri, dan `GROUP BY` atau `DISTINCT` mengizinkan kueri untuk menggunakan Pemindaian Indeks Longgar.
- Memperbaiki masalah di mana replikasi rusak jika pernyataan penghapusan multi-tabel dikeluarkan terhadap beberapa tabel dengan kunci asing. (Bug #80821)
- Memperbaiki masalah di mana dalam kasus khusus, kesalahan slave tertentu tidak diabaikan bahkan dengan slave_skip_errors diaktifkan. Dalam kasus ketika pembukaan dan penguncian tabel gagal atau ketika konversi bidang gagal pada server yang menjalankan replikasi berbasis baris, kesalahan tersebut dianggap kritis dan status slave_skip_errors diabaikan. Perbaikan ini memastikan bahwa dengan slave_skip_errors diaktifkan, semua kesalahan yang dilaporkan selama penerapan transaksi ditangani dengan benar. (Bug #70640, Bug #17653275)
- Memperbaiki masalah di mana pernyataan `SET PASSWORD` direplikasi dari master MySQL 5.6 ke slave MySQL 5.7, atau dari master MySQL 5.7 dengan variabel sistem

log_builtin_as_identified_by_password diatur ke ON ke slave MySQL 5.7, hash kata sandi itu sendiri juga di-hash sebelum disimpan di slave. Masalah tersebut telah diperbaiki dan hash kata sandi yang direplikasi disimpan seperti aslinya diteruskan ke slave. (Bug#24687073)

- Memperbaiki masalah di mana serialisasi nilai JSON yang terdiri dari sub-dokumen besar yang dibungkus dalam banyak tingkatan array JSON, objek, atau keduanya, terkadang membutuhkan waktu yang terlalu lama untuk diselesaikan. (Bug #23031146)
- Pernyataan yang tidak dapat diuraikan (karena, misalnya, kesalahan sintaks) tidak lagi ditulis ke log kueri lambat. (Bug #33732907)

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL versi 2 (kompatibel dengan MySQL 5.7).

- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok

- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2022-11-01 (versi 2.10.3) (Dihentikan)

Versi: 2.10.3

Aurora MySQL 2.10.3 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7, dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 2.11.*, 3.01.*, dan 3.02.*.

Anda dapat meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.10.3. Untuk klaster yang menjalankan Aurora MySQL versi 1, Anda dapat meningkatkan klaster Aurora MySQL 1.23 atau yang lebih tinggi yang sudah ada langsung ke 2.10.3. Anda juga dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.10.3.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Memperbaiki masalah keamanan dan CVEs tercantum di bawah ini:

Perbaikan dan penyempurnaan lain untuk penanganan fine-tune di lingkungan terkelola. Di bawah ini adalah beberapa perbaikan CVE tambahan:

- CVE-2022-21444
- CVE-2022-21344
- CVE-2022-21304
- CVE-2022-21245
- CVE-2021-36222
- CVE-2021-22926

Perbaikan umum:

- Memperbaiki masalah yang, dalam kondisi yang sangat jarang, menyebabkan server basis data memulai ulang karena menunggu semaphore yang lama ketika thread pendeteksi deadlock macet.
- Memperbaiki masalah yang dapat menyebabkan memori yang dapat dilepas pada instans basis data menjadi berkurang ketika pernyataan SQL Data Control Language (DCL) tertentu seperti GRANT, FLUSH PRIVILEGES dll. dijalankan pada instans tersebut. Sering menggunakan pernyataan tersebut dapat menyebabkan memori yang dapat dibebaskan terus berkurang dan dapat menyebabkan instance database dimulai ulang karena out-of-memory masalah. Penggunaan pernyataan tersebut pada instans penulis juga dapat menyebabkan memori yang dapat dilepas pada instans pembaca berkurang.
- Memperbaiki masalah di mana kueri terhadap tabel “performance_schema.events_waits_summary_global_by_event_name” mungkin menjadi lambat ketika instance database berada di bawah beban berat dengan “_respond_to_client” performance_schema wait event diaktifkan. wait/io/aurora
- Memperbaiki masalah yang, dalam kondisi yang sangat jarang, dapat menyebabkan server basis data terhenti dan memulai ulang ketika transaksi mengalami rollback sebagian karena pelanggaran batasan pada indeks sekunder.
- Memperbaiki masalah yang, dalam kondisi yang sangat jarang, dapat menyebabkan instans penulis memulai ulang atau melakukan failover ketika transaksi mengakses baris yang dihapus oleh transaksi lain.

- Memperbaiki masalah yang, dalam kondisi langka, dapat menyebabkan database dimulai ulang karena menunggu semaphore yang lama ketika I/O utas menemui jalan buntu.
- Memperbaiki masalah yang dapat menyebabkan replika baca memulai ulang selama failover dalam kondisi yang sangat jarang saat file kunci soket Unix sedang digunakan.
- Memperbaiki masalah di mana pembatalan cache kueri yang berlebihan menyebabkan penggunaan CPU dan latensi yang lebih tinggi dari yang diharapkan pada replika baca karena replika baca harus membaca data dari disk, bukan dari cache kueri.

Integrasi perbaikan bug MySQL Community Edition

Rilis ini meliputi semua perbaikan bug komunitas hingga dan termasuk 5.7, selain hal-hal di bawah ini. Untuk informasi selengkapnya, lihat Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 2.x.

- Memperbaiki masalah di mana kode untuk membaca informasi kumpulan karakter dari tabel peristiwa pernyataan Skema Kinerja (misalnya, events_statements_current) tidak mencegah penulisan serentak ke informasi kumpulan karakter tersebut. Akibatnya, kumpulan karakter teks kueri SQL bisa menjadi tidak valid, yang dapat mengakibatkan server keluar. Dengan perbaikan ini, kumpulan karakter yang tidak valid menyebabkan pemotongan kolom SQL_TEXT dan mencegah server keluar. (Bug #23540008)
- Memperbaiki masalah ketika UPDATE memerlukan tabel sementara yang memiliki kunci primer lebih besar dari 1024 byte dan tabel tersebut dibuat menggunakan InnoDB, server dapat keluar. (Bug #25153670)
- Memperbaiki masalah di mana dua sesi secara bersamaan mengeksekusi operasi INSERT... ON DUPLICATE KEY UPDATE menghasilkan deadlock. Selama rollback sebagian tuple, sesi lain dapat memperbaruinya. Perbaikan untuk bug ini mengembalikan perbaikan untuk Bug #11758237, Bug #17604730, dan Bug #20040791. (Bug #25966845)

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL versi 2 (kompatibel dengan MySQL 5.7).

- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.

- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2022-01-26 (versi 2.10.2) (Dihentikan)

Versi: 2.10.2

Aurora MySQL 2.10.2 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7, dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Anda dapat meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.10.0. Untuk klaster yang menjalankan Aurora MySQL versi 1, Anda dapat meningkatkan klaster Aurora MySQL 1.23 atau yang lebih tinggi yang sudah ada langsung ke 2.10.0. Anda juga dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.10.0.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Memperbaiki masalah keamanan dan CVEs tercantum di bawah ini:

Perbaikan dan penyempurnaan lain untuk penanganan fine-tune di lingkungan terkelola. Di bawah ini adalah beberapa perbaikan CVE tambahan:

- CVE-2021-36222
- CVE-2021-35624
- CVE-2021-35604
- CVE-2021-22926
- CVE-2021-2390
- CVE-2021-2389
- CVE-2021-2389
- CVE-2021-2356

- CVE-2019-17543
- CVE-2019-2960

Perbaikan umum:

- Menambahkan pengoptimalan kinerja untuk membantu mengurangi latensi IO basis data di kelas instans 24XL.
- Menambahkan dukungan untuk cipher SSL ECDHE. Untuk informasi selengkapnya tentang konfigurasi klien Anda untuk menggunakan Cipher SSL ini, silakan lihat dokumentasi MySQL berikut, cipher protokol koneksi terenkripsi
- Memperbaiki masalah keamanan yang terkait dengan integrasi Aurora MySQL dengan Layanan lain AWS seperti Amazon S3, Amazon ML, dan. AWS Lambda
- Memperbaiki masalah yang dapat menyebabkan instans basis data gagal memulai ulang ketika basis data tersebut memiliki kombinasi pengguna dan hak istimewa berukuran lebih dari 1GB.
- Memperbaiki masalah dengan Kueri Paralel yang dapat menyebabkan basis data mengembalikan pengelompokan atau pengurutan yang salah saat mengeksekusi kueri dengan klausa GROUP BY dan klausa WHERE yang berisi predikat rentang.
- Memperbaiki masalah yang menyebabkan tabel general_log dan slow_log menjadi tidak dapat diakses setelah peningkatan versi utama secara in-place dari Aurora MySQL 1.x (kompatibel dengan MySQL 5.6) ke Aurora MySQL 2.x (kompatibel dengan MySQL 5.7).
- Memperbaiki masalah yang, dalam kasus yang jarang terjadi, menyebabkan instans basis data memulai ulang ketika tabel innodb_trx, innodb_locks, atau innodb_lockwaits menerima kueri saat basis data berada di bawah beban kerja yang berat. Alat pemantauan seperti Wawasan Kinerja dapat mengkueri tabel-tabel tersebut.
- Memperbaiki masalah di mana nilai kolom TIMESTAMP baris yang ada diperbarui ke stempel waktu terbaru saat semua kondisi berikut terpenuhi:
  1. Terdapat pemicu untuk tabel.
  2. INSERT dijalankan pada tabel yang memiliki klausa ON DUPLICATE KEY UPDATE.
  3. Baris yang disisipkan menyebabkan pelanggaran nilai duplikat dalam indeks UNIQUE atau PRIMARY KEY.
  4. Satu atau beberapa kolom adalah tipe data TIMESTAMP dan memiliki nilai default CURRENT_TIMESTAMP.
- Memperbaiki masalah yang, dalam kasus yang sangat jarang, dapat mencegah replika binlog terhubung ke instans dengan binlog diaktifkan.

- Memperbaiki masalah di mana, dalam kondisi yang sangat jarang, transaksi tidak dapat di-commit saat berjalan pada instans dengan binlog diaktifkan.
- Memperbaiki masalah di mana koneksi baru tidak dapat dibuat ke instans dengan binlog diaktifkan.
- Memperbaiki masalah yang dapat menyebabkan pencatatan log internal yang berlebihan saat mencoba patching dan mulai ulang zero-downtime yang menyebabkan penyimpanan lokal penuh.
- Memperbaiki masalah yang menyebabkan replika binlog berhenti dengan kesalahan HA_ERR_FOUND_DUPP_KEY saat mereplikasi pernyataan DDL dan DCL tertentu. Masalah tersebut terjadi ketika instans sumber dikonfigurasi dengan format pencatatan log biner MIXED dan tingkat isolasi READ COMMITTED atau READ UNCOMMITTED.
- Memperbaiki masalah saat I/O utas replikasi binlog tidak dapat mengikuti instance utama, saat replikasi multi-utas diaktifkan
- Memperbaiki masalah di mana, dalam kondisi langka, jumlah koneksi aktif yang tinggi ke instance database dapat menyebabkan CloudWatch CommitLatency metrik dilaporkan secara tidak benar.
- Memperbaiki masalah yang menyebabkan penyimpanan lokal pada instans Graviton terisi penuh saat menjalankan LOAD FROM S3 atau SELECT INTO S3.
- Memperbaiki masalah yang dapat menyebabkan hasil kueri yang salah saat mengkueri tabel dengan kunci asing dan kedua kondisi berikut terpenuhi:

  1. Cache kueri diaktifkan
  2. Transaksi dengan penghapusan atau pembaruan berjenjang pada tabel tersebut dibatalkan
- Memperbaiki masalah yang, dalam kondisi yang sangat jarang, dapat menyebabkan instans pembaca Aurora memulai ulang. Kemungkinan terjadinya masalah ini meningkat seiring dengan meningkatnya jumlah rollback transaksi.
- Memperbaiki masalah di mana jumlah kemunculan 'lock_epoch_id_master' mutex di Skema Kinerja meningkat saat sesi dibuka dan ditutup.
- Memperbaiki masalah yang dapat menyebabkan peningkatan jumlah deadlock untuk beban kerja yang memiliki banyak transaksi yang memperbarui set baris yang sama secara bersamaan.
- Memperbaiki masalah yang, dalam kondisi yang sangat jarang, dapat menyebabkan instans memulai ulang saat volume basis data meningkat ke kelipatan 160GB.
- Memperbaiki masalah pada Kueri Paralel yang dapat menyebabkan basis data memulai ulang ketika mengeksekusi pernyataan SQL dengan klausa LIMIT.
- Memperbaiki masalah yang, dalam kondisi yang sangat jarang, dapat menyebabkan instans basis data memulai ulang saat menggunakan transaksi XA dengan tingkat isolasi READ COMMITTED.

- Memperbaiki masalah di mana, setelah instans Baca Aurora memulai ulang, instans tersebut dapat memulai ulang lagi jika ada beban kerja DDL yang berat selama mulai ulang.
- Memperbaiki masalah terkait pelaporan yang salah tentang lag replikasi baca Aurora.
- Memperbaiki masalah yang, dalam kondisi yang sangat jarang, dapat menyebabkan instans penulis memulai ulang ketika pemeriksaan integritas data dalam memori gagal.
- Memperbaiki masalah yang, dalam kondisi yang sangat jarang, salah menunjukkan bagan “Pemuatan Basis Data” di sesi Wawasan Kinerja (PI) sebagai aktif menggunakan CPU meskipun sesi telah selesai diproses dan sedang tidak aktif.
- Memperbaiki masalah yang, dalam kondisi yang sangat jarang, dapat menyebabkan server basis data memulai ulang ketika sebuah kueri diproses menggunakan Kueri Paralel.
- Memperbaiki masalah yang, dalam kondisi sangat jarang, dapat menyebabkan instans penulis di klaster Basis Data Global primer memulai ulang karena kondisi balapan selama replikasi Basis Data Global.
- Memperbaiki masalah yang dapat terjadi selama instans basis data memulai ulang, yang dapat menyebabkan lebih dari satu kali mulai ulang.

Integrasi perbaikan bug MySQL Community Edition

- Memperbaiki masalah di InnoDB di mana kesalahan dalam kode yang terkait dengan statistik tabel menimbulkan pernyataan dalam file sumber dict0stats.cc. (Bug #24585978)
- Memperbaiki masalah di mana indeks sekunder di atas kolom virtual menjadi rusak saat indeks tersebut dibuat secara online. Untuk pernyataan UPDATE, kami memperbaikinya sebagai berikut: Jika nilai kolom virtual catatan indeks diatur ke NULL, kami menghasilkan nilai ini dari catatan indeks klaster. (Bug #30556595))
- Memperbaiki masalah di InnoDB di mana menghapus baris yang ditandai dapat mendatangkan kunci baca eksternal sebelum rollback sebagian selesai. Kunci baca eksternal tersebut mencegah konversi kunci implisit ke kunci eksplisit selama rollback sebagian, sehingga menyebabkan kegagalan pernyataan. (Bug #29195848)
- Memperbaiki masalah di mana nama host kosong di akun dapat menyebabkan server berperilaku tidak semestinya. (Bug #28653104)
- Memperbaiki masalah di InnoDB di mana gangguan kueri selama waktu tunggu kunci menyebabkan kesalahan. (Bug #28068293)

- Memperbaiki masalah dalam replikasi di mana Transaksi yang berseling terkadang dapat menyebabkan deadlock pada pengaplikasi slave saat tingkat isolasi transaksi diatur ke REPEATABLE READ. (Bug #25040331)
- Memperbaiki masalah yang dapat menyebabkan replika binlog macet karena batas waktu tunggu kunci. (Bug #27189701)

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL versi 2 (kompatibel dengan MySQL 5.7).

- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB

- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2021-10-21 (versi 2.10.1) (Dihentikan)

Versi: 2.10.1

Aurora MySQL 2.10.1 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7, dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Anda dapat meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.10.0. Untuk klaster yang menjalankan Aurora MySQL versi 1, Anda dapat meningkatkan klaster Aurora MySQL 1.23 atau yang lebih tinggi yang sudah ada langsung ke 2.10.0. Anda juga dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.10.0.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Memperbaiki masalah keamanan dan CVEs tercantum di bawah ini:

Perbaikan dan penyempurnaan lain untuk penanganan fine-tune di lingkungan terkelola. Di bawah ini adalah beberapa perbaikan CVE tambahan:

- CVE-2021-2307
- CVE-2021-2226
- CVE-2021-2194
- CVE-2021-2174
- CVE-2021-2171
- CVE-2021-2169
- CVE-2021-2166
- CVE-2021-2160
- CVE-2021-2154
- CVE-2021-2032
- CVE-2021-2001

Perbaikan ketersediaan:

- Menambahkan kemampuan untuk mematikan klaster secara bersih untuk peningkatan versi utama di masa mendatang.

Perbaikan umum:

- Memperbaiki masalah yang dapat menyebabkan konsumsi CPU tinggi pada instans pembaca karena pencatatan log pesan informasi yang berlebihan dalam file log diagnostik internal.
- Memperbaiki masalah di mana nilai kolom TIMESTAMP baris yang ada diperbarui ke stempel waktu terbaru saat semua kondisi berikut terpenuhi:
  1. Terdapat pemicu untuk tabel.
  2. INSERT dijalankan pada tabel yang memiliki klausa ON DUPLICATE KEY UPDATE.
  3. Baris yang disisipkan menyebabkan pelanggaran nilai duplikat dalam indeks UNIQUE atau PRIMARY KEY.
  4. Satu atau beberapa kolom adalah tipe data TIMESTAMP dan memiliki nilai default CURRENT_TIMESTAMP.
- Memperbaiki masalah yang diperkenalkan di versi 2.10.0 yang menyebabkan penggunaan fungsi json_merge memunculkan kode kesalahan dalam kasus tertentu. Secara khusus, ketika fungsi

json_merge digunakan dalam DDL yang berisi kolom yang dihasilkan, fungsi tersebut dapat mengembalikan kode kesalahan 1305.

- Memperbaiki masalah di mana, dalam kondisi yang sangat jarang, replika baca memulai ulang ketika riwayat pembaruan objek besar sedang divalidasi untuk tampilan baca transaksi pada replika baca.
- Memperbaiki masalah yang, dalam kondisi yang sangat jarang, menyebabkan instans penulis memulai ulang ketika pemeriksaan integritas data dalam memori gagal.

Integrasi perbaikan bug MySQL Community Edition

- CURRENT_TIMESTAMP MENGHASILKAN NOL PADA PEMICU. (Bug #25209512)

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL versi 2 (kompatibel dengan MySQL 5.7).

- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2021-05-25 (versi 2.10.0) (Dihentikan)

Versi: 2.10.0

Aurora MySQL 2.10.0 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7, dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Anda dapat meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.10.0. Untuk klaster yang menjalankan Aurora MySQL versi 1, Anda dapat meningkatkan klaster Aurora MySQL 1.23 atau yang lebih tinggi yang sudah ada langsung ke 2.10.0. Anda juga dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.10.0.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Memperbaiki masalah keamanan dan CVEs tercantum di bawah ini:

Perbaikan dan penyempurnaan lain untuk penanganan fine-tune di lingkungan terkelola. Di bawah ini adalah beberapa perbaikan CVE tambahan:

- CVE-2021-23841
- CVE-2021-3449
- CVE-2020-28196
- CVE-2020-14790
- CVE-2020-14776
- CVE-2020-14567
- CVE-2020-14559
- CVE-2020-14553
- CVE-2020-14547
- CVE-2020-14540
- CVE-2020-14539
- CVE-2018-3251
- CVE-2018-3156
- CVE-2018-3143
- CVE-2016-5440

Fitur baru:

- Kelas instans `db.t3.large` sekarang didukung untuk Aurora MySQL.
- Replikasi log biner:

  - Memperkenalkan I/O cache binlog untuk meningkatkan kinerja binlog dengan mengurangi pertengkaran antara thread penulis dan utas dump. Untuk informasi selengkapnya, lihat Mengoptimalkan replikasi log biner di Panduan Pengguna Amazon Aurora.
  - Di Aurora MySQL versi 2.08, kami memperkenalkan pemrosesan log biner (binlog) yang ditingkatkan untuk mengurangi waktu pemulihan akibat crash dan latensi waktu commit saat melibatkan transaksi yang sangat besar. Perbaikan ini sekarang didukung untuk klaster yang mengaktifkan GTID.
- Ketersediaan instans pembaca yang ditingkatkan:
  - Sebelumnya, saat instans penulis memulai ulang, semua instans pembaca di klaster Aurora MySQL juga memulai ulang. Dengan peluncuran hari ini, instans pembaca dalam Wilayah terus melayani permintaan baca selama instans penulis memulai ulang, sehingga meningkatkan ketersediaan baca di klaster. Untuk informasi selengkapnya, lihat Booting ulang klaster Aurora MySQL (versi 2.10 dan lebih tinggi) di Panduan Pengguna Amazon Aurora.

    > **Important**
    >
    > Setelah Anda meningkatkan ke Aurora MySQL 2.10, booting ulang instans penulis tidak akan menyebabkan boot ulang pada seluruh klaster. Jika Anda ingin melakukan booting ulang seluruh klaster, sekarang Anda mem-booting ulang semua instans pembaca di klaster setelah booting ulang instans penulis.

- Memperbaiki kinerja pembacaan halaman read ahead yang diminta oleh teknik logika read ahead (LRA). Hal ini dilakukan melalui batching beberapa pembacaan halaman dalam satu permintaan yang dikirimkan ke penyimpanan Aurora. Hasilnya, kueri yang menggunakan optimasi LRA mengeksekusi hingga 3x lebih cepat.
- Mulai ulang dan patching zero-downtime:
  - Memperbaiki zero-downtime restart (ZDR) dan zero-downtime patching (ZDP) untuk mengaktifkan ZDR dan ZDP dalam skenario yang lebih luas, termasuk dukungan tambahan untuk kasus ketika pencatatan log biner diaktifkan. Selain itu, memperbaiki visibilitas ke peristiwa ZDR dan ZDP. Lihat dokumentasi untuk mengetahui detailnya: Zero-downtime restart (ZDR) untuk Amazon Aurora MySQL dan Menggunakan zero-downtime patching di Panduan Pengguna Amazon Aurora.

Perbaikan ketersediaan:

- Perbaikan untuk startup yang lebih cepat ketika basis data memiliki indeks dan tabel sementara dalam jumlah besar yang dibuat selama aktivitas DDL yang terganggu sebelumnya.
- Memperbaiki beberapa masalah yang berkaitan dengan mulai ulang beberapa kali selama pemulihan crash operasi DDL yang terganggu, seperti `DROP TRIGGER`, `ALTER TABLE`, dan khususnya `ALTER TABLE` yang memodifikasi jenis partisi atau jumlah partisi di dalam tabel.
- Memperbaiki masalah yang dapat menyebabkan mulai ulang server selama pemrosesan log Aliran Aktivitas Basis Data (DAS).
- Memperbaiki masalah yang memunculkan pesan kesalahan saat memproses kueri `ALTER` pada tabel sistem.

Perbaikan umum:

- Memperbaiki masalah di mana cache kueri dapat mengembalikan hasil usang pada instans pembaca.
- Memperbaiki masalah di mana beberapa metrik commit Aurora tidak diperbarui saat variabel sistem `innodb_flush_log_at_trx_commit` diatur ke 0 atau 2.
- Memperbaiki masalah di mana hasil kueri yang disimpan dalam cache kueri tidak disegarkan oleh transaksi multi-pernyataan.
- Memperbaiki masalah yang dapat menyebabkan stempel waktu yang terakhir diubah dari file log biner tidak diperbarui dengan benar. Hal ini dapat menyebabkan file log biner dibersihkan sebelum waktunya, sebelum mencapai periode retensi yang dikonfigurasi pelanggan.
- Memperbaiki kesalahan nama dan posisi file binlog yang dilaporkan dari InnoDB setelah pemulihan crash.
- Memperbaiki masalah yang dapat menyebabkan transaksi besar menghasilkan peristiwa binlog yang salah jika parameter `binlog_checksum` diatur ke `NONE`.
- Memperbaiki masalah yang menyebabkan replika binlog berhenti dengan kesalahan jika transaksi yang direplikasi berisi pernyataan DDL dan perubahan baris dalam jumlah besar.
- Memperbaiki masalah yang menyebabkan mulai ulang dalam instans pembaca saat menghapus tabel.
- Memperbaiki masalah yang menyebabkan kegagalan konektor sumber terbuka saat mencoba mengonsumsi file binlog dengan transaksi besar.
- Memperbaiki masalah yang dapat menyebabkan hasil kueri salah pada kolom geometri besar setelah membuat indeks spasial pada tabel dengan nilai geometri besar.

- Basis data sekarang membuat ulang ruang tabel sementara selama mulai ulang, yang memungkinkan ruang penyimpanan terkait dilepaskan dan diklaim kembali.
- Memperbaiki masalah yang mencegah terpotongnya tabel `performance_schema` pada instans pembaca Aurora.
- Memperbaiki masalah yang menyebabkan replika binlog berhenti dengan kesalahan `HA_ERR_KEY_NOT_FOUND`.
- Memperbaiki masalah yang menyebabkan basis data memulai ulang saat menjalankan pernyataan `FLUSH TABLES WITH READ LOCK`.
- Memperbaiki masalah yang mencegah penggunaan fungsi penguncian tingkat pengguna pada instans pembaca Aurora.

Integrasi perbaikan bug MySQL Community Edition

- Transaksi yang berseling terkadang dapat menyebabkan deadlock pada pengaplikasi replika saat tingkat isolasi transaksi diatur ke REPEATABLE READ. (Bug #25040331)
- Ketika prosedur tersimpan berisi pernyataan yang merujuk ke suatu tampilan yang pada gilirannya merujuk ke tampilan lain, prosedur tidak dapat berhasil diinvokasi lebih dari sekali. (Bug #87858, Bug #26864199)
- Untuk kueri dengan banyak kondisi `OR`, pengoptimal sekarang lebih hemat memori dan lebih kecil kemungkinannya untuk melebihi batas memori yang diberlakukan oleh variabel sistem range_optimizer_max_mem_size. Selain itu, nilai default untuk variabel tersebut telah dinaikkan dari 1.536.000 menjadi 8.388.608. (Bug #79450, Bug #22283790)
- Replikasi: Di fungsi `next_event()`, yang dipanggil oleh thread SQL replika untuk membaca peristiwa berikutnya dari log relay, thread SQL tersebut tidak melepaskan `relaylog.log_lock` yang diperolehnya saat mengalami kesalahan (misalnya, karena log relay tertutup), sehingga menyebabkan hang pada semua thread lain yang menunggu untuk mendapatkan kunci pada log relay. Dengan perbaikan ini, kunci dilepaskan sebelum thread SQL meninggalkan fungsi dalam situasi tersebut. (Bug #21697821)
- Memperbaiki kerusakan memori untuk `ALTER TABLE` dengan kolom virtual. (Bug #24961167; Bug #24960450)
- Replikasi: Replika multithread tidak dapat dikonfigurasi dengan ukuran antrean kecil menggunakan slave_pending_jobs_size_max jika diperlukan untuk memproses transaksi yang lebih besar dari ukuran tersebut. Setiap paket yang lebih besar daripada slave_pending_jobs_size_max ditolak dengan kesalahan `ER_MTS_EVENT_BIGGER_PENDING_JOBS_SIZE_MAX`, bahkan jika paket

lebih kecil dari batas yang ditetapkan oleh slave_max_allowed_packet. Dengan perbaikan ini, slave_pending_jobs_size_maxmenjadi batas lembut, bukan batas keras. Jika ukuran paket melebihi slave_pending_jobs_size_max tapi kurang dari slave_max_allowed_packet, transaksi ditangguhkan sampai semua pekerja replika memiliki antrean kosong, dan kemudian diproses. Semua transaksi berikutnya ditangguhkan sampai transaksi besar selesai. Oleh karena itu, ukuran antrean untuk pekerja replika dapat dibatasi sambil tetap memungkinkan transaksi sesekali yang lebih besar. (Bug #21280753, Bug #77406)

- Replikasi: Ketika menggunakan replika multithread, kesalahan applier menampilkan data ID pekerja yang tidak konsisten dengan data yang dieksternalkan dalam tabel replikasi Skema Kinerja. (Bug #25231367)
- Replikasi: Pada replika replikasi berbasis GTID yang berjalan dengan -GTID-mode=on, -log-bin=off, dan menggunakan -, ketika kesalahan ditemukan yang harus diabaikan tidak diperbarui dengan benar, menyebabkan slave-skip-errors sinkronisasi yang longgar dengan. `Exec_Master_Log_Pos` `Exec_Master_Log_Pos` `Read_master_log_pos` Jika `GTID_NEXT` tidak ditentukan, replika tidak akan pernah memperbarui status GTID ketika dibatalkan dari transaksi pernyataan tunggal. `Exec_Master_Log_Pos` tidak akan diperbarui karena meskipun transaksi selesai, status GTID akan menunjukkan sebaliknya. Perbaikan ini menghilangkan batasan pembaruan status GTID saat transaksi dibatalkan hanya jika `GTID_NEXT` ditentukan. (Bug #22268777)
- Replikasi: Pernyataan yang gagal sebagian tidak menggunakan secara benar GTID yang ditentukan atau dibuat secara otomatis saat pencatatan log biner dinonaktifkan. Perbaikan ini memastikan bahwa DROP TABLE yang gagal sebagian, DROP USER yang gagal sebagian, atau DROP VIEW yang gagal sebagian menggunakan GTID masing-masing yang relevan dan menyimpannya ke dalam tabel `@@GLOBAL.GTID_EXECUTED` dan `mysql.gtid_executed` ketika pencatatan log biner dinonaktifkan. (Bug #21686749)
- Replikasi: Replika yang menjalankan MySQL 5.7 tidak dapat terhubung ke sumber MySQL 5.5 karena kesalahan saat mengambil server_uuid, yang bukan bagian dari MySQL 5.5. Hal ini disebabkan oleh perubahan pada metode pengambilan `server_uuid`. (Bug #22748612)
- Replikasi: Mekanisme melewatkan transaksi GTID yang secara diam-diam melewatkan transaksi GTID yang sebelumnya dijalankan tidak berfungsi dengan baik untuk transaksi XA. (Bug #25041920)
- Pernyataan ">XA ROLLBACK yang gagal karena ID transaksi yang diberikan salah, dapat direkam dalam log biner dengan ID transaksi yang benar, oleh karena itu dapat ditindaklanjuti dengan replika replikasi. Pemeriksaan sekarang dilakukan untuk situasi kesalahan sebelum pencatatan log biner terjadi, dan pernyataan `ROLLBACK` XA yang gagal tidak dicatat. (Bug #26618925)

- Replikasi: Jika replika disiapkan menggunakan pernyataan CHANGE MASTER TO yang tidak menentukan nama file log sumber dan posisi log sumber, lalu matikan sebelum START SLAVE dikeluarkan, lalu restart dengan opsi - relay-log-recovery set, replikasi tidak dimulai. Hal ini terjadi karena thread penerima belum dimulai sebelum pemulihan log relay diupayakan, jadi tidak ada peristiwa rotasi log yang tersedia di log relay untuk memberikan nama file log sumber dan posisi log sumber. Dalam situasi ini, replika sekarang melompati pemulihan log relay dan mencatat log peringatan, lalu melanjutkan untuk memulai replikasi. (Bug #28996606, Bug #93397)
- Replikasi: Dalam replikasi berbasis baris, sebuah pesan yang salah menampilkan panjang bidang dikembalikan saat mereplikasi dari tabel dengan kolom `utf8mb3` ke tabel definisi yang sama di mana kolom didefinisikan dengan kumpulan karakter `utf8mb4`. (Bug #25135304, Bug #83918)
- Replikasi: Ketika pernyataan RESET SLAVE dikeluarkan pada replika replikasi yang sedang digunakan, file log relai yang ada dibersihkan, tetapi file log relai baru pengganti dibuat sebelum kumpulan yang diterima GTIDs untuk saluran telah dihapus. GTIDs Oleh karena itu, set GTID sebelumnya ditulis ke file log relai baru sebagai `PREVIOUS_GTIDS` acara, menyebabkan kesalahan fatal dalam replikasi yang menyatakan bahwa replika memiliki GTIDs lebih dari sumber, meskipun set gtid_execute untuk kedua server kosong. Sekarang, ketika `RESET SLAVE` dikeluarkan, set yang GTIDs diterima dihapus sebelum file log relai baru dihasilkan, sehingga situasi ini tidak terjadi. (Bug #27411175)
- Replikasi: GTIDs Dengan digunakan untuk replikasi, transaksi termasuk pernyataan yang menyebabkan kesalahan penguraian (ER_PARSE_ERROR) tidak dapat dilewati secara manual dengan metode yang disarankan untuk menyuntikkan transaksi kosong atau penggantian dengan GTID yang sama. Tindakan ini seharusnya menghasilkan replika yang mengidentifikasi GTID seperti yang sudah digunakan, dan oleh karena itu melewatkan transaksi yang tidak diinginkan yang membagikan GTID-nya. Namun, dalam kasus kesalahan penguraian, karena pernyataan diuraikan sebelum GTID diperiksa untuk melihat apakah perlu dilewati, thread applier replikasi berhenti karena kesalahan penguraian, meskipun tujuannya adalah agar transaksi dilewati juga. Dengan perbaikan ini, thread applier replikasi sekarang mengabaikan kesalahan penguraian jika transaksi yang bersangkutan perlu dilewati karena GTID sudah digunakan. Perhatikan bahwa perubahan perilaku ini tidak berlaku dalam kasus beban kerja yang terdiri dari output log biner yang dihasilkan oleh `mysqlbinlog`. Dalam situasi tersebut, akan ada risiko bahwa transaksi dengan kesalahan penguraian yang segera mengikuti transaksi yang dilewati juga akan secara diam-diam dilewati, ketika transaksi tersebut akan meningkatkan kesalahan. (Bug #27638268)
- Replikasi: Mengaktifkan thread SQL untuk GTID melewatkan transaksi parsial. (Bug #25800025)

- Replikasi: Ketika parameter batas waktu negatif atau pecahan diberikan ke `WAIT_UNTIL_SQL_THREAD_AFTER_GTIDS()`, server berperilaku dengan cara yang tidak terduga. Dengan perbaikan ini:
  - Nilai batas waktu pecahan dibaca apa adanya, tanpa pembulatan.
  - Nilai batas waktu negatif ditolak dengan kesalahan jika server menggunakan mode SQL yang ketat; jika server tidak dalam mode SQL yang ketat, nilainya membuat fungsi segera mengembalikan `NULL` tanpa menunggu apa pun dan kemudian mengeluarkan peringatan. (Bug #24976304, Bug #83537)
- Replikasi: Jika fungsi `WAIT_FOR_EXECUTED_GTID_SET()` digunakan dengan nilai batas waktu termasuk bagian pecahan (misalnya, 1,5), kesalahan dalam logika casting berarti batas waktu dibulatkan ke bawah ke seluruh detik terdekat, dan ke nol untuk nilai kurang dari 1 detik (misalnya, 0,1). Logika casting sekarang telah diperbaiki sehingga nilai batas waktu diterapkan seperti yang ditentukan semula tanpa pembulatan. Terima kasih kepada Dirkjan Bussink atas kontribusinya. (Bug #29324564, Bug #94247)
- Dengan GTIDs diaktifkan, XA COMMIT pada transaksi XA yang terputus dalam transaksi multi-pernyataan menimbulkan pernyataan. (Bug #22173903)
- Replikasi: Sebuah pernyataan diangkat dalam build debug jika pernyataan XA ROLLBACK dikeluarkan untuk pengenal transaksi yang tidak diketahui ketika nilai gtid_next telah ditetapkan secara manual. Server sekarang tidak mencoba memperbarui status GTID jika pernyataan `ROLLBACK` XA gagal dengan kesalahan. (Bug #27928837, Bug #90640)
- Memperbaiki masalah urutan penyortiran yang salah ketika beberapa fungsi `CASE` yang digunakan dalam klausa `ORDER BY` (Bug #22810883).
- Beberapa kueri yang menggunakan pengurutan dapat mengakses kolom yang tidak diinisialisasi selama pengoptimalan dan menyebabkan server keluar. (Bug #27389294)

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL versi 2 (kompatibel dengan MySQL 5.7).

- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.

- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2021-11-12 (versi 2.09.3) (Dihentikan)

Versi: 2.09.3

Aurora MySQL 2.09.3 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7, dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Anda dapat meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.10.0. Untuk klaster yang menjalankan Aurora MySQL versi 1, Anda dapat meningkatkan klaster Aurora MySQL 1.23 atau yang lebih tinggi yang sudah ada langsung ke 2.10.0. Anda juga dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.10.0.

Untuk membuat klaster dengan Aurora MySQL versi lama, tentukan versi mesin melalui Konsol Manajemen AWS, AWS CLI, atau API Amazon RDS.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan keamanan:

Perbaikan dan penyempurnaan lain untuk penanganan fine-tune di lingkungan terkelola. Di bawah ini adalah beberapa perbaikan CVE tambahan:

- CVE-2021-23841
- CVE-2021-3712
- CVE-2021-3449
- CVE-2021-2307
- CVE-2021-2226

- CVE-2021-2174
- CVE-2021-2171
- CVE-2021-2169
- CVE-2021-2166
- CVE-2021-2154
- CVE-2021-2060
- CVE-2021-2032
- CVE-2021-2001
- CVE-2020-28196
- CVE-2020-14769
- CVE-2019-17543
- CVE-2019-2960

Perbaikan ketersediaan:

- Memperkenalkan optimalisasi yang dapat mengurangi perselisihan untuk kueri yang dijalankan pada tabel di information_schema.
- Menambahkan dukungan untuk cipher SSL ECDHE.

Perbaikan umum:

- Memperbaiki masalah yang, dalam kondisi yang sangat jarang, dapat menyebabkan instans penulis memulai ulang ketika pemeriksaan integritas data dalam memori gagal.
- Memperbaiki masalah yang, dalam kondisi yang sangat jarang, dapat menyebabkan instans basis data memulai ulang saat volume klaster meluas saat pencatatatan log biner diaktifkan.
- Memperbaiki kondisi balapan yang sangat jarang selama instans basis data memulai ulang, yang dapat menyebabkan lebih dari satu kali mulai ulang.
- Memperbaiki masalah yang dapat menyebabkan instans basis data gagal memulai ulang ketika basis data tersebut memiliki kombinasi pengguna dan hak istimewa dalam jumlah besar.
- Memperbaiki masalah pada kueri paralel yang dapat menyebabkan basis data memulai ulang ketika mengeksekusi pernyataan SQL dengan klausa LIMIT.
- Memperbaiki masalah terkait pelaporan yang salah tentang lag replikasi aurora.

- Memperbaiki masalah yang dapat menyebabkan tabel general_log dan slow_log menjadi tidak dapat diakses setelah peningkatan versi utama secara in-place dari Aurora-MySQL 1.x (berdasarkan MySQL 5.6) ke Aurora-MySQL 2.x (berdasarkan MySQL 5.7).
- Memperbaiki masalah yang, dalam kasus yang jarang terjadi, dapat menyebabkan instans basis data memulai ulang ketika tabel innodb_trx, innodb_locks, atau innodb_lockwaits menerima kueri saat basis data berada di bawah beban kerja yang berat. Alat dan fitur pemantauan seperti wawasan kinerja dapat mengkueri tabel-tabel tersebut.
- Memperbaiki masalah yang dapat menyebabkan instans basis data memulai ulang ketika pernyataan SQL “FLUSH TABLES WITH READ LOCK” dijalankan.
- Memperbaiki masalah di mana proses pembersihan InnoDB terjeda selama penghapusan instans pembaca yang menyebabkan peningkatan sementara dalam panjang daftar riwayat.
- Memperbaiki masalah dengan kueri paralel yang dapat menyebabkan basis data memulai ulang ketika mengeksekusi pernyataan SQL terhadap tabel yang berisi kolom virtual.
- Memperbaiki masalah dengan kueri paralel yang dapat menyebabkan basis data mengembalikan pengelompokan atau pengurutan yang salah saat mengeksekusi kueri dengan klausa GROUP BY dan klausa WHERE yang berisi predikat rentang.
- Memperbaiki masalah pada kueri paralel yang, dalam kondisi yang sangat jarang, dapat menyebabkan basis data memulai ulang ketika mengeksekusi pernyataan SQL dengan fungsi JSON.
- Memperbaiki masalah yang, dalam kondisi sangat jarang, dapat menyebabkan instans penulis di klaster Basis Data Global primer memulai ulang karena kondisi balapan selama Replikasi Basis Data Global.
- Memperbaiki masalah yang dapat menyebabkan replika Binlog berhenti dengan kesalahan HA_ERR_FOUND_DUPP_KEY saat mereplikasi pernyataan DDL dan DCL tertentu. Masalah tersebut terjadi ketika instans sumber dikonfigurasi dengan format pencatatan log biner MIXED dan tingkat isolasi READ COMMITTED atau READ UNCOMMITTED.
- Memperbaiki masalah yang, dalam kondisi yang sangat jarak, dapat menyebabkan instans basis data memulai ulang saat menggunakan transaksi XA di tingkat isolasi READ COMMITTED.
- Memperbaiki masalah di mana nilai kolom TIMESTAMP baris yang sudah ada diperbarui ke stempel waktu terbaru ketika semua kondisi berikut ini terpenuhi: 1. terdapat sebuah pemicu untuk tabel; 2. INSERT dilakukan pada tabel yang memiliki klausa ON DUPLICATE KEY UPDATE; 3. baris yang disisipkan dapat menyebabkan pelanggaran nilai duplikat dalam indeks UNIQUE atau PRIMARY KEY; dan 4. satu kolom atau lebih adalah tipe data TIMESTAMP dan memiliki nilai default CURRENT_TIMESTAMP.

- Memperbaiki masalah yang, dalam kondisi yang sangat jarang, dapat menyebabkan instans pembaca memulai ulang karena pemrosesan pemeriksaan yang salah.
- Memperbaiki masalah yang dapat menyebabkan instans pembaca memulai ulang saat instans penulis meningkatkan volume basis data ke batas ukuran volume tertentu.
- Memperbaiki masalah yang dapat menyebabkan waktu mulai ulang yang lebih lama untuk instans basis data menggunakan volume klaster yang diklon.
- Memperbaiki masalah di mana instans basis data gagal memulai ulang satu kali atau beberapa kali setelah operasi TRUNCATE TABLE dilakukan pada instans penulis.
- Memperbaiki masalah yang, dalam kondisi yang sangat jarang, dapat menyebabkan instans basis data memulai ulang.
- Memperbaiki masalah yang, dalam kondisi yang sangat jarang, dapat menyebabkan instans penulis memulai ulang saat volume basis data meningkat ke kelipatan 160GB.

Integrasi perbaikan bug MySQL Community Edition

- Bug #23533396 - Saat menambahkan indeks baru, server menghapus indeks kunci asing yang didefinisikan secara internal dan mencoba menggunakan indeks sekunder yang ditentukan pada kolom yang dihasilkan secara virtual sebagai indeks kunci asing, sehingga menyebabkan server keluar. InnoDB sekarang mengizinkan batasan kunci asing untuk mereferensikan indeks sekunder yang didefinisikan pada kolom yang dihasilkan secara virtual.
- Bug #29550513 - Replikasi: Masalah penguncian di fungsi WAIT_FOR_EXECUTED_GTID_SET() dapat menyebabkan server macet dalam keadaan tertentu. Masalah tersebut telah diperbaiki.

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.

- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2021-02-26 (versi 2.09.2) (Dihentikan)

Versi: 2.09.2

Aurora MySQL 2.09.2 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Anda dapat meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.09.2. Untuk klaster yang menjalankan Aurora MySQL versi 1, Anda dapat meningkatkan klaster Aurora MySQL 1.23 atau yang lebih tinggi yang sudah ada langsung ke 2.09.2. Anda juga dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.09.2.

Untuk membuat klaster dengan Aurora MySQL versi lama, tentukan versi mesin melalui Konsol Manajemen AWS, AWS CLI, atau API Amazon RDS.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Fitur baru:

- Cluster Aurora MySQL sekarang mendukung instans EC2 R6g berikut yang didukung oleh prosesor Graviton2 berbasis AWS ARM:,,,,,,. `r6g.large` `r6g.xlarge` `r6g.2xlarge` `r6g.4xlarge` `r6g.8xlarge` `r6g.12xlarge` `r6g.16xlarge` Untuk informasi selengkapnya, lihat Kelas instans DB Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan keamanan:

Perbaikan dan penyempurnaan lain untuk penanganan fine-tune di lingkungan terkelola. Di bawah ini adalah beberapa perbaikan CVE tambahan:

- CVE-2020-14790
- CVE-2020-14793
- CVE-2020-14765
- CVE-2020-14769
- CVE-2020-14812
- CVE-2020-14760
- CVE-2020-14672
- CVE-2020-14790
- CVE-2020-1971

Perbaikan ketersediaan:

- Memperbaiki masalah yang muncul di 2.09.0 yang dapat menyebabkan peningkatan latensi tulis selama penskalaan volume penyimpanan klaster.
- Memperbaiki masalah pada fitur penyesuaian ukuran dinamis yang dapat menyebabkan Aurora Read Replica memulai ulang.
- Memperbaiki masalah yang dapat menyebabkan downtime yang lebih lama selama peningkatan dari 1.23.* ke 2.09.*.
- Memperbaiki masalah di mana DDL atau DML dapat menyebabkan mulai ulang mesin selama permintaan prefetch halaman.
- Memperbaiki masalah yang menyebabkan replika binlog berhenti dengan kesalahan jika transaksi yang direplikasi berisi pernyataan DDL dan perubahan baris dalam jumlah besar.
- Memperbaiki masalah di mana basis data yang bertindak sebagai replika binlog dapat memulai ulang saat mereplikasi peristiwa DDL di tabel `time_zone` mysql.
- Memperbaiki masalah yang dapat menyebabkan transaksi besar menghasilkan peristiwa binlog yang salah jika parameter `binlog_checksum` diatur ke `NONE`.
- Memperbaiki masalah yang menyebabkan replika binlog berhenti dengan kesalahan `HA_ERR_KEY_NOT_FOUND`.
- Peningkatan stabilitas keseluruhan.

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL Versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL Versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online

- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2020-12-11 (versi 2.09.1) (Dihentikan)

Versi: 2.09.1

Aurora MySQL 2.09.1 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Anda dapat meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.09.1. Untuk klaster yang menjalankan Aurora MySQL versi 1, Anda dapat meningkatkan klaster Aurora MySQL 1.23 atau yang lebih tinggi yang sudah ada langsung ke 2.09.1. Anda juga dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.09.1.

Untuk membuat klaster dengan Aurora MySQL versi lama, tentukan versi mesin melalui Konsol Manajemen AWS, AWS CLI, atau API RDS.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan keamanan:

Perbaikan dan penyempurnaan lain untuk penanganan fine-tune di lingkungan terkelola. Di bawah ini adalah beberapa perbaikan CVE tambahan:

- CVE-2020-14567
- CVE-2020-14559
- CVE-2020-14553
- CVE-2020-14547
- CVE-2020-14540
- CVE-2020-2812
- CVE-2020-2806
- CVE-2020-2780
- CVE-2020-2765
- CVE-2020-2763
- CVE-2020-2760
- CVE-2020-2579

Perubahan tidak kompatibel:

Versi ini memperkenalkan perubahan izin yang memengaruhi perilaku perintah `mysqldump`. Pengguna harus memiliki hak istimewa `PROCESS` untuk mengakses tabel `INFORMATION_SCHEMA.FILES`. Untuk menjalankan perintah `mysqldump` tanpa perubahan apa pun, berikan hak istimewa `PROCESS` untuk pengguna basis data yang terhubung dengan perintah `mysqldump`. Anda juga dapat menjalankan perintah `mysqldump` dengan opsi `--no-tablespaces`. Dengan opsi tersebut, output `mysqldump` tidak menyertakan pernyataan `CREATE LOGFILE GROUP` atau `CREATE TABLESPACE` apa pun. Dalam hal ini, perintah `mysqldump` tidak mengakses tabel `INFORMATION_SCHEMA.FILES`, dan Anda tidak perlu memberikan izin `PROCESS`.

Perbaikan ketersediaan:

- Memperbaiki masalah yang mungkin menyebabkan sesi klien macet saat mesin basis data mengalami kesalahan saat membaca dari atau menulis ke jaringan.
- Memperbaiki kebocoran memori pada fitur penyesuaian ukuran dinamis, yang diperkenalkan di 2.09.0.

Basis data global:

- Memperbaiki beberapa masalah yang menyebabkan replika Wilayah sekunder basis data global mungkin memulai ulang saat ditingkatkan ke rilis 2.09.0 sementara penulis Wilayah primer adalah versi rilis yang lebih lama.

Integrasi perbaikan bug MySQL Community Edition

- Replikasi: Transaksi yang berseling terkadang dapat menyebabkan deadlock pada pengaplikasi slave saat tingkat isolasi transaksi diatur ke REPEATABLE READ. (Bug #25040331)
- Untuk tabel dengan kolom TIMESTAMP atau DATETIME yang memiliki nilai default CURRENT_TIMESTAMP, kolom tersebut dapat diinisialisasi ke `0000-00-00 00:00:00` jika tabel memiliki pemicu `BEFORE INSERT`. (Bug #25209512, Bug #84077)
- Untuk pernyataan INSERT yang menerima pembuatan nilai oleh daftar `VALUES` untuk baris kedua atau setelahnya menggunakan subkueri yang berisi sambungan, server dapat keluar setelah gagal menyelesaikan hak istimewa yang diperlukan. (Bug #23762382)

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL Versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL Versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2020-09-17 (versi 2.09.0) (Dihentikan)

Versi: 2.09.0

Aurora MySQL 2.09.0 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Anda dapat memulihkan snapshot dari Aurora MySQL 1.23.* menjadi Aurora MySQL 2.09.0. Anda juga memiliki opsi untuk meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.09.0. Anda tidak dapat meningkatkan klaster Aurora MySQL 1.23.* yang ada secara langsung ke 2.09.0; namun, Anda dapat memulihkan snapshot-nya ke Aurora MySQL 2.09.0.

> **Important**
>
> Perbaikan pada penyimpanan Aurora dalam versi ini membatasi jalur peningkatan yang tersedia dari Aurora MySQL 1.* ke Aurora MySQL 2.09. Saat Anda meningkatkan klaster Aurora MySQL 1.* ke 2.09, Anda harus meningkatkan versi dari Aurora MySQL 1.23.

Untuk membuat cluster dengan versi lama dari Aurora MySQL, tentukan versi mesin melalui,, atau RDS Konsol Manajemen AWS API. AWS CLI

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Fitur baru:

- Dengan rilis ini, Anda dapat membuat instans basis data Amazon Aurora MySQL dengan penyimpanan hingga 128 tebibyte (TiB). Batas penyimpanan baru meningkat dari 64 TiB. Ukuran penyimpanan 128 TiB mendukung basis data yang lebih besar. Kapasitas ini tidak didukung pada ukuran instans kecil (db.t2 atau db.t3). Satu ruang tabel tidak dapat meluas melebihi 64 TiB karena adanya batasan InnoDB dengan ukuran halaman 16 KB.

  Aurora memperingatkan Anda saat ukuran volume klaster mendekati 128 TiB, sehingga Anda dapat mengambil tindakan sebelum mencapai batas ukuran. Peringatan muncul di log mysql dan Peristiwa RDS di Konsol Manajemen AWS.
- Anda sekarang dapat mengaktifkan atau menonaktifkan kueri paralel untuk klaster yang ada dengan mengubah nilai parameter klaster DB `aurora_parallel_query`. Anda tidak perlu menggunakan pengaturan `parallelquery` untuk parameter `--engine-mode` saat membuat klaster.

Kueri paralel sekarang diperluas agar tersedia di semua wilayah di mana Aurora MySQL tersedia.

Ada sejumlah peningkatan dan perubahan fungsi lain pada prosedur untuk peningkatan dan pengaktifan kueri paralel di klaster Aurora. Untuk informasi selengkapnya, lihat Menggunakan kueri paralel untuk Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

- Aurora secara dinamis menyesuaikan ukuran ruang penyimpanan klaster Anda. Dengan penyesuaian ukuran dinamis, ruang penyimpanan untuk klaster DB Aurora menurun secara otomatis saat Anda menghapus data dari klaster DB. Lihat informasi selengkapnya di Penskalaan penyimpanan di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Fitur pengubahan ukuran dinamis sedang digunakan secara bertahap ke AWS Wilayah tempat Aurora tersedia. Tergantung Wilayah tempat klaster Anda berada, fitur ini mungkin belum tersedia. Untuk informasi selengkapnya, lihat pengumuman Apa Yang Baru.

Perbaikan prioritas tinggi:

- Backport Bug Komunitas #27659490: SELECT MENGGUNAKAN DYNAMIC RANGE DAN INDEX MERGE MENGGUNAKAN TERLALU BANYAK MEMORI (OOM)
- Bug #26881508: MYSQL #1: DISABLE_ABORT_ON_ERROR IN AUTH_COMMON.H
- Backport Bug Komunitas #24437124: KEMUNGKINAN BUFFER OVERFLOW PADA CREATE TABLE
- Backport Bug #27158030: INNODB ONLINE ALTER CRASH DENGAN DML SECARA BERSAMAAN
- Bug #29770705: SERVER CRASH SAAT MELAKUKAN SELECT DENGAN KLAUSA WHERE KHUSUS
- Backport BUG #26502135: MYSQLD SEGFAULTS DALAM MDL_CONTEXT::TRY_ACQUIRE_LOCK_IMPL
- Bug #26935001: TABEL ALTERNATIF AUTO_INCREMENT MENCOBA MEMBACA INDEKS DARI RUANG TABEL YANG DIHAPUS
- Bug #28491099: [FATAL] BLOK MEMORI TIDAK VALID | INNODB: KEGAGALAN ASSERSI: UT0UT.CC:670

- Bug #30499288: GCC 9.2.1 MELAPORKAN PERINGATAN BARU UNTUK OS_FILE_GET_PARENT_DIR
- Bug #29952565 di mana MYSQLD MENDAPATKAN SIGNAL 11 SAAT MENJALANKAN KUERI (UNION + ORDER BY + SUB-QUERY)
- Bug #30628268: CRASH KEHABISAN MEMORI
- Bug #30441969: BUG #29723340: MYSQL SERVER CRASH SETELAH KUERI SQL DENGAN ? AST DATA
- Bug #30569003: 5.7 REPLICATION BREAKAGE DENGAN KESALAHAN SINTAKS DENGAN GRANT MANAGEMENT
- Bug #29915479: MENJALANKAN COM_REGISTER_SLAVE TANPA COM_BINLOG_DUMP DAPAT BERDAMPAK PADA KELUAR SERVER
- Bug #30569003: 5.7 REPLICATION BREAKAGE DENGAN KESALAHAN SINTAKS DENGAN GRANT MANAGEMENT
- Bug #29915479: MENJALANKAN COM_REGISTER_SLAVE TANPA COM_BINLOG_DUMP DAPAT BERDAMPAK PADA KELUAR SERVER
- Bug #20712046: TAMPILKAN TABEL PROCESSLIST DAN PERFORMANCE_SCHEMA TIDAK MENYEMBUNYIKAN KATA SANDI DARI KUERI
- Backport bug #18898433: KINERJA SANGAT LAMBAT DENGAN OUTER JOIN DAN JOIN BUFFER (diperbaiki dalam 5.7.21). Kueri dengan banyak sambungan tersisa akan lambat jika buffering sambungan digunakan (misalnya, menggunakan algoritma loop bersarang blok). (Bug #18898433, Bug #72854)"
- Backport bug #26402045: MYSQLD CRASH PADA KUERI (diperbaiki dalam MySQL 5.7.23). Kasus materialisasi subkueri tertentu dapat menyebabkan keluarnya server. Kueri ini sekarang menghasilkan pesan kesalahan yang menunjukkan bahwa materialisasi dinonaktifkan. (Bug #26402045)
- [Backport dari MySQL] pengguna selain rdsadmin tidak diizinkan untuk memperbarui tabel pfs di replika pembaca.
- Memperbaiki masalah di mana pelanggan tidak dapat memperbarui perfschema di replika pembaca
- Bug #26666274: LOOP TANPA HENTI DI KONTAINER BUFFER SKEMA KINERJA
- Bug #26997096: Nilai relay_log_space tidak diperbarui secara sinkron sehingga nilainya terkadang jauh lebih tinggi daripada ruang disk aktual yang digunakan oleh log relay.

- BUG #25082593: VALIDASI KUNCI ASING TIDAK PERLU MEMPEROLEH GAP LOCK DI READ COMMITTED
- CVE-2019-2731
- CVE-2018-2645
- CVE-2019-2581
- CVE-2018-2787
- CVE-2019-2482
- CVE-2018-2640
- CVE-2018-2784
- CVE-2019-2628
- CVE-2019-2911
- CVE-2019-2628
- CVE-2018-3284
- CVE-2018-3065
- CVE-2019-2537
- CVE-2019-2948
- CVE-2019-2434
- CVE-2019-2420

Perbaikan ketersediaan:

- Mengaktifkan perbaikan ABA manajer kunci secara default.
- Memperbaiki masalah pada manajer kunci di mana kondisi balapan dapat menyebabkan sebuah kunci dimiliki bersama oleh dua transaksi, yang menyebabkan basis data memulai ulang.
- Memperbaiki masalah saat membuat tabel sementara dengan format baris terkompresi dapat menyebabkan mulai ulang.
- Memperbaiki nilai default `table_open_cache` pada instans 16XL dan 24XL yang dapat menyebabkan failover berulang dan penggunaan CPU yang tinggi pada kelas instans besar (R4/R5-16XL, R5-12XL, R5-24XL). Hal ini berdampak pada 2.07.x.
- Memperbaiki masalah saat memulihkan klaster dari Amazon S3 ke Aurora MySQL versi 2.08.0 membutuhkan waktu lebih lama dari yang diperkirakan ketika cadangan S3 tidak menyertakan tabel `mysql.host`.

- Memperbaiki masalah yang mungkin menyebabkan failover beberapa kali karena pembaruan kolom virtual dengan indeks sekunder.
- Memperbaiki masalah yang berkaitan dengan manajemen memori kunci transaksi dengan transaksi tulis jangka panjang yang menyebabkan basis data memulai ulang.
- Memperbaiki beberapa masalah di mana mesin mungkin mengalami crash selama patching zero-downtime sambil memeriksa titik aman untuk patching.
- Memperbaiki masalah untuk melewatkan pencatatan log pengulangan tindakan untuk tabel sementara, yang sebelumnya menyebabkan crash.
- Memperbaiki kondisi balapan di manajer kunci antara pembunuhan connection/query dan sesi terbunuh.
- Memperbaiki masalah di mana basis data dapat mengalami crash jika berupa replika binlog dan menerima peristiwa DDL melalui tabel `time_zone` MySQL.

Basis data global:

- Tampilan `INFORMATION_SCHEMA.REPLICA_HOST_STATUS` MySQL di Wilayah sekunder sekarang menunjukkan entri untuk replika milik Wilayah tersebut.
- Memperbaiki kegagalan kueri tak terduga yang dapat terjadi di Wilayah sekunder DB Global setelah masalah konektivitas jaringan sementara antara Wilayah primer dan sekunder.
-

Kueri paralel:

- Memperbaiki rencana `EXPLAIN` untuk kueri Parallel Query, yang tidak benar untuk kueri tabel tunggal sederhana.
- Memperbaiki self-deadlatch yang mungkin terjadi saat Parallel Query diaktifkan.

Perbaikan umum:

- Ekspor ke S3 sekarang mendukung kata kunci `ENCRYPTION`.
- Parameter `aurora_binlog_replication_max_yield_seconds` saat ini memiliki nilai maksimal 36.000. Nilai maksimum yang diterima sebelumnya adalah 45. Parameter ini hanya berfungsi jika parameter `aurora_binlog_use_large_read_buffer` diatur ke 1.

- Mengubah perilaku untuk memetakan `MIXED` `binlog_format` ke `ROW` alih-alih `STATEMENT` saat menjalankan `LOAD DATA FROM INFILE | S3`.
- Memperbaiki masalah di mana replika binlog yang terhubung ke primer binlog Aurora MySQL mungkin menampilkan data yang tidak lengkap ketika primer yang dijalankan `LOAD DATA FROM S3` dan `binlog_format` diatur ke `STATEMENT`.
- Meningkatkan panjang maksimum yang diizinkan untuk variabel sistem audit `server_audit_incl_users` dan `server_audit_excl_users` dari 1024 byte menjadi 2000 byte.
- Memperbaiki masalah di mana pengguna dapat kehilangan akses ke basis data saat menurunkan parameter `max_connections` dalam grup parameter ketika koneksi saat ini lebih besar daripada nilai yang ditetapkan.
- Memperbaiki masalah pada Aliran Aktivitas Data di mana tanda kutip tunggal dan garis miring terbalik tidak dilepaskan dengan benar.

Integrasi perbaikan bug MySQL Community Edition

- Bug #27659490: SELECT MENGGUNAKAN DYNAMIC RANGE DAN INDEX MERGE MENGGUNAKAN TERLALU BANYAK MEMORI(OOM)
- Bug #26881508: MYSQL #1: DISABLE_ABORT_ON_ERROR IN AUTH_COMMON.H
- Bug #24437124: KEMUNGKINAN BUFFER OVERFLOW PADA CREATE TABLE
- Bug #27158030: INNODB ONLINE ALTER CRASH DENGAN DML SECARA BERSAMAAN
- Bug #29770705: SERVER CRASH SAAT MELAKUKAN SELECT DENGAN KLAUSA WHERE KHUSUS
- BUG #26502135: MYSQLD SEGFAULTS DALAM MDL_CONTEXT::TRY_ACQUIRE_LOCK_IMPL
- Bug #26935001: TABEL ALTERNATIF AUTO_INCREMENT MENCOBA MEMBACA INDEKS DARI RUANG TABEL YANG DIHAPUS
- Bug #28491099: [FATAL] BLOK MEMORI TIDAK VALID | INNODB: KEGAGALAN ASSERSI: UT0UT.CC:670
- Bug #30499288: GCC 9.2.1 MELAPORKAN PERINGATAN BARU UNTUK OS_FILE_GET_PARENT_DIR
- Bug #29952565 di mana MYSQLD MENDAPATKAN SIGNAL 11 SAAT MENJALANKAN KUERI (UNION + ORDER BY + SUB-QUERY)
- Bug #30628268: CRASH KEHABISAN MEMORI

- Bug #30441969: BUG #29723340: MYSQL SERVER CRASH SETELAH KUERI SQL DENGAN ? AST DATA
- Bug #30569003: 5.7 REPLICATION BREAKAGE DENGAN KESALAHAN SINTAKS DENGAN GRANT MANAGEMENT
- Bug #29915479: MENJALANKAN COM_REGISTER_SLAVE TANPA COM_BINLOG_DUMP DAPAT BERDAMPAK PADA KELUAR SERVER
- Bug #30569003: 5.7 REPLICATION BREAKAGE DENGAN KESALAHAN SINTAKS DENGAN GRANT MANAGEMENT
- Bug #29915479: MENJALANKAN COM_REGISTER_SLAVE TANPA COM_BINLOG_DUMP DAPAT BERDAMPAK PADA KELUAR SERVER
- Bug #20712046: TAMPILKAN TABEL PROCESSLIST DAN PERFORMANCE_SCHEMA TIDAK MENYEMBUNYIKAN KATA SANDI DARI KUERI
- Bug #18898433: KINERJA SANGAT LAMBAT DENGAN OUTER JOIN DAN JOIN BUFFER (diperbaiki dalam 5.7.21)
- Bug #26402045: MYSQLD CRASH PADA KUERI (diperbaiki dalam MySQL 5.7.23)
- Bug #23103937: PS_TRUNCATE_ALL_TABLES() TIDAK DAPAT DIGUNAKAN DALAM MODE SUPER_READ_ONLY
- Bug #26666274: LOOP TANPA HENTI DI KONTAINER BUFFER SKEMA KINERJA
- Bug #26997096: Nilai relay_log_space tidak diperbarui secara sinkron sehingga nilainya terkadang jauh lebih tinggi daripada ruang disk aktual yang digunakan oleh log relay. (https://github.com/mysql/mysql-server/commit/78f25d2809ad457e81f90342239c9bc32a36cdfa)
- Bug #25082593: VALIDASI KUNCI ASING TIDAK PERLU MEMPEROLEH GAP LOCK DI READ COMMITTED
- Bug #24764800: REPLIKASI GAGAL PADA SLAVE DENGAN KESALAHAN XAER_RMFAIL.
- Bug #81441: PERINGATAN TENTANG LOCALHOST SAAT MENGGUNAKAN. SKIP-NAME-RESOLVE

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL Versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL Versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2020-01-06 (versi 2.08.4) (Dihentikan)

Versi: 2.08.4

Aurora MySQL 2.08.4 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Anda dapat meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.10.0. Untuk klaster yang menjalankan Aurora MySQL versi 1, Anda dapat meningkatkan klaster Aurora MySQL 1.23 atau yang lebih tinggi yang sudah ada langsung ke 2.10.0. Anda juga dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.10.0.

Untuk membuat klaster dengan Aurora MySQL versi lama, tentukan versi mesin melalui Konsol Manajemen AWS, AWS CLI, atau API Amazon RDS.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan keamanan dan peningkatan umum:

- Memperbaiki masalah keamanan yang terkait dengan integrasi Aurora MySQL dengan Layanan lain AWS seperti Amazon S3, Amazon ML, dan. AWS Lambda
- Memperbaiki masalah di mana nilai kolom TIMESTAMP baris yang sudah ada diperbarui ke stempel waktu terbaru ketika semua kondisi berikut ini terpenuhi: 1. terdapat sebuah pemicu untuk

tabel; 2. INSERT dilakukan pada tabel yang memiliki klausa ON DUPLICATE KEY UPDATE; 3. baris yang disisipkan dapat menyebabkan pelanggaran nilai duplikat dalam indeks UNIQUE atau PRIMARY KEY; dan 4. satu kolom atau lebih adalah tipe data TIMESTAMP dan memiliki nilai default CURRENT_TIMESTAMP.

- Memperbaiki masalah yang, dalam kondisi yang sangat jarang, menyebabkan instans penulis memulai ulang ketika pemeriksaan integritas data dalam memori gagal.
- Memperbaiki masalah pada kueri paralel yang dapat menyebabkan basis data memulai ulang ketika mengeksekusi pernyataan SQL dengan klausa LIMIT.

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL Versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL Versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2020-11-12 (versi 2.08.3) (Dihentikan)

Versi: 2.08.3

Aurora MySQL 2.08.3 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Anda dapat meningkatkan klaster basis data Aurora MySQL 2.* yang ada langsung ke Aurora MySQL 2.08.3. Anda dapat meningkatkan klaster Aurora MySQL 1.* yang ada secara langsung ke 2.07.3 atau yang lebih tinggi, lalu langsung meningkatkan ke 2.08.3.

Untuk membuat cluster dengan versi lama dari Aurora MySQL, tentukan versi mesin melalui,, atau RDS Konsol Manajemen AWS API. AWS CLI

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan keamanan:

Perbaikan dan penyempurnaan lain untuk penanganan fine-tune di lingkungan terkelola. Di bawah ini adalah beberapa perbaikan CVE tambahan:

- CVE-2020-14567
- CVE-2020-14559
- CVE-2020-14553
- CVE-2020-14547
- CVE-2020-14540
- CVE-2020-2812
- CVE-2020-2806
- CVE-2020-2780
- CVE-2020-2765
- CVE-2020-2763
- CVE-2020-2760
- CVE-2020-2579

Perubahan tidak kompatibel:

Versi ini memperkenalkan perubahan izin yang memengaruhi perilaku perintah `mysqldump`. Pengguna harus memiliki hak istimewa `PROCESS` untuk mengakses tabel `INFORMATION_SCHEMA.FILES`. Untuk menjalankan perintah `mysqldump` tanpa perubahan apa pun, berikan hak istimewa `PROCESS` untuk pengguna basis data yang terhubung dengan perintah `mysqldump`. Anda juga dapat menjalankan perintah `mysqldump` dengan opsi `--no-tablespaces`.

Dengan opsi tersebut, output `mysqldump` tidak menyertakan pernyataan `CREATE LOGFILE GROUP` atau `CREATE TABLESPACE` apa pun. Dalam hal ini, perintah `mysqldump` tidak mengakses tabel `INFORMATION_SCHEMA.FILES`, dan Anda tidak perlu memberikan izin `PROCESS`.

Integrasi perbaikan bug MySQL Community Edition

- Bug #23762382 - PENYISIPAN KUERI NILAI DENGAN SAMBUNGAN DALAM PILIHAN MENYEBABKAN PERILAKU YANG SALAH.
- Bug #25209512 - CURRENT_TIMESTAMP MENGHASILKAN NOL SAAT DIPICU.

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL Versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL Versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2020-08-28 (versi 2.08.2) (Dihentikan)

Versi: 2.08.2

Aurora MySQL 2.08.2 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Anda dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.08.2. Anda juga memiliki opsi untuk meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.08.2. Anda tidak dapat meningkatkan klaster Aurora MySQL 1.* yang ada secara langsung ke 2.08.2; namun, Anda dapat memulihkan snapshot-nya ke Aurora MySQL 2.08.2. Untuk informasi selengkapnya tentang pemulihan snapshot, lihat Memulihkan dari snapshot klaster DB di Panduan Pengguna Amazon Aurora.

Untuk membuat cluster dengan versi lama dari Aurora MySQL, tentukan versi mesin melalui,, atau RDS Konsol Manajemen AWS API. AWS CLI

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan penting:

- Memperbaiki masalah yang mungkin menyebabkan pemadaman yang tidak direncanakan dan memengaruhi ketersediaan basis data.

Perbaikan ketersediaan:

- Memperbaiki masalah di mana basis data Aurora MySQL dapat memulai ulang jika berupa replika binlog dan mereplikasi peristiwa DDL melalui tabel `time_zone` mysql.

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL Versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL Versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.

- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2020-06-18 (versi 2.08.1) (Dihentikan)

Versi: 2.08.1

Aurora MySQL 2.08.1 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Anda dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.08.1. Anda juga memiliki opsi untuk meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.08.1. Anda tidak dapat meningkatkan klaster Aurora MySQL 1.* yang ada secara langsung ke 2.08.1; namun, Anda dapat memulihkan snapshot-nya ke Aurora MySQL 2.08.1.

Untuk membuat cluster dengan versi lama dari Aurora MySQL, tentukan versi mesin melalui,, atau RDS Konsol Manajemen AWS API. AWS CLI

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Fitur baru:

- Penerusan penulisan basis data global. Dalam basis data global Aurora, sekarang Anda dapat melakukan operasi tulis tertentu, seperti pernyataan DML, saat terhubung ke klaster sekunder. Operasi tulis diteruskan ke klaster primer, dan perubahan apa pun direplikasi kembali ke klaster sekunder. Untuk informasi selengkapnya, lihat Menggunakan penerusan penulisan dalam basis data global Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan stabilitas umum:

- Memperbaiki masalah saat memulihkan klaster dari Amazon S3 ke Aurora MySQL versi 2.08.0 membutuhkan waktu lebih lama dari yang diperkirakan jika cadangan S3 tidak menyertakan tabel `mysql.host`.

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL Versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL Versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online

- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2020-06-02 (versi 2.08.0) (Dihentikan)

Versi: 2.08.0

Aurora MySQL 2.08.0 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Anda dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.08.0. Anda juga memiliki opsi untuk meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.08.0. Anda tidak dapat meningkatkan klaster Aurora MySQL 1.* yang ada secara langsung ke 2.08.0; namun, Anda dapat memulihkan snapshot-nya ke Aurora MySQL 2.08.0.

Untuk membuat cluster dengan versi lama dari Aurora MySQL, tentukan versi mesin melalui,, atau RDS Konsol Manajemen AWS API. AWS CLI

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Fitur baru:

- Meningkatkan pemrosesan log biner (binlog) untuk mengurangi waktu pemulihan crash dan latensi waktu commit ketika terdapat transaksi yang sangat besar.
- Peluncuran fitur Aliran Aktivitas Basis Data (DAS) untuk Aurora MySQL. Fitur ini menyediakan aliran near-real-time data aktivitas database dalam database relasional Anda untuk membantu Anda memantau aktivitas. Untuk informasi selengkapnya, lihat Memantau Amazon Aurora dengan Aliran Aktivitas Basis Data di Panduan Pengguna Amazon Aurora.
- Memperbarui file zona waktu untuk mendukung perubahan zona waktu Brasil terbaru.
- Memperkenalkan kata kunci baru di SQL untuk menjalankan fungsionalitas bergabung hash untuk tabel and/or bagian dalam tabel tertentu:`HASH_JOIN`,`HASH_JOIN_PROBING`, dan. `HASH_JOIN_BUILDING` Untuk detail tambahan, lihat Petunjuk Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Memperkenalkan dukungan petunjuk join order di Aurora MySQL 5.7 dengan melakukan backporting fitur MySQL 8.0. Petunjuk baru tersebut adalah `JOIN_FIXED_ORDER`, `JOIN_ORDER`, `JOIN_PREFIX`, dan `JOIN_SUFFIX`. Untuk dokumentasi mendetail tentang dukungan petunjuk join order, lihat WL#9158: Petunjuk join order.
- Aurora Machine Learning kini mendukung fungsi yang ditentukan pengguna dengan `MEDIUMINT` sebagai jenis pengembalian.
- Prosedur tersimpan `lambda_async()` sekarang mendukung semua karakter `utf8` MySQL.

Perbaikan prioritas tinggi:

- Memperbaiki masalah yang dapat menyebabkan instans DB pembaca mengembalikan hasil yang tidak lengkap untuk kueri FTS setelah tabel `INFORMATION_SCHEMA.INNODB_SYS_TABLES` dikueri pada instans DB penulis.
- CVE-2019-5443
- CVE-2019-3822

Perbaikan ketersediaan:

- Memperbaiki masalah yang menyebabkan basis data memulai ulang setelah pernyataan multi-kueri yang mengakses beberapa tabel atau basis data dijalankan dengan cache kueri diaktifkan.
- Memperbaiki kondisi balapan di manajer kunci yang mengakibatkan basis data memulai ulang atau melakukan failover selama rollback transaksi.

- Memperbaiki masalah yang memicu basis data memulai ulang atau melakukan failover ketika beberapa koneksi mencoba untuk memperbarui tabel yang sama dengan indeks Pencarian Teks Lengkap.
- Memperbaiki masalah yang dapat memicu basis data memulai ulang atau melakukan failover selama perintah `kill session`. Jika Anda mengalami masalah ini, hubungi AWS dukungan untuk mengaktifkan perbaikan ini pada instans Anda.
- Memperbaiki masalah yang menyebabkan instans DB pembaca memulai ulang selama transaksi multi-pernyataan dengan beberapa pernyataan `SELECT` dan beban kerja tulis yang berat pada instans DB penulis dengan `AUTOCOMMIT` diaktifkan.
- Memperbaiki masalah yang menyebabkan instans DB pembaca memulai ulang setelah mengeksekusi kueri yang berlangsung lama sementara instans DB penulis berada di bawah beban kerja tulis OLTP yang berat.

Perbaikan umum:

- Memperbaiki waktu pemulihan basis data dan latensi commit untuk transaksi yang berlangsung lama saat binlog diaktifkan.
- Memperbaiki algoritma untuk menghasilkan statistik yang lebih baik untuk memperkirakan jumlah nilai yang berbeda pada kolom yang diindeks, termasuk kolom dengan distribusi data yang tidak simetris.
- Mengurangi waktu respons dan penggunaan CPU kueri sambungan yang mengakses tabel sementara MyISAM dan hasilnya tumpah ke penyimpanan lokal.
- Memperbaiki masalah yang menghalangi snapshot Aurora MySQL 5.6 dengan nama basis data atau tabel yang berisi spasi untuk dipulihkan ke klaster Aurora MySQL 5.7 baru.
- Menyertakan informasi transaksi korban ketika deadlock diselesaikan dalam `show engine innodb status`.
- Memperbaiki masalah yang menyebabkan koneksi terhenti ketika klien dari berbagai versi terhubung ke basis data yang sama dan mengakses cache kueri.
- Memperbaiki kebocoran memori yang diakibatkan oleh beberapa invokasi alur kerja Zero-Downtime Patch (ZDP) atau Zero-Downtime Restart (ZDR) selama masa pakai instans basis data.
- Memperbaiki pesan kesalahan dalam operasi Zero-Downtime Patch (ZDP) atau Zero-Downtime Restart (ZDR) yang salah menyatakan bahwa transaksi terakhir dibatalkan jika tanda auto-commit dinonaktifkan.

- Memperbaiki masalah dalam operasi Zero-Downtime Patch (ZDP) yang dapat menyebabkan pesan kesalahan kegagalan server saat memulihkan variabel sesi pengguna dalam proses basis data baru.
- Memperbaiki masalah dalam operasi Zero Downtime Patch (ZDP) yang dapat menyebabkan kegagalan basis data yang tidak pasti saat terdapat kueri yang berlangsung lama selama patching.
- Memperbaiki masalah saat kueri termasuk fungsi Machine Learning Aurora mengembalikan pesan kesalahan kosong karena respons kesalahan yang ditangani secara tidak benar dari layanan Machine Learning seperti Amazon AI dan SageMaker Amazon Comprehend.
- Memperbaiki masalah dalam fungsionalitas out-of-memory pemantauan yang tidak menghormati nilai kustom `table_definition_cache` parameter.
- Muncul pesan kesalahan "Pelaksanaan kueri terganggu" jika kueri Aurora Machine Learning terganggu. Sebelumnya, yang muncul adalah pesan umum "Kesalahan internal dalam memproses permintaan ML".
- Memperbaiki masalah yang dapat menyebabkan pekerja binlog mengalami waktu koneksi habis ketika parameter `slave_net_timeout` lebih sedikit daripada parameter `aurora_binlog_replication_max_yield_seconds` dan terdapat beban kerja yang rendah pada klaster induk binlog.
- Memperbaiki pemantauan kemajuan pemulihan binlog dengan mengeluarkan pesan informasi dalam log kesalahan pada frekuensi satu pesan per menit.
- Memperbaiki masalah yang dapat menyebabkan transaksi aktif tidak dilaporkan oleh kueri `SHOW ENGINE INNODB STATUS`.

Integrasi perbaikan bug MySQL Community Edition

- Bug #25289359: Kunci cache teks lengkap yang diambil saat data disinkronkan tidak dilepaskan jika ukuran cache teks lengkap melebihi batas ukuran cache teks lengkap.
- Bug #29138644: Mengubah waktu sistem secara manual saat server MySQL sedang berjalan menyebabkan penundaan thread pembersih halaman.
- Bug #25222337: Nama bidang kolom virtual NULL dalam indeks virtual menyebabkan server keluar selama perbandingan nama bidang yang terjadi saat pengisian kolom virtual yang dipengaruhi oleh batasan kunci asing.
- Bug #25053286: Menjalankan prosedur tersimpan dengan kueri yang mengakses suatu tampilan dapat mengalokasikan memori yang tidak dibebaskan hingga sesi berakhir.

- Bug #25586773: Menjalankan prosedur tersimpan dengan pernyataan yang menciptakan tabel dari konten pernyataan SELECT tertentu dapat menyebabkan kebocoran memori.
- Bug #28834208: Selama penerapan log, setelah operasi OPTIMIZE TABLE, InnoDB tidak mengisi kolom virtual sebelum memeriksa pembaruan indeks kolom virtual.
- Bug #26666274: Infinite loop dalam kontainer buffer skema kinerja karena overflow integer 32-bit yang tidak ditandai.

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL Versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL Versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin database Aurora MySQL 2023-08-15 (versi 2.07.10) (Usang)

Versi: 2.07.10

Aurora MySQL 2.07.10 tersedia secara umum. Aurora MySQL versi 2.07 kompatibel dengan MySQL 5.7.12. Untuk informasi selengkapnya tentang perubahan komunitas, lihat Perubahan di MySQL 5.7.12 (2016-04-11, Ketersediaan Umum).

Rilis Aurora MySQL yang didukung saat ini adalah 2.07.*, 2.11.*, 3.01.*, 3.02.*, 3.03.*, dan 3.04.*.

Anda dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.07.10. Anda juga memiliki opsi untuk meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.07.10. Peningkatan in-place tersedia untuk klaster Aurora MySQL 1.* ke Aurora MySQL 2.* (lihat Peningkatan versi dari Aurora MySQL 1.x ke 2.x). Peningkatan tersebut juga tersedia untuk klaster Aurora MySQL 2.* ke Aurora MySQL 3.* (lihat Peningkatan versi dari Aurora MySQL 2.x ke 3.x).

Segera setelah peningkatan versi mesin secara in-place ke Aurora MySQL 2.07.10 dilakukan, peningkatan sistem operasi diterapkan secara otomatis ke semua instans yang terpengaruh pada kelas instans DB db.r4, db.r5, db.t2, dan db.t3, jika instans-instans tersebut menjalankan versi sistem operasi lama. Dalam klaster DB Multi-AZ, semua instans pembaca menerapkan peningkatan sistem operasi terlebih dahulu. Ketika peningkatan sistem operasi pada instans pembaca pertama selesai, failover terjadi dan instans penulis sebelumnya ditingkatkan.

> **Note**
>
> Peningkatan sistem operasi tidak diterapkan secara otomatis ke basis data global Aurora selama peningkatan versi utama.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Memperbaiki masalah keamanan dan CVEs tercantum di bawah ini:

Perbaikan dan penyempurnaan lain untuk penanganan fine-tune di lingkungan terkelola. Di bawah ini adalah beberapa perbaikan CVE tambahan:

- CVE-2023-21963
- CVE-2023-21912
- CVE-2023-0215
- CVE-2022-43551
- CVE-2022-37434
- Memperbaiki masalah di mana peristiwa yang dilaporkan saat memproses rotasi log audit mungkin tidak ditulis ke log audit.
- Memperbarui cipher SSL default yang digunakan oleh Aurora MySQL untuk mengecualikan nilai DES- -SHA yang kurang aman dari parameter database SSL_CIPHER. CBC3 Jika Anda mengalami masalah koneksi SSL karena penghapusan cipher DES- CBC3 -SHA, silakan gunakan cipher aman yang berlaku dari informasi berikut, Mengonfigurasi cipher suite untuk koneksi ke cluster DB MySQL Aurora.
- OpenSSL telah ditingkatkan ke versi 1.0.2zh.

Perbaikan umum:

- Menambahkan dukungan untuk cipher SSL ECDHE-RSA yang menggunakan ukuran kunci yang lebih kecil untuk enkripsi.
- Memperbaiki masalah manajemen memori saat mengeksekusi kueri dengan sambungan hash.

Fitur yang tidak didukung di Aurora MySQL versi 2

Fitur-fitur berikut saat ini tidak didukung di Aurora MySQL versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP).
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Pernyataan SQL `CREATE TABLESPACE`
- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Protokol X

Pembaruan mesin database Aurora MySQL 2023-05-04 (versi 2.07.9) (Usang)

Versi: 2.07.9

Aurora MySQL 2.07.9 tersedia secara umum. Aurora MySQL versi 2.07 kompatibel dengan MySQL 5.7.12. Untuk informasi selengkapnya tentang perubahan komunitas, lihat Perubahan di MySQL 5.7.12 (2016-04-11, Ketersediaan Umum).

Rilis Aurora MySQL yang didukung saat ini adalah 2.07.*, 2.11.*, 3.01.*, 3.02.*, dan 3.03.*.

Anda dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.07.9. Anda juga memiliki opsi untuk meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.07.9. Peningkatan in-place tersedia untuk klaster Aurora MySQL 1.* ke Aurora MySQL 2.* (lihat Peningkatan versi dari Aurora MySQL 1.x ke 2.x). Peningkatan tersebut juga tersedia untuk klaster Aurora MySQL 2.* ke Aurora MySQL 3.* (lihat Peningkatan versi dari Aurora MySQL 2.x ke 3.x).

Segera setelah peningkatan versi mesin secara in-place ke Aurora MySQL 2.07.9 dilakukan, peningkatan sistem operasi diterapkan secara otomatis ke semua instans yang terpengaruh pada kelas instans DB db.r4, db.r5, db.t2, dan db.t3, jika instans-instans tersebut menjalankan versi sistem operasi lama. Dalam klaster DB Multi-AZ, semua instans pembaca menerapkan peningkatan sistem operasi terlebih dahulu. Ketika peningkatan sistem operasi pada instans pembaca pertama selesai, failover terjadi dan instans penulis sebelumnya ditingkatkan.

> **Note**
>
> Peningkatan sistem operasi tidak diterapkan secara otomatis ke basis data global Aurora selama peningkatan versi utama.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Memperbaiki masalah keamanan dan CVEs tercantum di bawah ini:

Perbaikan dan penyempurnaan lain untuk penanganan fine-tune di lingkungan terkelola. Di bawah ini adalah beberapa perbaikan CVE tambahan:

- CVE-2022-32221

Perbaikan ketersediaan:

- Memperbaiki masalah di mana rotasi log Audit Lanjutan dapat mengurangi memori yang dapat dilepas, yang dapat menyebabkan instans DB memulai ulang.
- Memperbaiki masalah yang dapat terjadi selama mulai ulang basis data dan mengakibatkan basis data tidak berhasil dimulai untuk durasi yang lebih lama.

Perbaikan umum:

- Memperbaiki masalah yang, dalam kondisi yang sangat jarang, dapat menyebabkan instans memulai ulang saat volume basis data meningkat ke kelipatan 160GB.

Fitur yang tidak didukung di Aurora MySQL versi 2

Fitur-fitur berikut saat ini tidak didukung di Aurora MySQL versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP).
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2022-06-16 (versi 2.07.8) (Dihentikan)

Versi: 2.07.8

Aurora MySQL 2.07.8 tersedia secara umum. Aurora MySQL versi 2.* kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.* kompatibel dengan MySQL 5.6.

> **Note**
>
> Versi ini ditetapkan sebagai rilis dukungan jangka panjang (LTS). Untuk informasi selengkapnya, lihat Rilis dukungan jangka panjang (LTS) Aurora MySQL di Panduan Pengguna Amazon Aurora.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Anda dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.07.8. Anda juga memiliki opsi untuk meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.07.8. Peningkatan in-place tersedia untuk klaster Aurora MySQL 1.* ke Aurora MySQL 2.* (lihat Peningkatan versi dari Aurora MySQL 1.x ke 2.x). Peningkatan tersebut juga tersedia untuk klaster Aurora MySQL 2.* ke Aurora MySQL 3.* (lihat Peningkatan versi dari Aurora MySQL 2.x ke 3.x).

Untuk membuat cluster dengan versi lama dari Aurora MySQL, tentukan versi mesin melalui,, atau RDS Konsol Manajemen AWS API. AWS CLI

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan keamanan:

Perbaikan dan penyempurnaan lain untuk penanganan fine-tune di lingkungan terkelola. Di bawah ini adalah beberapa perbaikan CVE tambahan:

- CVE-2022-21245
- CVE-2021-36222
- CVE-2021-22926

Perbaikan umum:

- Memperbaiki masalah yang, dalam kasus yang sangat jarang, menyebabkan server basis data memulai ulang ketika thread pendeteksi deadlock terhenti karena kondisi balapan.

Integrasi perbaikan bug MySQL Community Edition

- Ketika UPDATE memerlukan tabel sementara yang memiliki kunci primer lebih besar dari 1024 byte dan tabel tersebut dibuat menggunakan InnoDB, server dapat keluar. (Bug #25153670)

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL Versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL Versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online

- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2021-11-24 (versi 2.07.7) (Dihentikan)

Versi: 2.07.7

Aurora MySQL 2.07.7 tersedia secara umum. Aurora MySQL versi 2.* kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.* kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Anda dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.07.7. Anda juga memiliki opsi untuk meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.07.7. Anda tidak dapat meningkatkan klaster Aurora MySQL 1.* yang ada secara langsung ke 2.07.7; namun, Anda dapat memulihkan snapshot-nya ke Aurora MySQL 2.07.7.

Untuk membuat cluster dengan versi lama dari Aurora MySQL, tentukan versi mesin melalui,, atau RDS Konsol Manajemen AWS API. AWS CLI

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan keamanan:

Perbaikan dan penyempurnaan lain untuk penanganan fine-tune di lingkungan terkelola. Di bawah ini adalah beberapa perbaikan CVE tambahan:

- CVE-2019-17543
- CVE-2019-2960

Perbaikan umum:

- Memperbaiki masalah keamanan yang terkait dengan integrasi Aurora MySQL dengan Layanan lain AWS seperti Amazon S3, Amazon ML, Lambda.
- Memperbaiki masalah terkait pelaporan yang salah tentang lag replikasi Aurora.
- Memperbaiki masalah yang dapat menyebabkan instans basis data gagal memulai ulang ketika basis data tersebut memiliki kombinasi pengguna dan hak istimewa dalam jumlah besar.
- Memperbaiki masalah yang dapat menyebabkan tabel general_log dan slow_log menjadi tidak dapat diakses setelah peningkatan versi utama secara in-place dari Aurora MySQL 1.x (berdasarkan MySQL 5.6) ke Aurora MySQL 2.x (berdasarkan MySQL 5.7).
- Memperbaiki masalah yang, dalam kondisi yang sangat jarang, dapat menyebabkan instans pembaca memulai ulang karena pemrosesan pemeriksaan yang salah.
- Memperbaiki masalah yang, dalam kondisi yang sangat jarang, menunjukkan bagan "Pemuatan Basis Data" di sesi Wawasan Kinerja (PI) sebagai aktif menggunakan CPU meskipun sesi telah selesai diproses dan sedang tidak aktif.
- Memperbaiki masalah pada kueri paralel yang dapat menyebabkan basis data memulai ulang ketika mengeksekusi pernyataan SQL dengan klausa LIMIT.
- Memperbaiki masalah di mana nilai kolom TIMESTAMP baris yang ada diperbarui ke stempel waktu terbaru saat semua kondisi berikut terpenuhi: 1. Terdapat sebuah pemicu untuk tabel; 2. INSERT dilakukan pada tabel yang memiliki klausa ON DUPLICATE KEY UPDATE; 3. baris yang disisipkan dapat menyebabkan pelanggaran nilai duplikat dalam indeks UNIQUE atau PRIMARY KEY; dan, 4. satu kolom atau lebih adalah tipe data TIMESTAMP dan memiliki nilai default CURRENT_TIMESTAMP.
- Memperbaiki masalah yang, dalam kondisi yang sangat jarak, dapat menyebabkan instans basis data memulai ulang saat menggunakan transaksi XA di tingkat isolasi READ COMMITTED.

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL Versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL Versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi

- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2021-09-02 (versi 2.07.6) (Dihentikan)

Versi: 2.07.6

Aurora MySQL 2.07.6 tersedia secara umum. Aurora MySQL versi 2.* kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.* kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Anda dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.07.6. Anda juga memiliki opsi untuk meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.07.6. Anda tidak dapat meningkatkan klaster Aurora MySQL 1.* yang ada secara langsung ke 2.07.6; namun, Anda dapat memulihkan snapshot-nya ke Aurora MySQL 2.07.6.

Untuk membuat cluster dengan versi lama dari Aurora MySQL, harap tentukan versi mesin melalui,, atau RDS Konsol Manajemen AWS API. AWS CLI

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Integrasi perbaikan bug MySQL Community Edition

- DIPERLUKAN TERLALU BANYAK WAKTU UNTUK MENYISIPKAN CATATAN UKURAN 644K. (Bug#23031146)

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL Versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL Versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2021-07-06 (versi 2.07.5) (Dihentikan)

Versi: 2.07.5

Aurora MySQL 2.07.5 tersedia secara umum. Aurora MySQL versi 2.* kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.* kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Anda dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.07.5. Anda juga memiliki opsi untuk meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.07.5. Anda tidak dapat meningkatkan klaster Aurora MySQL 1.* yang ada secara langsung ke 2.07.5; namun, Anda dapat memulihkan snapshot-nya ke Aurora MySQL 2.07.5.

Untuk membuat cluster dengan versi lama dari Aurora MySQL, harap tentukan versi mesin melalui,, atau RDS Konsol Manajemen AWS API. AWS CLI

> **Note**
>
> Versi ini ditetapkan sebagai rilis dukungan jangka panjang (LTS). Untuk informasi selengkapnya, lihat Rilis dukungan jangka panjang (LTS) Aurora MySQL di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan ketersediaan:

- Memperbaiki masalah yang menyebabkan kunci tingkat pengguna tidak diizinkan di Aurora Replica.
- Memperbaiki masalah yang dapat menyebabkan mulai ulang basis data saat menggunakan transaksi XA di tingkat isolasi `READ COMMITTED`.

- Menambah panjang maksimum yang diizinkan hingga 2000 untuk parameter global `server_audit_incl_users` dan `server_audit_excl_users`.

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL Versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL Versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB

- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2021-03-04 (versi 2.07.4) (Dihentikan)

Versi: 2.07.4

Aurora MySQL 2.07.4 tersedia secara umum. Aurora MySQL versi 2.* kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.* kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Anda dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.07.4. Anda juga memiliki opsi untuk meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.07.4. Anda tidak dapat meningkatkan klaster Aurora MySQL 1.* yang ada secara langsung ke 2.07.4; namun, Anda dapat memulihkan snapshot-nya ke Aurora MySQL 2.07.4.

Untuk membuat cluster dengan versi lama dari Aurora MySQL, harap tentukan versi mesin melalui,, atau RDS Konsol Manajemen AWS API. AWS CLI

> **Note**
>
> Versi ini ditetapkan sebagai rilis dukungan jangka panjang (LTS). Untuk informasi selengkapnya, lihat Rilis dukungan jangka panjang (LTS) Aurora MySQL di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan keamanan:

- CVE-2020-14812
- CVE-2020-14793
- CVE-2020-14790
- CVE-2020-14790
- CVE-2020-14769
- CVE-2020-14765
- CVE-2020-14760
- CVE-2020-14672
- CVE-2020-1971

Perbaikan ketersediaan:

- Memperbaiki masalah yang dapat menyebabkan klien macet jika terjadi kesalahan jaringan saat membaca atau menulis paket jaringan.
- Mempercepat waktu mulai ulang mesin dalam beberapa kasus setelah DDL terganggu.
- Memperbaiki masalah di mana DDL atau DML dapat menyebabkan mulai ulang mesin selama permintaan prefetch halaman.
- Memperbaiki masalah di mana replika dapat memulai ulang saat melakukan pemindaian terbalik table/index pada Replika Baca Aurora.
- Memperbaiki masalah dalam operasi klaster klon yang dapat menyebabkan klon memakan waktu lebih lama.
- Memperbaiki masalah yang dapat menyebabkan mulai ulang basis data saat menggunakan optimasi kueri paralel untuk kolom geospasial.
- Memperbaiki masalah yang menyebabkan replika binlog berhenti dengan kesalahan `HA_ERR_KEY_NOT_FOUND`.

Integrasi perbaikan bug MySQL Community Edition

- Memperbaiki masalah dalam pengurai ngram Teks lengkap saat menangani token yang berisi ' ' (spasi), '%', atau ','. Pelanggan harus membangun kembali indeks FTS mereka jika menggunakan pengurai ngram. (Bug #25873310)
- Memperbaiki masalah yang dapat menyebabkan mulai ulang mesin selama eksekusi kueri dengan tampilan SQL bersarang. (Bug #27214153, Bug #26864199)

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL Versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL Versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2020-11-10 (versi 2.07.3) (Dihentikan)

Versi: 2.07.3

Aurora MySQL 2.07.3 tersedia secara umum. Aurora MySQL versi 2.* kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.* kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Anda dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.07.3. Anda juga memiliki opsi untuk meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.07.3. Anda tidak dapat meningkatkan klaster Aurora MySQL 1.* yang ada secara langsung ke 2.07.3; namun, Anda dapat memulihkan snapshot-nya ke Aurora MySQL 2.07.3.

Untuk membuat cluster dengan versi lama dari Aurora MySQL, harap tentukan versi mesin melalui,, atau RDS Konsol Manajemen AWS API. AWS CLI

> **Note**
>
> Versi ini ditetapkan sebagai rilis dukungan jangka panjang (LTS). Untuk informasi selengkapnya, lihat Rilis dukungan jangka panjang (LTS) Aurora MySQL di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan keamanan:

Perbaikan dan penyempurnaan lain untuk penanganan fine-tune di lingkungan terkelola.

- CVE-2021-2144
- CVE-2020-14567
- CVE-2020-14559
- CVE-2020-14553
- CVE-2020-14547
- CVE-2020-14540
- CVE-2020-2812
- CVE-2020-2806
- CVE-2020-2780
- CVE-2020-2765
- CVE-2020-2763
- CVE-2020-2760
- CVE-2020-2579
- CVE-2019-2740

Perubahan tidak kompatibel:

Versi ini memperkenalkan perubahan izin yang memengaruhi perilaku perintah `mysqldump`. Pengguna harus memiliki hak istimewa `PROCESS` untuk mengakses tabel `INFORMATION_SCHEMA.FILES`. Untuk menjalankan perintah `mysqldump` tanpa perubahan apa pun, berikan hak istimewa `PROCESS` untuk pengguna basis data yang terhubung dengan perintah `mysqldump`. Anda juga dapat menjalankan perintah `mysqldump` dengan opsi `--no-tablespaces`. Dengan opsi tersebut, output `mysqldump` tidak menyertakan pernyataan `CREATE LOGFILE GROUP` atau `CREATE TABLESPACE` apa pun. Dalam hal ini, perintah `mysqldump` tidak mengakses tabel `INFORMATION_SCHEMA.FILES`, dan Anda tidak perlu memberikan izin `PROCESS`.

Perbaikan ketersediaan:

- Memperbaiki kondisi balapan di pengelola kunci antara pembunuhan a connection/query dan penghentian sesi yang mengakibatkan database dimulai ulang.
- Memperbaiki masalah yang menyebabkan basis data memulai ulang setelah pernyataan multi-kueri yang mengakses beberapa tabel atau basis data dijalankan dengan cache kueri diaktifkan.
- Memperbaiki masalah yang mungkin menyebabkan mulai ulang beberapa kali karena pembaruan kolom virtual dengan indeks sekunder.

Integrasi perbaikan bug MySQL Community Edition

- InnoDB: Transaksi XA bersamaan yang berhasil berjalan ke tahap persiapan XA pada master bertentangan saat diputar ulang pada slave, sehingga tunggu kunci mengalami waktu habis di thread pengaplikasi. Pertentangan tersebut disebabkan oleh rentang kunci GAP yang berbeda ketika transaksi diputar ulang secara seri pada slave. Untuk mencegah jenis pertentangan ini, kunci GAP yang diambil oleh transaksi XA pada tingkat isolasi READ COMMITTED sekarang dilepaskan (dan sudah tidak diwariskan) saat transaksi XA mencapai tahap persiapan. (Bug #27189701, Bug #25866046)
- InnoDB: Kunci gap diambil secara tidak perlu selama validasi kunci asing saat menggunakan tingkat isolasi READ COMMITTED. (Bug #25082593)
- Replikasi: Saat menggunakan transaksi XA, jika tunggu kunci mengalami waktu habis atau deadlock terjadi untuk thread pengaplikasi (SQL) pada slave replikasi, coba ulang otomatis tidak berfungsi. Penyebabnya adalah meskipun thread SQL akan melakukan rollback, thread tersebut tidak akan melakukan rollback transaksi XA. Ini berarti ketika transaksi dicoba kembali, peristiwa pertama adalah XA START yang tidak valid karena transaksi XA sudah berlangsung, yang menyebabkan kesalahan XAER_RMFAIL. (Bug #24764800)
- Replikasi: Transaksi yang berseling terkadang dapat menyebabkan deadlock pada pengaplikasi slave saat tingkat isolasi transaksi diatur ke REPEATABLE READ. (Bug #25040331)
- Replikasi: Nilai dikembalikan oleh pernyataan SHOW SLAVE STATUS untuk total ukuran gabungan dari semua file log relay yang ada (Relay_Log_Space) dapat menjadi jauh lebih besar daripada ruang disk yang digunakan oleh file log relay tersebut. I/O Utas tidak mengunci variabel saat memperbarui nilainya, sehingga utas SQL dapat secara otomatis menghapus file log relai dan menulis nilai yang dikurangi sebelum I/O utas selesai memperbarui nilainya. Thread I/O kemudian menulis perhitungan ukuran aslinya, dengan mengabaikan pembaruan thread SQL dan oleh karena itu menambahkan kembali ruang untuk file yang dihapus. Nilai Relay_Log_Space sekarang

dikunci selama pembaruan untuk mencegah pembaruan bersamaan dan memastikan perhitungan yang akurat. (Bug #26997096, Bug #87832)

- Untuk pernyataan INSERT yang menerima pembuatan nilai oleh daftar VALUES untuk baris kedua atau setelahnya menggunakan subkueri yang berisi sambungan, server dapat keluar setelah gagal menyelesaikan hak istimewa yang diperlukan. (Bug #23762382)
- Untuk tabel dengan kolom TIMESTAMP atau DATETIME yang memiliki nilai default CURRENT_TIMESTAMP, kolom tersebut dapat diinisialisasi ke `0000-00-00 00:00:00` jika tabel memiliki pemicu `BEFORE INSERT`. (Bug #25209512, Bug #84077)
- Keluarnya server dapat disebabkan oleh upaya serentak beberapa thread yang mendaftarkan dan membatalkan pendaftaran objek Skema Kinerja metadata. (Bug #26502135)
- Menjalankan prosedur tersimpan yang berisi pernyataan yang menciptakan tabel dari konten pernyataan SELECT tertentu dapat menyebabkan kebocoran memori. (Bug #25586773)
- Menjalankan prosedur tersimpan dengan kueri yang mengakses suatu tampilan dapat mengalokasikan memori yang tidak dibebaskan hingga sesi berakhir. (Bug #25053286)
- Kasus materialisasi subkueri tertentu dapat menyebabkan keluarnya server. Kueri ini sekarang menghasilkan pesan kesalahan yang menunjukkan bahwa materialisasi dinonaktifkan. (Bug #26402045)
- Kueri dengan banyak sambungan tersisa akan lambat jika buffering sambungan digunakan (misalnya, menggunakan algoritma loop bersarang blok). (Bug #18898433, Bug #72854)
- Pengoptimal melewatkan kolom kedua dalam indeks komposit saat mengeksekusi sambungan bagian dalam dengan klausa `LIKE` di kolom kedua. (Bug #26402045)

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL Versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL Versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.

- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2020-04-17 (versi 2.07.2) (Dihentikan)

Versi: 2.07.2

Aurora MySQL 2.07.2 tersedia secara umum. Aurora MySQL versi 2.* kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.* kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Anda dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.07.2. Anda juga memiliki opsi untuk meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.07.2. Anda tidak dapat meningkatkan klaster Aurora MySQL 1.* yang ada secara langsung ke 2.07.2; namun, Anda dapat memulihkan snapshot-nya ke Aurora MySQL 2.07.2.

Untuk membuat cluster dengan versi lama dari Aurora MySQL, harap tentukan versi mesin melalui,, atau RDS Konsol Manajemen AWS API. AWS CLI

> **Note**
>
> Versi ini ditetapkan sebagai rilis dukungan jangka panjang (LTS). Untuk informasi selengkapnya, lihat Rilis dukungan jangka panjang (LTS) Aurora MySQL di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan keamanan:

- CVE-2016-8287
- CVE-2016-5634

Perbaikan prioritas tinggi:

- Memperbaiki masalah yang menyebabkan kloning memakan waktu lebih lama di beberapa klaster basis data dengan beban tulis yang tinggi.
- Memperbaiki masalah yang dapat menyebabkan kueri pada instans DB pembaca dengan rencana eksekusi yang menggunakan indeks sekunder mengembalikan data yang belum di-commit.

Masalah ini terkait dengan data yang terpengaruh oleh operasi bahasa manipulasi data (DML) yang memodifikasi kolom kunci indeks primer atau sekunder.

Perbaikan umum:

- Memperbaiki masalah yang mengakibatkan pemulihan lambat klaster DB Aurora 1.x yang mengandung indeks FTS (Pencarian Teks Penuh) ke klaster DB Aurora 2.x.
- Memperbaiki masalah yang menyebabkan pemulihan yang lebih lambat pada snapshot basis data Aurora 1.x yang mengandung karakter khusus dalam tabel yang dipisah dengan huruf kecil pada nama tabel ke klaster DB Aurora 2.x.
- Memperbaiki masalah yang menyebabkan kesalahan saat melakukan kueri log kueri lambat dan log umum dalam instans DB pembaca.

Integrasi perbaikan bug MySQL Community Edition

- Bug #23104498: Memperbaiki masalah dalam Skema Kinerja dalam melaporkan total memori yang digunakan. (https://github.com/mysql/mysql-server/commit/20b6840df5452f47313c6f9a6ca075bfbc00a96b)
- Bug #22551677: Memperbaiki masalah di Skema Kinerja yang dapat menyebabkan kerusakan mesin basis data saat mencoba untuk melakukannya secara offline.(https://github.com/mysql/mysql-server/commit/05e2386eccd32b6b444b900c9f8a87a1d8d531e9)
- Bug #23550835, Bug #23298025, Bug #81464: Memperbaiki masalah di Skema Kinerja yang menyebabkan crash pada mesin basis data karena melebihi kapasitas buffer internal. (https://github.com/mysql/mysql-server/commit/b4287f93857bf2f99b18fd06f555bbe5b12debfc)

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL Versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL Versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.

- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2019-12-23 (versi 2.07.1) (Dihentikan)

Versi: 2.07.1

Aurora MySQL 2.07.1 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Anda dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.07.1. Anda juga memiliki opsi untuk meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.07.1. Anda tidak dapat meningkatkan klaster Aurora MySQL 1.* yang ada secara langsung ke 2.07.1; namun, Anda dapat memulihkan snapshot-nya ke Aurora MySQL 2.07.1.

Untuk membuat cluster dengan versi lama dari Aurora MySQL, harap tentukan versi mesin melalui,, atau RDS Konsol Manajemen AWS API. AWS CLI

> **Note**
>
> Versi ini saat ini tidak tersedia di AWS Wilayah berikut: AWS GovCloud (AS-Timur) [us-gov-east-1], (AS-Barat) [-1], Tiongkok AWS GovCloud (Ningxia) [cn-barat laut us-gov-west 1], Asia Pasifik (Hong Kong) [ap-east-1], dan Timur Tengah (Bahrain) [me-south-1]. Akan ada pengumuman terpisah setelah tersedia.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan prioritas tinggi:

- Memperbaiki kebocoran memori yang lambat di sistem pelacakan basis data dan pencatatan log subsistem Aurora tertentu yang menurunkan memori yang dapat dilepas.

Perbaikan Stabilitas Umum:

- Memperbaiki kegagalan selama eksekusi kueri kompleks yang melibatkan sambungan multitabel dan agregasi yang menggunakan tabel menengah secara internal.

Perbandingan dengan Aurora MySQL Versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL Versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL Versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok

- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2019-11-25 (versi 2.07.0) (Dihentikan)

Versi: 2.07.0

Aurora MySQL 2.07.0 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Anda dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.07.0. Anda juga memiliki opsi untuk meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.07.0. Anda tidak dapat meningkatkan klaster Aurora MySQL 1.* yang ada secara langsung ke 2.07.0; namun, Anda dapat memulihkan snapshot-nya ke Aurora MySQL 2.07.0.

Untuk membuat cluster dengan versi lama dari Aurora MySQL, harap tentukan versi mesin melalui,, atau RDS Konsol Manajemen AWS API. AWS CLI

> **Note**
>
> Versi ini saat ini tidak tersedia di AWS Wilayah berikut: AWS GovCloud (AS-Timur) [us-gov-east-1], (AS-Barat) [-1], Tiongkok AWS GovCloud (Ningxia) [cn-barat laut us-gov-west 1], Asia Pasifik (Hong Kong) [ap-east-1], Timur Tengah (Bahrain) [me-south-1], dan Amerika Selatan (São Paulo) [sa-sa-east-1]. Akan ada pengumuman terpisah setelah tersedia.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Fitur baru:

- Database Global sekarang memungkinkan penambahan wilayah replika hanya-baca sekunder untuk kluster basis data yang digunakan di Wilayah ini: AWS wilayah: AS Timur (Virginia N.) [us-timur-1], AS Timur (Ohio) [us-timur-2], AS Barat (California N.) [us-west-1], AS Barat (Oregon) [us-west-2], Eropa (Irlandia) [eu-west-1], Eropa (London) [eu-west-2], Eropa (Paris) [eu-west-3], Asia Pasifik (Tokyo) [ap-northeast-1], Asia Pasifik (Tokyo) [ap-northeast-1], Asia Pasifik (Seoul) [ap-northeast-2], Asia Pasifik (Singapura) [ap-southeast-1], Asia Pasifik (Sydney) [ap-southeast-2], Kanada (Tengah) [ca-central-1], Eropa (Frankfurt) [eu-central-1], dan Asia Pasifik (Mumbai) [ap-south-1].
- Pembelajaran mesin Amazon Aurora adalah integrasi yang sangat dioptimalkan antara database Aurora MySQL dan layanan pembelajaran mesin (ML). AWS Machine learning Aurora memungkinkan developer untuk menambahkan berbagai prediksi berbasis ML ke aplikasi basis data mereka dengan menginvokasi model ML menggunakan bahasa pemrograman SQL yang sudah mereka gunakan untuk pengembangan basis data, tanpa harus membangun integrasi khusus atau mempelajari alat terpisah. Untuk informasi selengkapnya, lihat Menggunakan kemampuan machine learning (ML) dengan Amazon Aurora.
- Menambahkan Support untuk ANSI `READ COMMITTED` tingkat isolasi pada replika yang dibaca. Tingkat isolasi ini memungkinkan kueri yang berlangsung lama pada replika baca untuk mengeksekusi tanpa memengaruhi throughput tulis yang tinggi pada simpul penulis. Untuk informasi selengkapnya, lihat Tingkat isolasi Aurora MySQL.

Perbaikan penting:

- CVE-2019-2922
- CVE-2019-2923
- CVE-2019-2924
- CVE-2019-2910

Perbaikan prioritas tinggi:

- Memperbaiki masalah dalam pemulihan DDL yang menghasilkan waktu henti basis data yang lama. Klaster yang menjadi tidak tersedia setelah menjalankan pernyataan drop multitabel, misalnya `DROP TABLE t1, t2, t3`, harus diperbarui ke versi ini.
- Memperbaiki masalah dalam pemulihan DDL yang menghasilkan waktu henti basis data yang lama. Klaster yang menjadi tidak tersedia setelah mengeksekusi pernyataan DDL `INPLACE ALTER TABLE` harus diperbarui ke versi ini.

Perbaikan stabilitas umum:

- Memperbaiki masalah yang menghasilkan data yang tidak konsisten di `information_schema.replica_host_status` yang sesuai.

Integrasi perbaikan bug MySQL Community Edition

- Bug #26251621: PERILAKU YANG SALAH DENGAN TRIGGER DAN GCOL
- Bug #22574695: ASSERTION `!TABLE || (!TABLE->READ_SET || BITMAP_IS_SET(TABLE->READ_SET, FIEL
- Bug #25966845: INSERT PADA KUNCI DUPLIKAT MENGHASILKAN DEADLOCK
- Bug #23070734: TRUNCATE TABLES BERSAMAAN MENYEBABKAN MACET
- Bug #26191879: FOREIGN KEY CASCADES MENGGUNAKAN MEMORI BERLEBIHAN
- Bug #20989615: INNODB AUTO_INCREMENT MENGHASILKAN NILAI YANG SAMA DUA KALI

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL Versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL Versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Aurora MySQL 2.07.0 kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Aurora MySQL 2.07.0 saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2019-11-22 (versi 2.06.0) (Dihentikan)

Versi: 2.06.0

Aurora MySQL 2.06.0 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang didukung saat ini adalah 1.14.*, 1.15.*, 1.16.*, 1.17.*, 1.18.*, 1.19.*, 2.01.*, 2.02.*, 2.03.*, 2.04.*, 2.05.*, dan 2.06.*.

Anda dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.06.0. Anda juga memiliki opsi untuk meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.06.0. Anda tidak dapat meningkatkan klaster Aurora MySQL 1.* yang ada secara langsung ke 2.06.0; namun, Anda dapat memulihkan snapshot-nya ke Aurora MySQL 2.06.0.

Untuk membuat cluster dengan versi lama dari Aurora MySQL, harap tentukan versi mesin melalui,, atau RDS Konsol Manajemen AWS API. AWS CLI

> **Note**
>
> Versi ini saat ini tidak tersedia di AWS Wilayah berikut: AWS GovCloud (AS-Timur) [us-gov-east-1], (AS-Barat) [-1], Tiongkok AWS GovCloud (Ningxia) [cn-barat laut us-gov-west 1], Asia Pasifik (Hong Kong) [ap-east-1], dan Timur Tengah (Bahrain) [me-south-1]. Akan ada pengumuman terpisah setelah tersedia.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Fitur baru:

- Klaster MySQL Aurora kini mendukung jenis instans db.r5.8xlarge, db.r5.16xlarge, dan db.r5.24xlarge. Untuk informasi selengkapnya tentang jenis instans untuk klaster Aurora MySQL, lihat Kelas-kelas instans basis data Aurora dalam Panduan Pengguna Amazon Aurora.

- Fitur hash join sudah tersedia secara umum dan mode lab Aurora tidak perlu ditetapkan ke ON. Fitur ini dapat meningkatkan performa kueri saat Anda harus menggabungkan data dalam jumlah besar menggunakan equi-join. Untuk informasi selengkapnya tentang penggunaan fitur ini, lihat Menggunakan API Data untuk Aurora Serverless di Panduan Pengguna Amazon Aurora.
- Fitur hot row contention sudah tersedia secara umum dan mode lab Aurora tidak perlu diatur ke ON. Fitur ini meningkatkan secara signifikan throughput untuk beban kerja dengan banyak transaksi yang bersaing untuk baris di halaman yang sama.
- Aurora MySQL 2.06 dan yang lebih tinggi mendukung tindakan "memundurkan" klaster DB ke waktu tertentu, tanpa memulihkan data dari cadangan. Fitur ini, yang dikenal sebagai Backtrack, memberikan cara yang cepat untuk memulihkan dari kesalahan pengguna, seperti menghapus tabel yang salah atau menghapus baris yang salah. Backtrack selesai dalam hitungan detik, bahkan untuk basis data besar. Baca AWS blog untuk ikhtisar, dan lihat Backtracking cluster Aurora DB di Panduan Pengguna Amazon Aurora untuk detail selengkapnya.
- Aurora 2.06 dan yang lebih tinggi mendukung AWS Lambda pemanggilan sinkron melalui fungsi asli. `lambda_sync()` Fungsi asli `lambda_async()` juga tersedia, yang dapat digunakan sebagai alternatif untuk prosedur tersimpan yang ada untuk invokasi Lambda asinkron. Untuk informasi tentang pemanggilan fungsi Lambda, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan penting:

Tidak ada.

Perbaikan prioritas tinggi:

Perbaikan keamanan

- CVE-2019-2805
- CVE-2019-2791
- CVE-2019-2778
- CVE-2019-2758
- CVE-2019-2739
- CVE-2019-2730
- CVE-2018-3064
- CVE-2018-3058
- CVE-2018-2786

- CVE-2017-3653
- CVE-2017-3465
- CVE-2017-3455
- CVE-2017-3244
- CVE-2016-5612

Penanganan koneksi

- Ketersediaan basis data telah ditingkatkan untuk melayani lonjakan koneksi klien dengan lebih baik saat menjalankan satu atau lebih. DDLs Perbaikan ini ditangani dengan cara menciptakan thread tambahan sementara ketika diperlukan. Anda disarankan untuk melakukan peningkatan versi jika basis data menjadi tidak responsif setelah terjadi lonjakan pada koneksi saat memproses DDL.

Mulai ulang mesin

- Memperbaiki masalah ketidaktersediaan berkepanjangan saat memulai ulang mesin. Perbaikan ini mengatasi masalah dalam inisialisasi pool buffer. Masalah ini sangat jarang terjadi tetapi dapat berpotensi memengaruhi rilis apa pun yang didukung.
- Memperbaiki masalah yang menyebabkan basis data yang dikonfigurasi sebagai master log biner (binlog) memulai ulang selama beban kerja tulis yang berat sedang berjalan.

Perbaikan stabilitas umum:

- Melakukan perbaikan di mana kueri yang mengakses data yang tidak tersimpan dalam cache menjadi lebih lambat dari biasanya. Pelanggan yang mengalami latensi baca yang lebih tinggi tanpa alasan yang jelas saat mengakses data yang tidak tersimpan dalam cache diimbau untuk melakukan peningkatan versi karena mereka mungkin mengalami masalah ini.
- Memperbaiki masalah kegagalan pemulihan tabel terpartisi dari snapshot basis data. Pelanggan yang mengalami kesalahan saat mengakses tabel terpartisi dalam basis data yang telah dipulihkan dari snapshot basis data Aurora MySQL 1.* disarankan untuk menggunakan versi ini.
- Meningkatkan stabilitas Replika Aurora dengan memperbaiki perselisihan kunci antara thread yang melayani kueri baca dan yang menerapkan perubahan skema ketika kueri DDL sedang berlangsung pada instans DB penulis.
- Memperbaiki masalah stabilitas yang terkait dengan pembaruan tabel `mysql.innodb_table_stats` yang dipicu oleh operasi DDL.

- Memperbaiki masalah kesalahan pelaporan `ERROR 1836` ketika kueri bersarang dijalankan terhadap tabel sementara di Replika Aurora.

Perbaikan kinerja:

- Memperbaiki kinerja replikasi binlog dengan mencegah panggilan API yang tidak perlu ke cache jika cache kueri tersebut dinonaktifkan pada pekerja binlog.

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL Versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL Versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Aurora MySQL 2.06.0 kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Aurora MySQL 2.06.0 saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB

- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2019-11-11 (versi 2.05.0) (Dihentikan)

Versi: 2.05.0

Aurora MySQL 2.05.0 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang didukung saat ini adalah 1.14.*, 1.15.*, 1.16.*, 1.17.*, 1.18.*, 1.19.*, 2.01.*, 2.02.*, 2.03.*, dan 2.04.*.

Anda dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.05.0. Anda juga memiliki opsi untuk meningkatkan klaster basis data Aurora MySQL 2.* yang ada, hingga 2.04.6, ke Aurora MySQL 2.05.0. Anda tidak dapat meningkatkan klaster Aurora MySQL 1.* yang ada secara langsung ke 2.05.0; namun, Anda dapat memulihkan snapshot-nya ke Aurora MySQL 2.05.0.

Untuk membuat cluster dengan versi lama dari Aurora MySQL, harap tentukan versi mesin melalui,, atau RDS Konsol Manajemen AWS API. AWS CLI

> **Note**
>
> Versi ini saat ini tidak tersedia di AWS Wilayah berikut: AWS GovCloud (AS-Timur) [us-gov-east-1], (AS-Barat) [-1], Tiongkok AWS GovCloud (Ningxia) [cn-barat laut us-gov-west 1], Asia Pasifik (Hong Kong) [ap-east-1], Eropa (Stockholm) [eu-north-1], dan Timur Tengah (Bahrain) [me-south-1] me-south-1]. Akan ada pengumuman terpisah setelah tersedia.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan penting:

- CVE-2018-0734
- CVE-2019-2534
- CVE-2018-3155
- CVE-2018-2612
- CVE-2017-3599
- CVE-2018-3056
- CVE-2018-2562
- CVE-2017-3329
- CVE-2018-2696
- Memperbaiki masalah yang menyebabkan peristiwa dalam file binlog saat ini pada master tidak direplikasi pada pekerja jika nilai parameter `sync_binlog` tidak diatur ke 1.

Perbaikan prioritas tinggi:

- Pelanggan dengan ukuran basis data yang mendekati 64 tebibyte (TiB) sangat disarankan untuk melakukan peningkatan ke versi ini untuk menghindari waktu henti akibat bug stabilitas yang memengaruhi volume yang mendekati batas penyimpanan Aurora.
- Nilai default dari parameter `aurora_binlog_replication_max_yield_seconds` telah diubah menjadi nol untuk mencegah peningkatan lah replikasi demi mendukung kinerja kueri depan pada master binlog.

Integrasi perbaikan bug MySQL

- Bug #23054591: PURGE BINARY LOGS TO sedang membaca seluruh file binlog dan menyebabkan Stall MySql

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL Versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL Versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Aurora MySQL 2.05.0 kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Aurora MySQL 2.05.0 saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman

- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2020-08-14 (versi 2.04.9) (Dihentikan)

Versi: 2.04.9

Aurora MySQL 2.04.9 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Untuk membuat cluster dengan versi lama dari Aurora MySQL, harap tentukan versi mesin melalui,, atau RDS Konsol Manajemen AWS API. AWS CLI

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Versi ini saat ini tidak tersedia di AWS Wilayah berikut: AWS GovCloud (AS-Timur) [us-gov-east-1], AWS GovCloud (AS-Barat) [us-gov-west-1], Asia Pasifik (Hong Kong) [ap-east-1], dan Timur Tengah (Bahrain) [me-south-1]. Akan ada pengumuman terpisah setelah tersedia.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan prioritas tinggi:

Perbaikan penting:

- CVE-2020-2760
- CVE-2019-5443
- CVE-2019-3822
- CVE-2019-2924
- CVE-2019-2923
- CVE-2019-2922
- CVE-2019-2911
- CVE-2019-2910
- CVE-2019-2805
- CVE-2019-2791
- CVE-2019-2778
- CVE-2019-2758
- CVE-2019-2740
- CVE-2019-2739
- CVE-2019-2730
- CVE-2019-2628
- CVE-2018-3064
- CVE-2018-3058
- CVE-2018-2813
- CVE-2018-2786
- CVE-2017-3653

- CVE-2017-3465
- CVE-2017-3464
- CVE-2017-3455
- CVE-2017-3244
- CVE-2016-5612
- CVE-2016-5436

Perbaikan ketersediaan:

- Memperbaiki masalah yang dapat menyebabkan mulai ulang atau failover basis data karena eksekusi perintah `kill session`. Jika Anda mengalami masalah ini, hubungi AWS dukungan untuk mengaktifkan perbaikan ini pada instans Anda.
- Memperbaiki masalah yang menyebabkan mulai ulang basis data selama eksekusi kueri kompleks yang melibatkan sambungan multitabel dan agregasi yang menggunakan tabel menengah secara internal.
- Memperbaiki masalah yang menyebabkan mulai ulang basis data karena `DROP TABLE` yang terganggu pada beberapa tabel.
- Memperbaiki masalah yang menyebabkan failover basis data selama pemulihan basis data.
- Memperbaiki mulai ulang basis data yang disebabkan oleh pelaporan threads_running yang salah saat log audit dan kueri lambat diaktifkan.
- Memperbaiki masalah di mana perintah `kill query` mungkin terhenti selama eksekusi.
- Memperbaiki kondisi balapan di manajer kunci yang mengakibatkan basis data memulai ulang atau melakukan failover selama rollback transaksi.
- Memperbaiki masalah yang memicu basis data memulai ulang atau melakukan failover ketika beberapa koneksi mencoba untuk memperbarui tabel yang sama dengan indeks Pencarian Teks Lengkap.
- Memperbaiki masalah yang dapat menyebabkan deadlatch saat menghapus indeks yang mengakibatkan failover atau mulai ulang.

Perbaikan umum:

- Memperbaiki masalah yang dapat menyebabkan kueri pada replika baca untuk menggunakan data dari transaksi yang tidak di-commit. Masalah ini terbatas pada transaksi yang dimulai segera setelah basis data memulai ulang.

- Memperbaiki masalah yang dihadapi selama `INPLACE ALTER TABLE` untuk tabel dengan pemicu yang ditentukan dan saat DDL tidak berisi klausa `RENAME`.
- Memperbaiki masalah yang menyebabkan kloning memakan waktu lebih lama di beberapa klaster basis data dengan beban tulis yang tinggi.
- Memperbaiki masalah yang dihadapi selama peningkatan saat tabel yang dipartisi memiliki ruang yang disematkan di dalam nama.
- Memperbaiki masalah replika baca yang mungkin melihat sekilas hasil parsial transaksi yang baru saja diterapkan pada penulis.
- Memperbaiki masalah di mana kueri pada replika baca terhadap tabel FTS dapat memberikan hasil yang usang. Ini hanya akan terjadi ketika kueri FTS pada replika baca benar-benar mengikuti kueri pada `INFORMATION_SCHEMA.INNODB_SYS_TABLES` untuk tabel FTS yang sama di penulis.
- Memperbaiki masalah yang mengakibatkan pemulihan lambat klaster basis data Aurora 1.x yang mengandung indeks FTS (Pencarian Teks Penuh) ke klaster basis data Aurora 2.x.
- Menambah panjang maksimum yang diizinkan hingga 2000 untuk parameter global `server_audit_incl_users` dan `server_audit_excl_users`.
- Memperbaiki masalah yang menyebabkan pemulihan Aurora 1.x ke Aurora 2.x membutuhkan waktu lebih lama.
- Memperbaiki masalah yang menyebabkan invokasi `lambda_async` melalui prosedur yang disimpan tidak berfungsi dengan Unicode.
- Memperbaiki masalah yang ditemui saat indeks spasial tidak menangani kolom geometri off-record dengan benar.
- Memperbaiki masalah yang mungkin menyebabkan kueri gagal pada instans DB pembaca dengan kesalahan `InternalFailureException` dengan pesan "Operation terminated (internal error)".

Integrasi perbaikan bug MySQL

- Bug #23070734, Bug #80060: Pemotongan Bersamaan menyebabkan macet TABLEs
- Bug #23103937: PS_TRUNCATE_ALL_TABLES() TIDAK DAPAT DIGUNAKAN DALAM MODE SUPER_READ_ONLY
- Bug#22551677: Saat membuat server offline, kondisi balapan dalam Skema Kinerja dapat menyebabkan server keluar.
- Bug #27082268: Sinkronisasi sync FTS tidak valid.

- BUG #12589870: Memperbaiki masalah yang menyebabkan mulai ulang dengan pernyataan multikueri saat cache kueri diaktifkan.
- Bug #26402045: Kasus materialisasi subkueri tertentu dapat menyebabkan keluarnya server. Kueri ini sekarang menghasilkan pesan kesalahan yang menunjukkan bahwa materialisasi dinonaktifkan.
- Bug #18898433: Kueri dengan banyak sambungan tersisa akan lambat jika buffering sambungan digunakan (misalnya, menggunakan algoritma loop bersarang blok).
- Bug #25222337: Nama bidang kolom virtual NULL dalam indeks virtual menyebabkan server keluar selama perbandingan nama bidang yang terjadi saat pengisian kolom virtual yang dipengaruhi oleh batasan kunci asing. (https://github.com/mysql/mysql-server/commit/273d5c9d7072c63b6c47dbef6963d7dc491d5131)
- Bug #25053286: Menjalankan prosedur tersimpan dengan kueri yang mengakses suatu tampilan dapat mengalokasikan memori yang tidak dibebaskan hingga sesi berakhir. (https://github.com/mysql/mysql-server/commit/d7b37d4d141a95f577916448650c429f0d6e193d)
- Bug #25586773: Menjalankan prosedur tersimpan yang berisi pernyataan yang membuat tabel dari isi pernyataan SELECT (https://dev.mysql.com/doc/refman/5.7/en/select.html) tertentu dapat mengakibatkan kebocoran memori. (https://github.com/mysql/mysql-server/commit/88301e5adab65f6750f66af284be410c4369d0c1)
- Bug #26666274: LOOP TANPA HENTI DI KONTAINER BUFFER SKEMA KINERJA.
- Bug #23550835, Bug #23298025, Bug #81464: Tabel Skema Kinerja SELECT ketika buffer internal penuh dapat menyebabkan server keluar.

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL Versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL Versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.

- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Aurora MySQL 2.04.9 kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Aurora MySQL 2.05.0 saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2019-11-20 (versi 2.04.8) (Dihentikan)

Versi: 2.04.8

Aurora MySQL 2.04.8 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Anda dapat memulihkan snapshot rilis Aurora MySQL 2.* mana pun ke Aurora MySQL 2.04.8. Anda juga memiliki opsi untuk meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.04.8.

Untuk membuat cluster dengan versi lama dari Aurora MySQL, harap tentukan versi mesin melalui,, atau RDS Konsol Manajemen AWS API. AWS CLI

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Versi ini saat ini tidak tersedia di AWS Wilayah berikut: AWS GovCloud (AS-Timur) [us-gov-east-1], AWS GovCloud (AS-Barat) [us-gov-west-1], Asia Pasifik (Hong Kong) [ap-east-1], dan Timur Tengah (Bahrain) [me-south-1]. Akan ada pengumuman terpisah setelah tersedia.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Fitur baru:

- Perbaikan replika baca:
  - Mengurangi lalu lintas jaringan dari instans penulis dengan mengirimkan data secara efisien ke instans pembaca dalam klaster DB Aurora. Perbaikan ini diaktifkan secara default, karena membantu mencegah replika tertinggal di belakang dan memulai ulang. Parameter untuk fitur ini adalah `aurora_enable_repl_bin_log_filtering`.
  - Mengurangi lalu lintas jaringan dari instans penulis ke instans pembaca dalam klaster DB Aurora menggunakan kompresi. Perbaikan ini diaktifkan secara default untuk kelas instans 8xlarge dan

16xlarge, karena kedua instans ini dapat menoleransi overhead CPU tambahan untuk kompresi. Parameter untuk fitur ini adalah `aurora_enable_replica_log_compression`.

Perbaikan prioritas tinggi:

- Memperbaiki manajemen memori dalam instans penulis Aurora yang mencegah pengaktifan ulang penulis akibat kondisi kehabisan memori selama beban kerja berat dengan adanya instans pembaca di dalam klaster DB Aurora.
- Perbaikan untuk kondisi acak di dalam penjadwal yang menghasilkan mulai ulang mesin saat mengakses objek skema kinerja secara bersamaan.

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL Versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL Versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Aurora MySQL 2.04.8 kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan

implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Aurora MySQL 2.04.8 saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2019-11-14 (versi 2.04.7) (Dihentikan)

Versi: 2.04.7

Aurora MySQL 2.04.7 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Anda dapat memulihkan snapshot dari rilis Aurora MySQL yang saat ini didukung ke Aurora MySQL 2.04.7. Anda juga memiliki opsi untuk meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.04.7. Anda tidak dapat meningkatkan klaster Aurora MySQL 1.* yang ada secara langsung ke 2.04.7; namun, Anda dapat memulihkan snapshot-nya ke Aurora MySQL 2.04.7.

Untuk membuat cluster dengan versi lama dari Aurora MySQL, harap tentukan versi mesin melalui,, atau RDS Konsol Manajemen AWS API. AWS CLI

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Versi ini saat ini tidak tersedia di AWS Wilayah berikut: AWS GovCloud (AS-Timur) [us-gov-east-1], AWS GovCloud (AS-Barat) [us-gov-west-1], Asia Pasifik (Hong Kong) [ap-east-1], dan Timur Tengah (Bahrain) [me-south-1]. Akan ada pengumuman terpisah setelah tersedia.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan prioritas tinggi:

Penanganan Koneksi

- Ketersediaan basis data telah ditingkatkan untuk melayani lonjakan koneksi klien dengan lebih baik saat menjalankan satu atau lebih. DDLs Perbaikan ini ditangani dengan cara menciptakan thread tambahan sementara ketika diperlukan. Anda disarankan untuk melakukan peningkatan versi jika basis data menjadi tidak responsif setelah terjadi lonjakan pada koneksi saat memproses DDL.
- Memperbaiki masalah yang menghasilkan nilai yang tidak tepat untuk variabel status global `Threads_running`.

Mulai Ulang Mesin

- Memperbaiki masalah ketidaktersediaan berkepanjangan saat memulai ulang mesin. Perbaikan ini mengatasi masalah dalam inisialisasi pool buffer. Masalah ini sangat jarang terjadi tetapi dapat berpotensi memengaruhi rilis apa pun yang didukung.

Perbaikan stabilitas umum:

- Melakukan perbaikan di mana kueri yang mengakses data yang tidak tersimpan dalam cache menjadi lebih lambat dari biasanya. Pelanggan yang mengalami latensi baca yang lebih tinggi tanpa alasan yang jelas saat mengakses data yang tidak tersimpan dalam cache diimbau untuk melakukan peningkatan versi karena mereka mungkin mengalami masalah ini.

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL Versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL Versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Aurora MySQL 2.04.7 kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Aurora MySQL 2.04.7 saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman

- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2019-09-19 (versi 2.04.6) (Dihentikan)

Versi: 2.04.6

Aurora MySQL 2.04.6 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Anda memiliki opsi untuk meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.04.6. Kami tidak mengizinkan peningkatan in-place klaster Aurora MySQL 1.*. Pembatasan ini akan dicabut pada rilis Aurora MySQL 2.* selanjutnya. Anda dapat memulihkan snapshot Aurora MySQL 1.14.*, 1.15.*, 1.16.*, 1.17.*, 1.18.*, 1.19.*, 2.01.*, 2.02.*, 2.03.* and 2.04.* ke Aurora MySQL 2.04.6.

Untuk menggunakan versi lama Aurora MySQL, Anda dapat membuat cluster database baru dengan menentukan versi mesin melalui, API, Konsol Manajemen AWS atau Amazon RDS. AWS CLI

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Versi ini saat ini tidak tersedia di AWS Wilayah berikut: Eropa (London) [eu-west-2] AWS GovCloud , (AS-Timur) [-1], (AS-Barat) us-gov-east [-1], Tiongkok AWS GovCloud (Ningxia)

[cn-barat laut - us-gov-west 1], dan Asia Pasifik (Hong Kong) [ap-east-1]. Akan ada pengumuman terpisah setelah tersedia.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

- Memperbaiki masalah di mana peristiwa dalam file binlog di master saat ini tidak direplikasi pada pekerja jika nilai parameter `sync_binlog` tidak diatur ke 1.
- Nilai default dari parameter `aurora_binlog_replication_max_yield_seconds` telah diubah menjadi nol untuk mencegah peningkatan lah replikasi demi mendukung kinerja kueri depan pada master binlog.

Integrasi perbaikan bug MySQL

- Bug #23054591: PURGE BINARY LOGS TO sedang membaca seluruh file binlog dan menyebabkan Stall MySql

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL Versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL Versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.

- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Aurora MySQL 2.04.6 kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Aurora MySQL 2.04.6 saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2019-07-08 (versi 2.04.5) (Dihentikan)

Versi: 2.04.5

Aurora MySQL 2.04.5 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Anda memiliki opsi untuk meningkatkan klaster basis data Aurora MySQL 2.* yang ada ke Aurora MySQL 2.04.5. Kami tidak mengizinkan peningkatan in-place klaster Aurora MySQL 1.*. Pembatasan ini akan dicabut pada rilis Aurora MySQL 2.* selanjutnya. Anda dapat memulihkan snapshot Aurora MySQL 1.14.*, 1.15.*, 1.16.*, 1.17.*, 1.18.*, 1.19.*, 2.01.*, 2.02.*, 2.03.* and 2.04.* ke Aurora MySQL 2.04.5.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan keamanan:

- CVE-2016-3518

Perbaikan umum:

- Memperbaiki kondisi race selama pertumbuhan volume penyimpanan yang menyebabkan basis data memulai ulang.
- Memperbaiki kegagalan komunikasi internal selama pembukaan volume yang menyebabkan basis data memulai ulang.
- Menambahkan dukungan pemulihan DDL untuk `ALTER TABLE ALGORITHM=INPLACE` di tabel yang dipartisi.

- Memperbaiki masalah pemulihan DDL `ALTER TABLE ALGORITHM=COPY` yang menyebabkan basis data memulai ulang.
- Meningkatkan stabilitas Replika Aurora di bawah beban kerja yang sering terhapus di penulis.
- Memperbaiki mulai ulang basis data yang disebabkan oleh deadlatch antara thread yang menjalankan sinkronisasi indeks pencarian teks penuh dan thread yang menjalankan pengosongan tabel pencarian teks penuh dari cache kamus.
- Memperbaiki masalah stabilitas pada pekerja binlog selama replikasi DDL ketika koneksi ke master binlog tidak stabil.
- Memperbaiki out-of-memory masalah dalam kode pencarian teks lengkap yang menyebabkan database dimulai ulang.
- Memperbaiki masalah di Penulis Aurora yang menyebabkannya memulai ulang ketika seluruh volume 64 tebibyte (TiB) digunakan.
- Memperbaiki kondisi race di fitur Skema Kinerja yang menyebabkan basis data memulai ulang.
- Memperbaiki masalah yang menyebabkan koneksi yang dibatalkan saat menangani kesalahan dalam manajemen protokol jaringan.

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL Versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL Versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).

- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Aurora MySQL 2.04.5 kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Aurora MySQL 2.04.5 saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2019-05-29 (versi 2.04.4) (Dihentikan)

Versi: 2.04.4

Aurora MySQL 2.04.4 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Saat membuat klaster DB Aurora MySQL baru (termasuk memulihkan snapshot), Anda memiliki opsi untuk memilih kompatibilitas dengan MySQL 5.7 atau MySQL 5.6. Kami tidak mengizinkan peningkatan in-place klaster Aurora MySQL 1.* atau pemulihan klaster Aurora MySQL 1.* dari backup Amazon S3 ke Aurora MySQL 2.04.4. Kami berencana menghapus batasan ini di rilis Aurora MySQL 2.* mendatang.

Anda dapat memulihkan snapshot Aurora MySQL 1.14.*, 1.15.*, 1.16.*, 1.17.*, 1.18.*, 1.19.*, 2.01.*, 2.02.*, 2.03.*, and 2.04.* ke Aurora MySQL 2.04.4.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Versi ini saat ini tidak tersedia di Wilayah AWS GovCloud (AS-Barat) [us-gov-west-1], Eropa (Stockholm) [eu-north-1], Tiongkok (Ningxia) [cn-barat laut 1], dan Asia Pasifik (Hong Kong) [ap-east-1]. AWS Akan ada pengumuman terpisah setelah tersedia.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

- Memperbaiki masalah yang dapat menyebabkan kegagalan saat memuat data ke Aurora dari S3.
- Memperbaiki masalah yang dapat menyebabkan kegagalan saat mengunggah data dari Aurora ke S3.
- Memperbaiki masalah yang menyebabkan koneksi yang dibatalkan saat menangani kesalahan dalam manajemen protokol jaringan.
- Memperbaiki masalah yang dapat menyebabkan crash saat menangani tabel yang dipartisi.
- Memperbaiki masalah pada fitur Wawasan Kinerja yang tidak tersedia di beberapa wilayah.

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL Versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL Versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Aurora MySQL 2.04.4 kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Aurora MySQL 2.04.4 saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online

- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2019-05-09 (versi 2.04.3) (Dihentikan)

Versi: 2.04.3

Aurora MySQL 2.04.3 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Saat membuat klaster DB Aurora MySQL baru (termasuk memulihkan snapshot), Anda memiliki opsi untuk memilih kompatibilitas dengan MySQL 5.7 atau MySQL 5.6. Kami tidak mengizinkan peningkatan in-place klaster Aurora MySQL 1.* atau pemulihan klaster Aurora MySQL 1.* dari backup Amazon S3 ke Aurora MySQL 2.04.3. Kami berencana menghapus batasan ini di rilis Aurora MySQL 2.* mendatang.

Anda dapat memulihkan snapshot Aurora MySQL 1.14.*, 1.15.*, 1.16.*, 1.17.*, 1.18.*, 1.19.*, 2.01.*, 2.02.*, 2.03.*, and 2.04.* ke Aurora MySQL 2.04.3.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Versi ini saat ini tidak tersedia di Wilayah AWS GovCloud (AS-Barat) [us-gov-west-1] dan Tiongkok (Ningxia) [cn-northwest-1]. AWS Akan ada pengumuman terpisah setelah tersedia.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

- Memperbaiki bug dalam replikasi binlog yang dapat menyebabkan masalah pada instans Aurora yang dikonfigurasi sebagai pekerja binlog.
- Memperbaiki out-of-memory kondisi saat menangani rutinitas tersimpan besar.
- Memperbaiki kesalahan dalam menangani jenis perintah `ALTER TABLE`.
- Memperbaiki masalah terkait koneksi yang dibatalkan karena kesalahan dalam manajemen protokol jaringan.

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL Versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL Versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Aurora MySQL 2.04.3 kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Aurora MySQL 2.04.3 saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2019-05-02 (versi 2.04.2) (Dihentikan)

Versi: 2.04.2

Aurora MySQL 2.04.2 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Saat membuat klaster DB Aurora MySQL baru (termasuk memulihkan snapshot), Anda memiliki opsi untuk memilih kompatibilitas dengan MySQL 5.7 atau MySQL 5.6. Kami tidak mengizinkan peningkatan in-place klaster Aurora MySQL 1.* atau pemulihan klaster Aurora MySQL 1.* dari backup Amazon S3 ke Aurora MySQL 2.04.2. Kami berencana menghapus batasan ini di rilis Aurora MySQL 2.* mendatang.

Anda dapat memulihkan snapshot Aurora MySQL 1.14.*, 1.15.*, 1.16.*, 1.17.*, 1.18.*, 1.19.*, 2.01.*, 2.02.*, 2.03.*, 2.04.0, dan 2.04.1 ke Aurora MySQL 2.04.2.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Versi ini saat ini tidak tersedia di Wilayah AWS GovCloud (AS-Barat) [us-gov-west-1] dan Tiongkok (Ningxia) [cn-northwest-1]. AWS Akan ada pengumuman terpisah setelah tersedia.

> **Note**
>
> Untuk informasi tentang cara meningkatkan versi klaster basis data MySQL Aurora Anda, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

- Menambahkan dukungan untuk replikasi binlog SSL menggunakan sertifikat khusus. Untuk informasi tentang penggunaan replikasi binlog SSL di Aurora MySQL, lihat mysql_rds_import_binlog_ssl_material.
- Memperbaiki deadlatch pada instans primer Aurora yang terjadi ketika tabel dengan indeks Pencarian Teks Penuh dioptimalkan.
- Memperbaiki masalah pada Replika Aurora di mana kinerja kueri tertentu yang menggunakan `SELECT(*)` dapat terkena dampak pada tabel yang memiliki indeks sekunder.
- Memperbaiki kondisi yang menghasilkan munculnya Kesalahan 1032.
- Meningkatkan stabilitas Replika Aurora dengan memperbaiki beberapa deadlatch.

Integrasi perbaikan bug MySQL

- Bug #24829050 - OPTIMALISASI INDEX_MERGE_INTERSECTION MENYEBABKAN HASIL KUERI YANG SALAH

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL Versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL Versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Aurora MySQL 2.04.2 kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Aurora MySQL 2.04.2 saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online

- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2019-03-25 (versi 2.04.1) (Dihentikan)

Versi: 2.04.1

Aurora MySQL 2.04.1 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Saat membuat klaster DB Aurora MySQL baru (termasuk memulihkan snapshot), Anda memiliki opsi untuk memilih kompatibilitas dengan MySQL 5.7 atau MySQL 5.6. Kami tidak mengizinkan peningkatan in-place klaster Aurora MySQL 1.* atau pemulihan klaster Aurora MySQL 1.* dari backup Amazon S3 ke Aurora MySQL 2.04.1. Kami berencana menghapus batasan ini di rilis Aurora MySQL 2.* mendatang.

Anda dapat memulihkan snapshot Aurora MySQL 1.14.*, 1.15.*, 1.16.*, 1.17.*, 1.18.*, 1.19.*, 2.01.*, 2.02.*, 2.03.*, 2.04.0 ke Aurora MySQL 2.04.1.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Versi ini saat ini tidak tersedia di wilayah AWS GovCloud (AS-Barat) [us-gov-west-1]. Akan ada pengumuman terpisah setelah tersedia.

> **Note**
>
> Prosedur untuk meningkatkan klaster DB Anda telah berubah. Untuk informasi selengkapnya, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

- Memperbaiki masalah yang menyebabkan snapshot Aurora MySQL 5.6 untuk versi yang lebih rendah dari 1.16 tidak dapat dipulihkan ke klaster Aurora MySQL 5.7 terbaru.

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL Versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL Versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan

implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2019-03-25 (versi 2.04.0) (Dihentikan)

Versi: 2.04

Aurora MySQL 2.04 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Saat membuat klaster DB Aurora MySQL baru (termasuk memulihkan snapshot), Anda memiliki opsi untuk memilih kompatibilitas dengan MySQL 5.7 atau MySQL 5.6. Kami tidak mengizinkan peningkatan in-place klaster Aurora MySQL 1.* atau pemulihan klaster Aurora MySQL 1.* dari backup Amazon S3 ke Aurora MySQL 2.04.0. Kami berencana menghapus batasan ini di rilis Aurora MySQL 2.* mendatang.

Anda dapat memulihkan snapshot Aurora MySQL 1.19.*, 2.01.*, 2.02.*, dan 2.03.* ke Aurora MySQL 2.04.0. Anda tidak dapat memulihkan snapshot Aurora MySQL 1.14.* atau lebih rendah, 1.15.*, 1.16.*, 1.17.*, 1.18.* ke Aurora MySQL 2.04.0. Batasan ini dihapus di Aurora MySQL 2.04.1.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Versi ini saat ini tidak tersedia di wilayah AWS GovCloud (AS-Barat) [us-gov-west-1]. Akan ada pengumuman terpisah setelah tersedia.

> **Note**
>
> Prosedur untuk meningkatkan klaster DB Anda telah berubah. Untuk informasi selengkapnya, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

- Mendukung replikasi berbasis GTID. Untuk informasi tentang penggunaan replikasi berbasis GTID dengan Aurora Aurora MySQL, lihat Menggunakan replikasi berbasis GTID untuk Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Memperbaiki masalah yang menyebabkan Replika Aurora secara tidak tepat memunculkan kesalahan `Running in read-only mode` saat pernyataan yang menghapus atau memperbarui baris dalam tabel sementara mengandung subkueri InnoDB.

Integrasi perbaikan bug MySQL

- Bug #26225783: MYSQL CRASH ON CREATE TABLE (REPRODUCEABLE) -> INNODB: ALONG SEMAPHORE WAIT.

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL Versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL Versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Versi Aurora MySQL ini kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Versi Aurora MySQL ini saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`

Pembaruan mesin basis data Aurora MySQL 2019-02-07 (versi 2.03.4) (Dihentikan)

Versi: 2.03.4

Aurora MySQL 2.03.4 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Saat membuat klaster DB Aurora MySQL baru (termasuk memulihkan snapshot), Anda dapat memilih kompatibilitas dengan MySQL 5.7 atau MySQL 5.6.

Kami tidak mengizinkan peningkatan in-place klaster Aurora MySQL 1.* ke Aurora MySQL 2.03.4 atau pemulihan ke Aurora MySQL 2.03.4 dari backup Amazon S3. Kami berencana untuk menghapus batasan ini di rilis Aurora MySQL 2.* mendatang.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Versi ini saat ini tidak tersedia di wilayah AWS GovCloud (AS-Barat) [us-gov-west-1] dan Tiongkok (Beijing) [cn-north-1]. Akan ada pengumuman terpisah setelah tersedia.

> **Note**
>
> Prosedur untuk meningkatkan klaster DB Anda telah berubah. Untuk informasi selengkapnya, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

- Support untuk pemeriksaan sensitif aksen UTF8 MB4 Unicode 9.0 dan tidak peka huruf besar/kecil,. `utf8mb4_0900_as_ci`

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Aurora MySQL 2.03.4 kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Aurora MySQL 2.03.4 saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Pengidentifikasi transaksi global (GTIDs). Aurora MySQL mendukung GTIDs dalam versi 2.04 dan lebih tinggi.
- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB

- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`
- Protokol X

Pembaruan mesin basis data Aurora MySQL 2019-01-18 (versi 2.03.3) (Dihentikan)

Versi: 2.03.3

Aurora MySQL 2.03.3 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Saat membuat klaster DB Aurora MySQL baru (termasuk memulihkan snapshot), Anda dapat memilih kompatibilitas dengan MySQL 5.7 atau MySQL 5.6.

Kami tidak mengizinkan peningkatan in-place klaster Aurora MySQL 1.* ke Aurora MySQL 2.03.3 atau pemulihan ke Aurora MySQL 2.03.3 dari backup Amazon S3. Kami berencana menghapus batasan ini di rilis Aurora MySQL 2.* mendatang.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Versi ini saat ini tidak tersedia di wilayah AWS GovCloud (AS-Barat) [us-gov-west-1] dan Tiongkok (Beijing) [cn-north-1]. Akan ada pengumuman terpisah setelah tersedia.

> **Note**
>
> Prosedur untuk meningkatkan klaster DB Anda telah berubah. Untuk informasi selengkapnya, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan CVE

- CVE-2016-5436

Perbaikan penting:

- Memperbaiki masalah yang menyebabkan Replika Aurora mengalami deadlatch saat menjalankan pemindaian mundur pada indeks.
- Memperbaiki masalah yang menyebabkan Replika Aurora memulai ulang saat instans primer Aurora menjalankan operasi DDL pada tabel yang dipartisi.
- Memperbaiki masalah yang menyebabkan Replika Aurora memulai ulang selama pembatalan cache kueri setelah operasi DDL pada instans primer Aurora.
- Memperbaiki masalah yang menyebabkan Replika Aurora memulai ulang selama kueri `SELECT` pada tabel sementara instans primer Aurora menjalankan pemenggalan pada tabel tersebut.
- Memperbaiki masalah kesalahan hasil dengan tabel sementara MyISAM yang mengakibatkan hanya kolom terindeks yang dapat diakses.
- Memperbaiki masalah dalam log lambat yang menghasilkan nilai besar yang tidak benar untuk `query_time` dan `lock_time` secara berkala setelah sekitar 40.000 kueri.
- Memperbaiki masalah di mana skema bernama "tmp" dapat menyebabkan migrasi dari RDS for MySQL ke Aurora MySQL terhenti.
- Memperbaiki masalah di mana log audit mungkin melewatkan peristiwa selama rotasi log.
- Memperbaiki masalah yang menyebabkan instans primer Aurora yang dipulihkan dari snapshot Aurora 5.6 mungkin memulai ulang saat fitur DDL Cepat di mode lab diaktifkan.
- Memperbaiki masalah di mana penggunaan CPU menjadi 100% yang disebabkan oleh thread statistik kamus.

- Memperbaiki masalah di mana Replika Aurora dapat memulai ulang saat menjalankan pernyataan `CHECK TABLE`.

Integrasi perbaikan bug MySQL

- Bug #25361251: INCORRECT BEHAVIOR WITH INSERT ON DUPLICATE KEY IN SP
- Bug #26734162: INCORRECT BEHAVIOR WITH INSERT OF BLOB + ON DUPLICATE KEY UPDATE
- Bug #27460607: INCORRECT BEHAVIOR OF IODKU WHEN INSERT SELECT's SOURCE TABLE IS EMPTY
- Kueri yang menggunakan klausa `DISTINCT` atau `GROUP BY` dapat mengembalikan hasil yang salah. (MySQL 5.7 Bug #79591, Bug #22343910)
- Sebuah `DELETE` dari tabel gabungan yang menggunakan tabel turunan di klausa `WHERE` gagal dengan kesalahan 1093 (Bug #23074801).
- GCOLS: INCORRECT BEHAVIOR WITH CHARSET CHANGES (Bug #25287633).

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).

- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Aurora MySQL 2.03.3 kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Aurora MySQL 2.03.3 saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Pengidentifikasi transaksi global (GTIDs). Aurora MySQL mendukung GTIDs dalam versi 2.04 dan lebih tinggi.
- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`
- Protokol X

Pembaruan mesin basis data Aurora MySQL 2019-01-09 (versi 2.03.2) (Dihentikan)

Versi: 2.03.2

Aurora MySQL 2.03.2 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Saat membuat klaster DB Aurora MySQL baru (termasuk memulihkan snapshot), Anda dapat memilih kompatibilitas dengan MySQL 5.7 atau MySQL 5.6.

Kami tidak mengizinkan peningkatan in-place klaster Aurora MySQL 1.* ke Aurora MySQL 2.03.2 atau pemulihan ke Aurora MySQL 2.03.2 dari backup Amazon S3. Kami berencana menghapus batasan ini di rilis Aurora MySQL 2.* mendatang.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Versi ini saat ini tidak tersedia di wilayah AWS GovCloud (AS-Barat) [us-gov-west-1] dan Tiongkok (Beijing) [cn-north-1]. Akan ada pengumuman terpisah setelah tersedia.

> **Note**
>
> Prosedur untuk meningkatkan klaster DB Anda telah berubah. Untuk informasi selengkapnya, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

- Pemilih Versi Aurora – Dimulai dengan Aurora MySQL 2.03.2, Anda dapat memilih dari beberapa versi Aurora yang kompatibel dengan MySQL 5.7 di Konsol Manajemen AWS. Untuk informasi selengkapnya, lihat Memeriksa atau menentukan versi mesin Aurora MySQL melalui AWS di Panduan Pengguna Amazon Aurora.

Perbaikan penting:

- CVE-2016-3495

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Aurora MySQL 2.03.2 kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Aurora MySQL 2.03.2 saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Pengidentifikasi transaksi global (GTIDs). Aurora MySQL mendukung GTIDs dalam versi 2.04 dan lebih tinggi.
- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB

- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`
- Protokol X

Pembaruan mesin basis data Aurora MySQL 2018-10-24 (versi 2.03.1) (Dihentikan)

Versi: 2.03.1

Aurora MySQL 2.03.1 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Saat membuat klaster DB Aurora MySQL baru, Anda dapat memilih kompatibilitas dengan MySQL 5.7 atau MySQL 5.6. Saat memulihkan snapshot yang kompatibel dengan MySQL 5.6, Anda dapat memilih kompatibilitas dengan MySQL 5.7 atau MySQL 5.6.

Anda dapat memulihkan snapshot Aurora MySQL 1.14.*, 1.15.*, 1.16.*, 1.17.*, 1.18.*, 2.01.*, 2.02.*, dan 2.03 ke Aurora MySQL 2.03.1.

Kami tidak mengizinkan peningkatan in-place klaster Aurora MySQL 1.* ke Aurora MySQL 2.03.1 atau pemulihan ke Aurora MySQL 2.03.1 dari backup Amazon S3. Kami berencana menghapus batasan ini di rilis Aurora MySQL 2.* mendatang.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Versi ini saat ini tidak tersedia di wilayah AWS GovCloud (AS-Barat) [us-gov-west-1] dan Tiongkok (Beijing) [cn-north-1]. Akan ada pengumuman terpisah setelah tersedia.

Perbaikan

- Memperbaiki masalah di mana Penulis Aurora mungkin memulai ulang saat menjalankan deteksi deadlock transaksi.

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Aurora MySQL 2.03.1 kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Aurora MySQL 2.03.1 saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Pengidentifikasi transaksi global (GTIDs). Aurora MySQL mendukung GTIDs dalam versi 2.04 dan lebih tinggi.

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`
- Protokol X

Pembaruan mesin basis data Aurora MySQL 2018-10-11 (versi 2.03) (Dihentikan)

Versi: 2.03

Aurora MySQL 2.03 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Saat membuat klaster DB Aurora MySQL baru, Anda dapat memilih kompatibilitas dengan MySQL 5.7 atau MySQL 5.6. Saat memulihkan snapshot yang kompatibel dengan MySQL 5.6, Anda dapat memilih kompatibilitas dengan MySQL 5.7 atau MySQL 5.6.

Anda dapat memulihkan snapshot Aurora MySQL 1.14.*, 1.15.*, 1.16.*, 1.17.*, 1.18.*, 2.01.*, dan 2.02.* ke Aurora MySQL 2.03.

Kami tidak mengizinkan peningkatan in-place klaster Aurora MySQL 1.* ke Aurora MySQL 2.03 atau pemulihan ke Aurora MySQL 2.03 dari backup Amazon S3. Kami berencana menghapus batasan ini di rilis Aurora MySQL 2.* mendatang.

> **Note**
>
> Versi ini saat ini tidak tersedia di wilayah AWS GovCloud (AS-Barat) [us-gov-west-1] dan Tiongkok (Beijing) [cn-north-1]. Akan ada pengumuman terpisah setelah tersedia.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

- Skema kinerja tersedia.
- Memperbaiki masalah yang menyebabkan sesi zombie dengan status dimatikan mungkin menggunakan lebih banyak CPU.
- Memperbaiki masalah deadlatch saat transaksi hanya-baca sedang mendapatkan kunci pada catatan di Penulis Aurora.
- Memperbaiki masalah di mana beban kerja Replika Aurora mungkin memiliki penggunaan CPU yang tinggi.
- Beberapa perbaikan pada masalah yang mungkin menyebabkan Replika Aurora atau penulis Aurora memulai ulang.
- Menambahkan kemampuan untuk melewati pencatatan log diagnostik ketika batas throughput antar disk tercapai.
- Memperbaiki masalah kebocoran memori saat binlog diaktifkan di Penulis Aurora.

Integrasi perbaikan bug MySQL Community Edition

- PINDAI BALIK DI TABEL TERPARTISI MENJALANKAN PERINTAH ICP BERDASARKAN DESC (Bug #24929748).
- JSON_OBJECT MEMBUAT KODE JSON TIDAK VALID (Bug#26867509).
- MEMASUKKAN DATA JSON BESAR MEMBUTUHKAN WAKTU SANGAT LAMA (Bug #22843444).
- TABEL PARTISI MENGGUNAKAN LEBIH BANYAK MEMORI DALAM 5.7 DARIPADA 5.6 (Bug #25080442).

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Aurora MySQL 2.03 kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Aurora MySQL 2.03 saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Pengidentifikasi transaksi global (GTIDs). Aurora MySQL mendukung GTIDs dalam versi 2.04 dan lebih tinggi.
- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi

- Pernyataan SQL `CREATE TABLESPACE`
- Protokol X

Pembaruan mesin basis data Aurora MySQL 2018-10-08 (versi 2.02.5) (Dihentikan)

Versi: 2.02.5

Aurora MySQL 2.02.5 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Saat membuat klaster DB Aurora MySQL baru, Anda dapat memilih kompatibilitas dengan MySQL 5.7 atau MySQL 5.6. Saat memulihkan snapshot yang kompatibel dengan MySQL 5.6, Anda dapat memilih kompatibilitas dengan MySQL 5.7 atau MySQL 5.6.

Anda dapat memulihkan snapshot Aurora MySQL 1.14.*, 1.15.*, 1.16.*, 1.17.*, 1.18.*, 2.01.*, dan 2.02.* ke Aurora MySQL 2.02.5. Anda juga dapat melakukan peningkatan in-place dari Aurora MySQL 2.01.* atau 2.02.* ke Aurora MySQL 2.02.5.

Kami tidak mengizinkan peningkatan in-place klaster Aurora MySQL 1.* ke Aurora MySQL 2.02.5 atau pemulihan ke Aurora MySQL 2.02.5 dari backup Amazon S3. Kami berencana menghapus batasan ini di rilis Aurora MySQL 2.* mendatang.

Skema kinerja dinonaktifkan untuk rilis Aurora MySQL 5.7 ini. Tingkatkan ke Aurora 2.03 untuk dukungan skema kinerja.

> **Note**
>
> Versi ini saat ini tidak tersedia di wilayah AWS GovCloud (AS-Barat) [us-gov-west-1] dan Tiongkok (Beijing) [cn-north-1]. Akan ada pengumuman terpisah setelah tersedia.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

- Memperbaiki masalah di mana Replika Aurora dapat memulai ulang saat melakukan pemindaian balik pada tabel.

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Aurora MySQL 2.02.5 kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Aurora MySQL 2.02.5 saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Pengidentifikasi transaksi global (GTIDs). Aurora MySQL mendukung GTIDs dalam versi 2.04 dan lebih tinggi.

- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`
- Protokol X

Pembaruan mesin basis data Aurora MySQL 2018-09-21 (versi 2.02.4) (Dihentikan)

Versi: 2.02.4

Aurora MySQL 2.02.4 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Saat membuat klaster DB Aurora MySQL baru, Anda dapat memilih kompatibilitas dengan MySQL 5.7 atau MySQL 5.6. Saat memulihkan snapshot yang kompatibel dengan MySQL 5.6, Anda dapat memilih kompatibilitas dengan MySQL 5.7 atau MySQL 5.6.

Anda dapat memulihkan snapshot Aurora MySQL 1.14.*, 1.15.*, 1.16.*, 1.17.*, 1.18.*, 2.01.*, dan 2.02.* ke Aurora MySQL 2.02.4. Anda juga dapat melakukan peningkatan in-place dari Aurora MySQL 2.01.* atau 2.02.* ke Aurora MySQL 2.02.4.

Kami tidak mengizinkan peningkatan in-place klaster Aurora MySQL 1.* ke Aurora MySQL 2.02.4 atau pemulihan ke Aurora MySQL 2.02.4 dari backup Amazon S3. Kami berencana menghapus batasan ini di rilis Aurora MySQL 2.* mendatang.

Skema kinerja dinonaktifkan untuk rilis Aurora MySQL 5.7 ini. Tingkatkan ke Aurora 2.03 untuk dukungan skema kinerja.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

- Memperbaiki masalah stabilitas terkait indeks Pencarian Teks Lengkap pada tabel yang dipulihkan dari snapshot Aurora MySQL 5.6.

Integrasi perbaikan bug MySQL Community Edition

- BUG#13651665 INNODB MAY BE UNABLE TO LOAD TABLE DEFINITION AFTER RENAME
- BUG#21371070 INNODB: CANNOT ALLOCATE 0 BYTES.
- BUG#21378944 FTS ASSERT ENC.SRC_ILIST_PTR != NULL, FTS_OPTIMIZE_WORD(), OPTIMIZE TABLE
- BUG#21508537 ASSERTION FAILURE UT_A(!VICTIM_TRX->READ_ONLY)
- BUG#21983865 UNEXPECTED DEADLOCK WITH INNODB_AUTOINC_LOCK_MODE=0
- BUG#22679185 INVALID INNODB FTS DOC ID DURING INSERT
- BUG#22899305 GCOLS: ASSERTION: !(COL->PRTYPE & 256).
- BUG#22956469 MEMORY LEAK INTRODUCED IN 5.7.8 IN MEMORY/INNODB/OS0FILE
- BUG#22996488 CRASH IN FTS_SYNC_INDEX WHEN DOING DDL IN A LOOP
- BUG#23014521 GCOL:INNODB: ASSERTION: !IS_V
- BUG#23021168 REPLICATION STOPS AFTER TRX IS ROLLED BACK ASYNC
- BUG#23052231 ASSERTION: ADD_AUTOINC < DICT_TABLE_GET_N_USER_COLS
- BUG#23149683 ROTATE INNODB MASTER KEY WITH KEYRING_OKV_CONF_DIR MISSING: SIGSEGV; SIGNAL 11
- BUG#23762382 INSERT VALUES QUERY WITH JOIN IN A SELECT CAUSES INCORRECT BEHAVIOR
- BUG#25209512 CURRENT_TIMESTAMP PRODUCES ZEROS IN TRIGGER
- BUG#26626277 BUG IN "INSERT... ON DUPLICATE KEY UPDATE" QUERY
- BUG#26734162 INCORRECT BEHAVIOR WITH INSERT OF BLOB + ON DUPLICATE KEY UPDATE

- `BUG#27460607 INCORRECT WHEN INSERT SELECT's SOURCE TABLE IS EMPTY`

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Kompatibilitas MySQL 5.7

Aurora MySQL 2.02.4 kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Aurora MySQL 2.02.4 saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Pengidentifikasi transaksi global (GTIDs). Aurora MySQL mendukung GTIDs dalam versi 2.04 dan lebih tinggi.
- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup

- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`
- Protokol X

Pembaruan mesin basis data Aurora MySQL 2018-08-23 (versi 2.02.3) (Dihentikan)

Versi: 2.02.3

Aurora MySQL 2.02.3 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Saat membuat klaster DB Aurora MySQL baru, Anda dapat memilih kompatibilitas dengan MySQL 5.7 atau MySQL 5.6. Saat memulihkan snapshot yang kompatibel dengan MySQL 5.6, Anda dapat memilih kompatibilitas dengan MySQL 5.7 atau MySQL 5.6.

Anda dapat memulihkan snapshot Aurora MySQL 1.14.*, 1.15.*, 1.16.*, 1.17.*, 2.01.*, dan 2.02.* ke Aurora MySQL 2.02.3. Anda juga dapat melakukan peningkatan in-place dari Aurora MySQL 2.01.* atau 2.02.* ke Aurora MySQL 2.02.3.

Kami tidak mengizinkan peningkatan in-place klaster Aurora MySQL 1.* ke Aurora MySQL 2.02.3 atau pemulihan ke Aurora MySQL 2.02.3 dari backup Amazon S3. Kami berencana menghapus batasan ini di rilis Aurora MySQL 2.* mendatang.

Skema kinerja dinonaktifkan untuk rilis Aurora MySQL 5.7 ini. Tingkatkan ke Aurora 2.03 untuk dukungan skema kinerja.

> **Note**
>
> Versi ini saat ini tidak tersedia di wilayah AWS GovCloud (AS-Barat) [us-gov-west-1] dan Tiongkok (Beijing) [cn-north-1]. Akan ada pengumuman terpisah setelah tersedia.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbandingan dengan Aurora MySQL versi 1

Fitur Amazon Aurora MySQL berikut ini didukung di Aurora MySQL versi 1 (kompatibel dengan MySQL 5.6), tetapi fitur-fitur tersebut saat ini tidak didukung di Aurora MySQL versi 2 (kompatibel dengan MySQL 5.7).

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Saat ini, Aurora MySQL 2.01 tidak mendukung fitur yang ditambahkan di Aurora MySQL versi 1.16 dan yang lebih baru. Untuk informasi tentang Aurora MySQL versi 1.16, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).

Kompatibilitas MySQL 5.7

Aurora MySQL 2.02.3 kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Aurora MySQL 2.02.3 saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Pengidentifikasi transaksi global (GTIDs). Aurora MySQL mendukung GTIDs dalam versi 2.04 dan lebih tinggi.
- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`
- Protokol X

Perbedaan CLI antara Aurora MySQL 2.x dan Aurora MySQL 1.x

- Nama mesin untuk Aurora MySQL 2.x adalah `aurora-mysql`; nama mesin untuk Aurora MySQL 1.x tetap `aurora`.
- Grup parameter default untuk Aurora MySQL 2.x adalah `default.aurora-mysql5.7`; grup parameter default untuk Aurora MySQL 1.x tetap `default.aurora5.6`.
- Nama jajaran grup parameter klaster DB untuk Aurora MySQL 2.x adalah `aurora-mysql5.7`; nama jajaran grup parameter klaster DB untuk Aurora MySQL 1.x tetap `aurora5.6`.

Lihat dokumentasi Aurora untuk rangkaian lengkap perintah dan perbedaan CLI antara Aurora MySQL 2.x dan Aurora MySQL 1.x.

Perbaikan

- Memperbaiki masalah di mana Replika Aurora dapat memulai ulang saat menggunakan pemulihan kursor optimis saat membaca catatan.
- Memperbarui nilai default parameter `innodb_stats_persistent_sample_pages` ke 128 untuk meningkatkan statistik indeks.

- Memperbaiki masalah di mana Replika Aurora dapat memulai ulang saat mengakses tabel kecil yang pada saat bersamaan dimodifikasi pada instans primer Aurora.
- Memperbaiki `ANALYZE TABLE` untuk berhenti mem-flush cache definisi tabel.
- Memperbaiki masalah di mana instans primer Aurora atau Replika Aurora dapat memulai ulang saat mengonversi kueri titik untuk geospasial ke rentang pencarian.

Pembaruan mesin basis data Aurora MySQL 2018-06-04 (versi 2.02.2) (Dihentikan)

Versi: 2.02.2

Aurora MySQL 2.02.2 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Saat membuat klaster DB Aurora MySQL baru, Anda dapat memilih kompatibilitas dengan MySQL 5.7 atau MySQL 5.6. Saat memulihkan snapshot yang kompatibel dengan MySQL 5.6, Anda dapat memilih kompatibilitas dengan MySQL 5.7 atau MySQL 5.6.

Anda dapat memulihkan snapshot Aurora MySQL 1.14*, 1.15*, 1.16*, 1.17*, 2.01*, dan 2.02 ke Aurora MySQL 2.02.2. Anda juga dapat melakukan peningkatan in-place dari Aurora MySQL 2.01* atau 2.02 ke Aurora MySQL 2.02.2.

Kami tidak mengizinkan peningkatan in-place klaster Aurora MySQL 1.* ke Aurora MySQL 2.02.2 atau pemulihan ke Aurora MySQL 2.02.2 dari backup Amazon S3. Kami berencana menghapus batasan ini di rilis Aurora MySQL 2.* mendatang.

Skema kinerja dinonaktifkan untuk rilis Aurora MySQL 5.7 ini. Tingkatkan ke Aurora 2.03 untuk dukungan skema kinerja.

> **Note**
>
> Versi ini saat ini tidak tersedia di wilayah AWS GovCloud (AS-Barat) [us-gov-west-1] dan Tiongkok (Beijing) [cn-north-1]. Akan ada pengumuman terpisah setelah tersedia.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan penting:

- CVE-2016-3486

Perbandingan dengan Aurora MySQL versi 5.6

Fitur Amazon Aurora MySQL berikut didukung di Aurora MySQL 5.6, tetapi fitur ini saat ini tidak didukung di Aurora MySQL 5.7.

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Saat ini, Aurora MySQL 2.01 tidak mendukung fitur yang ditambahkan di Aurora MySQL versi 1.16 dan yang lebih baru. Untuk informasi tentang Aurora MySQL versi 1.16, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).

Kompatibilitas MySQL 5.7

Aurora MySQL 2.02.2 kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Aurora MySQL 2.02.2 saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Pengidentifikasi transaksi global (GTIDs). Aurora MySQL mendukung GTIDs dalam versi 2.04 dan lebih tinggi.
- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`
- Protokol X

Perbedaan CLI antara Aurora MySQL 2.x dan Aurora MySQL 1.x

- Nama mesin untuk Aurora MySQL 2.x adalah `aurora-mysql`; nama mesin untuk Aurora MySQL 1.x tetap `aurora`.
- Grup parameter default untuk Aurora MySQL 2.x adalah `default.aurora-mysql5.7`; grup parameter default untuk Aurora MySQL 1.x tetap `default.aurora5.6`.
- Nama jajaran grup parameter klaster DB untuk Aurora MySQL 2.x adalah `aurora-mysql5.7`; nama jajaran grup parameter klaster DB untuk Aurora MySQL 1.x tetap `aurora5.6`.

Lihat dokumentasi Aurora untuk rangkaian lengkap perintah dan perbedaan CLI antara Aurora MySQL 2.x dan Aurora MySQL 1.x.

Perbaikan

- Memperbaiki masalah yang menyebabkan Penulis Aurora terkadang dapat memulai ulang saat melacak kemajuan Replika Aurora.
- Memperbaiki masalah di mana Replika Aurora memulai ulang atau memunculkan kesalahan saat tabel terpartisi diakses setelah menjalankan pernyataan create atau drop indeks pada tabel di Penulis Aurora.

- Memperbaiki masalah di mana tabel pada Replika Aurora tidak dapat diakses saat menerapkan perubahan yang disebabkan oleh menjalankan ADD/DROP pernyataan kolom tabel ALTER pada Penulis Aurora.

Pembaruan mesin basis data Aurora MySQL 2018-05-03 (versi 2.02) (Dihentikan)

Versi: 2.02

Aurora MySQL 2.02 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Saat membuat klaster DB Aurora MySQL baru, Anda dapat memilih kompatibilitas dengan MySQL 5.7 atau MySQL 5.6. Saat memulihkan snapshot yang kompatibel dengan MySQL 5.6, Anda dapat memilih kompatibilitas dengan MySQL 5.7 atau MySQL 5.6.

Anda dapat memulihkan snapshot Aurora MySQL 1.14*, 1.15*, 1.16*, 1.17*, dan 2.01* ke Aurora MySQL 2.02. Anda juga dapat melakukan peningkatan in-place dari Aurora MySQL 2.01.* ke Aurora MySQL 2.02.

Kami tidak mengizinkan peningkatan in-place klaster Aurora MySQL 1.x ke Aurora MySQL 2.02 atau pemulihan ke Aurora MySQL 2.02 dari backup Amazon S3. Kami berencana menghapus batasan ini di rilis Aurora MySQL 2.x mendatang.

Skema kinerja dinonaktifkan untuk rilis Aurora MySQL 5.7 ini. Tingkatkan ke Aurora 2.03 untuk dukungan skema kinerja.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbandingan dengan Aurora MySQL versi 5.6

Fitur Amazon Aurora MySQL berikut didukung di Aurora MySQL 5.6, tetapi fitur ini saat ini tidak didukung di Aurora MySQL 5.7.

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.

- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Saat ini, Aurora MySQL 2.01 tidak mendukung fitur yang ditambahkan di Aurora MySQL versi 1.16 dan yang lebih baru. Untuk informasi tentang Aurora MySQL versi 1.16, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).

Kompatibilitas MySQL 5.7

Aurora MySQL 2.02 kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Aurora MySQL 2.02 saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Pengidentifikasi transaksi global (GTIDs). Aurora MySQL mendukung GTIDs dalam versi 2.04 dan lebih tinggi.
- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri

- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`
- Protokol X

Perbedaan CLI antara Aurora MySQL 2.x dan Aurora MySQL 1.x

- Nama mesin untuk Aurora MySQL 2.x adalah `aurora-mysql`; nama mesin untuk Aurora MySQL 1.x tetap `aurora`.
- Grup parameter default untuk Aurora MySQL 2.x adalah `default.aurora-mysql5.7`; grup parameter default untuk Aurora MySQL 1.x tetap `default.aurora5.6`.
- Nama jajaran grup parameter klaster DB untuk Aurora MySQL 2.x adalah `aurora-mysql5.7`; nama jajaran grup parameter klaster DB untuk Aurora MySQL 1.x tetap `aurora5.6`.

Lihat dokumentasi Aurora untuk rangkaian lengkap perintah dan perbedaan CLI antara Aurora MySQL 2.x dan Aurora MySQL 1.x.

Perbaikan

- Memperbaiki masalah di mana Penulis Aurora memulai ulang saat menjalankan pernyataan INSERT dan memanfaatkan optimalisasi Fast Insert.
- Memperbaiki masalah di mana Replika Aurora memulai ulang saat menjalankan pernyataan ALTER DATABASE pada Replika Aurora.
- Memperbaiki masalah di mana Replika Aurora memulai ulang saat menjalankan kueri pada tabel yang baru saja dihapus di Penulis Aurora.
- Memperbaiki masalah di mana Replika Aurora memulai ulang saat mengatur `innodb_adaptive_hash_index` ke `OFF` di Replika Aurora.
- Memperbaiki masalah yang menyebabkan Replika Aurora memulai ulang saat menjalankan pertanyaan TRUNCATE TABLE di Penulis Aurora.
- Memperbaiki masalah di mana Penulis Aurora macet dalam keadaan tertentu saat menjalankan pernyataan INSERT. Pada klaster multisimpul, hal ini dapat mengakibatkan failover.
- Memperbaiki kebocoran memori yang terkait dengan pengaturan variabel sesi.
- Memperbaiki masalah di mana Penulis Aurora macet dalam keadaan tertentu terkait dengan pengurungan pembersihan untuk kolom yang dihasilkan.

- Memperbaiki masalah di mana Penulis Aurora terkadang dapat memulai ulang ketika pencatatan log biner diaktifkan.

Integrasi perbaikan bug MySQL

- Sambungan kiri menampilkan hasil yang salah pada sisi luar (Bug #22833364).

Pembaruan mesin basis data Aurora MySQL 2018-03-13 (versi 2.01.1) (Dihentikan)

Versi: 2.01.1

Aurora MySQL 2.01.1 tersedia secara umum. Aurora MySQL versi 2.x kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x kompatibel dengan MySQL 5.6.

Saat membuat klaster DB Aurora MySQL baru, Anda dapat memilih kompatibilitas dengan MySQL 5.7 atau MySQL 5.6. Saat memulihkan snapshot yang kompatibel dengan MySQL 5.6, Anda dapat memilih kompatibilitas dengan MySQL 5.7 atau MySQL 5.6.

Anda dapat memulihkan snapshot Aurora MySQL 1.14*, 1.15*, 1.16*, dan 1.17* ke Aurora MySQL 2.01.1.

Kami tidak mengizinkan peningkatan in-place klaster Aurora MySQL 1.x ke Aurora MySQL 2.01.1 atau pemulihan ke Aurora MySQL 2.01.1 dari backup Amazon S3. Kami berencana menghapus batasan ini di rilis Aurora MySQL 2.x mendatang.

Skema kinerja dinonaktifkan untuk rilis Aurora MySQL 5.7 ini. Tingkatkan ke Aurora 2.03 untuk dukungan skema kinerja.

Perbandingan dengan Aurora MySQL versi 5.6

Fitur Amazon Aurora MySQL berikut didukung di Aurora MySQL 5.6, tetapi fitur ini saat ini tidak didukung di Aurora MySQL 5.7.

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.

- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Saat ini, Aurora MySQL 2.01.1 tidak mendukung fitur yang ditambahkan di Aurora MySQL versi 1.16 dan yang lebih baru. Untuk informasi tentang Aurora MySQL versi 1.16, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).

Kompatibilitas MySQL 5.7

Aurora MySQL 2.01.1 kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Aurora MySQL 2.01.1 saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Pengidentifikasi transaksi global (GTIDs). Aurora MySQL mendukung GTIDs dalam versi 2.04 dan lebih tinggi.
- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri

- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`
- Protokol X

Perbedaan CLI antara Aurora MySQL 2.x dan Aurora MySQL 1.x

- Nama mesin untuk Aurora MySQL 2.x adalah `aurora-mysql`; nama mesin untuk Aurora MySQL 1.x tetap `aurora`.
- Grup parameter default untuk Aurora MySQL 2.x adalah `default.aurora-mysql5.7`; grup parameter default untuk Aurora MySQL 1.x tetap `default.aurora5.6`.
- Nama jajaran grup parameter klaster DB untuk Aurora MySQL 2.x adalah `aurora-mysql5.7`; nama jajaran grup parameter klaster DB untuk Aurora MySQL 1.x tetap `aurora5.6`.

Lihat dokumentasi Aurora untuk rangkaian lengkap perintah dan perbedaan CLI antara Aurora MySQL 2.x dan Aurora MySQL 1.x.

Perbaikan

- Memperbaiki masalah pada pemulihan snapshot di mana hak istimewa basis data khusus Aurora dibuat secara tidak benar ketika snapshot yang kompatibel dengan MySQL 5.6 dipulihkan dengan kompatibilitas MySQL 5.7.
- Menambahkan dukungan untuk pemulihan snapshot 1.17.

Pembaruan mesin basis data Aurora MySQL 2018-02-06 (versi 2.01) (Dihentikan)

Versi: 2.01

Aurora MySQL 2.01 tersedia secara umum. Selanjutnya, Aurora MySQL versi 2.x akan kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.x akan kompatibel dengan MySQL 5.6.

Saat membuat klaster DB Aurora MySQL baru, termasuk yang dipulihkan dari snapshot, Anda dapat memilih kompatibilitas dengan MySQL 5.7 atau MySQL 5.6.

Anda dapat memulihkan snapshot Aurora MySQL 1.14*, 1.15*, dan 1.16* ke Aurora MySQL 2.01.

Kami tidak mengizinkan peningkatan in-place klaster Aurora MySQL 1.x ke Aurora MySQL 2.01 atau pemulihan ke Aurora MySQL 2.01 dari backup Amazon S3. Kami berencana menghapus batasan ini di rilis Aurora MySQL 2.x mendatang.

Skema kinerja dinonaktifkan untuk rilis Aurora MySQL 5.7 ini. Tingkatkan ke Aurora 2.03 untuk dukungan skema kinerja.

Perbandingan dengan Aurora MySQL versi 5.6

Fitur Amazon Aurora MySQL berikut didukung di Aurora MySQL 5.6, tetapi fitur ini saat ini tidak didukung di Aurora MySQL 5.7.

- Asynchronous key prefetch (AKP). Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Sambungan hash. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Fungsi asli untuk memanggil AWS Lambda fungsi secara sinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dengan fungsi asli Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Pindai batching. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).
- Memigrasikan data dari MySQL menggunakan bucket Amazon S3. Untuk informasi selengkapnya, lihat Memigrasikan data dari MySQL menggunakan bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Saat ini, Aurora MySQL 2.01 tidak mendukung fitur yang ditambahkan di Aurora MySQL versi 1.16 dan yang lebih baru. Untuk informasi tentang Aurora MySQL versi 1.16, lihat Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan).

Kompatibilitas MySQL 5.7

Aurora MySQL 2.01 kompatibel dengan kabel dengan MySQL 5.7 dan menyertakan fitur seperti dukungan JSON, indeks spasial, dan kolom yang dihasilkan. Aurora MySQL menggunakan implementasi asli pengindeksan spasial menggunakan kurva z-order untuk memberikan kinerja tulis >20x lebih baik dan kinerja baca >10x lebih baik daripada MySQL 5.7 untuk set data spasial.

Aurora MySQL 2.01 saat ini tidak mendukung fitur MySQL 5.7 berikut:

- Pengidentifikasi transaksi global (GTIDs). Aurora MySQL mendukung GTIDs dalam versi 2.04 dan lebih tinggi.
- Plugin replikasi kelompok
- Peningkatan ukuran halaman
- Pemuatan pool buffer InnoDB saat startup
- Plugin pengurai teks lengkap InnoDB
- Replikasi multisumber
- Perubahan ukuran pool buffer online
- Plugin validasi kata sandi
- Plugin tulis ulang kueri
- Penyaringan replikasi
- Pernyataan SQL `CREATE TABLESPACE`
- Protokol X

Perbedaan CLI antara Aurora MySQL 2.x dan Aurora MySQL 1.x

- Nama mesin untuk Aurora MySQL 2.x adalah `aurora-mysql`; nama mesin untuk Aurora MySQL 1.x tetap `aurora`.
- Grup parameter default untuk Aurora MySQL 2.x adalah `default.aurora-mysql5.7`; grup parameter default untuk Aurora MySQL 1.x tetap `default.aurora5.6`.
- Nama jajaran grup parameter klaster DB untuk Aurora MySQL 2.x adalah `aurora-mysql5.7`; nama jajaran grup parameter klaster DB untuk Aurora MySQL 1.x tetap `aurora5.6`.

Lihat dokumentasi Aurora untuk rangkaian lengkap perintah dan perbedaan CLI antara Aurora MySQL 2.x dan Aurora MySQL 1.x.

Pembaruan mesin basis data untuk Amazon Aurora MySQL versi 1 (Dihentikan)

Berikut ini adalah pembaruan mesin basis data Amazon Aurora versi 1:

- Pembaruan mesin basis data Aurora MySQL 2021-09-30 (versi 1.23.4) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2021-06-28 (versi 1.23.3) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2021-03-18 (versi 1.23.2) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2020-11-24 (versi 1.23.1) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2020-09-02 (versi 1.23.0) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2021-06-03 (versi 1.22.5) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2021-03-04 (versi 1.22.4) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2020-11-09 (versi 1.22.3) (Usang)
- Pembaruan mesin basis data Aurora MySQL 2020-03-05 (versi 1.22.2) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2019-12-23 (versi 1.22.1) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2019-11-25 (versi 1.22.0) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2019-11-25 (versi 1.21.0) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2020-03-05 (versi 1.20.1) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2019-11-11 (versi 1.20.0) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2020-03-05 (versi 1.19.6) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2019-09-19 (versi 1.19.5) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2019-06-05 (versi 1.19.2) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2019-05-09 (versi 1.19.1) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2019-02-07 (versi 1.19.0) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2018-09-20 (versi 1.18.0) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2020-03-05 (versi 1.17.9) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2019-01-17 (versi 1.17.8) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2018-10-08 (versi 1.17.7) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2018-09-06 (versi 1.17.6) (Dihentikan)

- Pembaruan mesin basis data Aurora MySQL 2018-08-14 (versi 1.17.5) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2018-08-07 (versi 1.17.4) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2018-06-05 (versi 1.17.3) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2018-04-27 (versi 1.17.2) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2018-03-23 (versi 1.17.1) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2018-03-13 (versi 1.17) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2017-11-20 (versi 1.15.1) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2017-10-24 (versi 1.15) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2018-03-13 (versi 1.14.4) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL: 2017-09-22 (versi 1.14.1) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2017-08-07 (versi 1.14) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL: 2017-05-15 (versi 1.13) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL: 2017-04-05 (versi 1.12) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL: 2017-02-23 (versi 1.11) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL: 2017-01-12 (versi 1.10.1) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL: 2016-12-14 (versi 1.10) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL 2016-11-10 (versi 1.9.0, 1.9.1) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL: 2016-10-26 (versi 1.8.1) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL: 2016-10-18 (versi 1.8) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL: 2016-09-20 (versi 1.7.1) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL: 2016-08-30 (versi 1.7.0) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL: 2016-06-01 (versi 1.6.5) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL: 2016-04-06 (versi 1.6) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL: 2016-01-11 (versi 1.5) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL: 2015-12-03 (versi 1.4) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL: 2015-10-16 (versi 1.2, 1.3) (Dihentikan)
- Pembaruan mesin basis data Aurora MySQL: 2015-08-24 (versi 1.1) (Dihentikan)

Pembaruan mesin basis data Aurora MySQL 2021-09-30 (versi 1.23.4) (Dihentikan)

Versi: 1.23.4

Aurora MySQL 1.23.4 tersedia secara umum. Aurora MySQL versi 2.* kompatibel dengan MySQL 5.7 dan Aurora MySQL versi 1.* kompatibel dengan MySQL 5.6.

Versi mesin ini dijadwalkan akan dihentikan pada 28 Februari 2023. Untuk informasi selengkapnya, lihat Mempersiapkan akhir masa pakai Amazon Aurora Edisi Kompatibel MySQL versi 1.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.* dan 3.02.*.

Untuk membuat cluster dengan versi lama Aurora MySQL, tentukan versi engine melalui RDS Console, AWS CLI, atau Amazon RDS API.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan umum:

- Memperbaiki masalah yang dapat menyebabkan konsumsi CPU tinggi pada instans pembaca karena pencatatan log pesan informasi yang berlebihan dalam file log diagnostik internal.

Perbaikan prioritas tinggi:

- CVE-2021-2307
- CVE-2021-2226
- CVE-2021-2160
- CVE-2021-2154
- CVE-2021-2060
- CVE-2021-2032

- CVE-2021-2001

Pembaruan mesin basis data Aurora MySQL 2021-06-28 (versi 1.23.3) (Dihentikan)

Versi: 1.23.3

Aurora MySQL 1.23.3 tersedia secara umum. Aurora MySQL versi 1.* kompatibel dengan MySQL 5.6 dan Aurora MySQL versi 2.* kompatibel dengan MySQL 5.7.

Versi mesin ini dijadwalkan akan dihentikan pada 28 Februari 2023. Untuk informasi selengkapnya, lihat Mempersiapkan akhir masa pakai Amazon Aurora Edisi Kompatibel MySQL versi 1.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.* dan 3.02.*.

Untuk membuat cluster dengan versi lama Aurora MySQL, tentukan versi engine melalui RDS Console, AWS CLI, atau Amazon RDS API.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Peningkatan stabilitas dan ketersediaan umum.

Perbaikan keamanan:

- CVE-2021-23841
- CVE-2021-3449
- CVE-2020-28196

Pembaruan mesin basis data Aurora MySQL 2021-03-18 (versi 1.23.2) (Dihentikan)

Versi: 1.23.2

Aurora MySQL 1.23.2 tersedia secara umum. Aurora MySQL versi 1.* kompatibel dengan MySQL 5.6 dan Aurora MySQL versi 2.* kompatibel dengan MySQL 5.7.

Versi mesin ini dijadwalkan akan dihentikan pada 28 Februari 2023. Untuk informasi selengkapnya, lihat Mempersiapkan akhir masa pakai Amazon Aurora Edisi Kompatibel MySQL versi 1.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.* dan 3.02.*.

Untuk membuat cluster dengan versi lama Aurora MySQL, tentukan versi engine melalui RDS Console, AWS CLI, atau Amazon RDS API.

> **Note**
>
> Versi ini saat ini tidak tersedia di wilayah berikut: AWS GovCloud (AS-Timur) [us-gov-east-1], AWS GovCloud (AS-Barat) [us-gov-west-1]. Akan ada pengumuman terpisah setelah tersedia.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan prioritas tinggi:

- CVE-2020-14867
- CVE-2020-14812
- CVE-2020-14769
- CVE-2020-14765
- CVE-2020-14793
- CVE-2020-14672
- CVE-2020-1971
- CVE-2018-3143

Perbaikan ketersediaan:

- Memperbaiki masalah pada fitur penyesuaian ukuran penyimpanan klaster dinamis yang dapat menyebabkan instans DB pembaca memulai ulang.
- Memperbaiki masalah failover karena kondisi balapan dalam pernyataan `RESET QUERY CACHE`.
- Memperbaiki crash dalam panggilan prosedur tersimpan bersarang dengan cache kueri.
- Memperbaiki masalah untuk mencegah mulai ulang `mysqld` beberapa kali saat memulihkan dari pemotongan tabel yang dipartisi atau disubpartisi yang tidak lengkap.
- Memperbaiki masalah yang dapat menyebabkan kegagalan migrasi dari on-premise atau RDS for MySQL ke Aurora MySQL.
- Memperbaiki kondisi balapan yang sangat jarang terjadi di mana basis data dapat memulai ulang selama penskalaan volume penyimpanan.
- Memperbaiki masalah pada manajer kunci di mana kondisi balapan dapat menyebabkan sebuah kunci dimiliki bersama oleh dua transaksi, yang menyebabkan basis data memulai ulang.
- Memperbaiki masalah yang berkaitan dengan manajemen memori kunci transaksi dengan transaksi tulis jangka panjang yang menyebabkan basis data memulai ulang.
- Memperbaiki kondisi balapan di manajer kunci yang mengakibatkan basis data memulai ulang atau melakukan failover selama rollback transaksi.
- Memperbaiki masalah selama peningkatan dari 5.6 ke 5.7 ketika tabel mengaktifkan DDL Online Cepat dalam mode lab di 5.6.
- Memperbaiki beberapa masalah di mana mesin mungkin memulai ulang selama patching zero-downtime sambil memeriksa titik quiesced dalam aktivitas basis data untuk patching.
- Memperbaiki beberapa masalah yang berkaitan dengan mulai ulang beberapa kali karena operasi DDL yang terganggu, seperti `DROP TRIGGER`, `ALTER TABLE`, dan khususnya `ALTER TABLE` yang memodifikasi jenis partisi atau jumlah partisi di dalam tabel.
- Memperbarui nilai default `table_open_cache` pada instans 16XL dan 24XL untuk menghindari mulai ulang yang terjadi berulang-ulang dan penggunaan CPU yang tinggi pada kelas instans besar (R4/R5-16XL, R5-12XL, R5-24XL). Hal ini berdampak pada rilis 1.21.x dan 1.22.x.
- Memperbaiki masalah yang menyebabkan replika binlog berhenti dengan kesalahan `HA_ERR_KEY_NOT_FOUND`.

Integrasi perbaikan bug MySQL Community Edition

- Replikasi: Selama pernyataan `SHOW BINLOG EVENTS` dijalankan, setiap transaksi paralel diblokir. Perbaikan ini memastikan proses `SHOW BINLOG EVENTS` sekarang hanya memperoleh sebuah

kunci selama menghitung posisi akhir file, oleh karena itu transaksi paralel tidak diblokir dalam jangka waktu lama. (Bug #76618, Bug #20928790)

Pembaruan mesin basis data Aurora MySQL 2020-11-24 (versi 1.23.1) (Dihentikan)

Versi: 1.23.1

Aurora MySQL 1.23.1 tersedia secara umum. Aurora MySQL versi 1.* kompatibel dengan MySQL 5.6 dan Aurora MySQL versi 2.* kompatibel dengan MySQL 5.7.

Versi mesin ini dijadwalkan akan dihentikan pada 28 Februari 2023. Untuk informasi selengkapnya, lihat Mempersiapkan akhir masa pakai Amazon Aurora Edisi Kompatibel MySQL versi 1.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.* dan 3.02.*.

Untuk membuat cluster dengan versi lama Aurora MySQL, tentukan versi engine melalui RDS Console, AWS CLI, atau Amazon RDS API.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan keamanan:

Perbaikan dan penyempurnaan lain untuk penanganan fine-tune di lingkungan terkelola. Di bawah ini adalah beberapa perbaikan CVE tambahan:

- CVE-2020-14559
- CVE-2020-14539

Perubahan tidak kompatibel:

Versi ini memperkenalkan perubahan izin yang memengaruhi perilaku perintah `mysqldump`. Pengguna harus memiliki hak istimewa `PROCESS` untuk mengakses tabel

`INFORMATION_SCHEMA.FILES`. Untuk menjalankan perintah `mysqldump` tanpa perubahan apa pun, berikan hak istimewa `PROCESS` untuk pengguna basis data yang terhubung dengan perintah `mysqldump`. Anda juga dapat menjalankan perintah `mysqldump` dengan opsi `--no-tablespaces`. Dengan opsi tersebut, output `mysqldump` tidak menyertakan pernyataan `CREATE LOGFILE GROUP` atau `CREATE TABLESPACE` apa pun. Dalam hal ini, perintah `mysqldump` tidak mengakses tabel `INFORMATION_SCHEMA.FILES`, dan Anda tidak perlu memberikan izin `PROCESS`.

Perbaikan ketersediaan:

- Memperbaiki masalah yang menyebabkan instans pembaca Aurora di dalam klaster sekunder basis data global yang menjalankan 1.23.0 memulai ulang berulang kali.
- Memperbaiki masalah yang menyebabkan replika Wilayah sekunder basis data global mungkin memulai ulang saat ditingkatkan ke rilis 1.23.0 sementara penulis Wilayah primer adalah versi rilis yang lebih lama.
- Memperbaiki kebocoran memori pada fitur penyesuaian ukuran dinamis, yang diperkenalkan di Aurora MySQL 1.23.0.
- Memperbaiki masalah yang mungkin menyebabkan server memulai ulang selama pelaksanaan kueri menggunakan fitur kueri paralel.
- Memperbaiki masalah yang mungkin menyebabkan sesi klien macet saat mesin basis data mengalami kesalahan saat membaca dari atau menulis ke jaringan.

Pembaruan mesin basis data Aurora MySQL 2020-09-02 (versi 1.23.0) (Dihentikan)

Versi: 1.23.0

Aurora MySQL 1.23.0 tersedia secara umum. Aurora MySQL versi 1.* kompatibel dengan MySQL 5.6 dan Aurora MySQL versi 2.* kompatibel dengan MySQL 5.7.

Versi mesin ini dijadwalkan akan dihentikan pada 28 Februari 2023. Untuk informasi selengkapnya, lihat Mempersiapkan akhir masa pakai Amazon Aurora Edisi Kompatibel MySQL versi 1.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Anda dapat memulihkan snapshot basis data Aurora MySQL 1.* ke Aurora MySQL 1.23.0.

> **Important**
>
> Perbaikan pada penyimpanan Aurora dalam versi ini membatasi jalur peningkatan yang tersedia dari Aurora MySQL 1.23 ke Aurora MySQL 2.*. Saat Anda meningkatkan klaster Aurora MySQL 1.23 ke 2.*, Anda harus meningkatkan versi ke Aurora MySQL 2.09.0 atau yang lebih baru.

Untuk membuat cluster dengan versi lama Aurora MySQL, harap tentukan versi mesin melalui Konsol RDS, AWS CLI, atau Amazon RDS API.

> **Note**
>
> Versi ini saat ini tidak tersedia di wilayah berikut: AWS GovCloud (AS-Timur) [us-gov-east-1], AWS GovCloud (AS-Barat) [us-gov-west-1]. Akan ada pengumuman terpisah setelah tersedia.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Fitur baru:

- Anda sekarang dapat mengaktifkan atau menonaktifkan kueri paralel untuk klaster yang ada dengan mengubah nilai parameter klaster DB `aurora_parallel_query`. Anda tidak perlu menggunakan pengaturan `parallelquery` untuk parameter `--engine-mode` saat membuat klaster.

  Kueri paralel sekarang diperluas agar tersedia di semua wilayah di mana Aurora MySQL tersedia.

  Ada sejumlah peningkatan dan perubahan fungsi lain pada prosedur untuk peningkatan dan pengaktifan kueri paralel di klaster Aurora. Untuk informasi selengkapnya, lihat Menggunakan kueri paralel untuk Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Dengan rilis ini, Anda dapat membuat instans basis data Amazon Aurora MySQL dengan penyimpanan hingga 128 tebibyte (TiB). Batas penyimpanan baru meningkat dari 64 TiB. Ukuran

penyimpanan 128 TiB mendukung basis data yang lebih besar. Kapasitas ini tidak didukung pada ukuran instans kecil (db.t2 atau db.t3). Satu ruang tabel tidak dapat meluas melebihi 64 TiB karena adanya batasan InnoDB dengan ukuran halaman 16 KB.

Aurora memperingatkan Anda saat ukuran volume klaster mendekati 128 TiB, sehingga Anda dapat mengambil tindakan sebelum mencapai batas ukuran. Peringatan muncul di log mysql dan Peristiwa RDS di Konsol Manajemen AWS.

- Meningkatkan pemrosesan log biner (binlog) untuk mengurangi waktu pemulihan crash dan latensi waktu commit ketika terdapat transaksi yang sangat besar.
- Aurora secara dinamis menyesuaikan ukuran ruang penyimpanan klaster Anda. Dengan penyesuaian ukuran dinamis, ruang penyimpanan untuk klaster DB Aurora menurun secara otomatis saat Anda menghapus data dari klaster DB. Lihat informasi selengkapnya di Penskalaan penyimpanan di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Fitur pengubahan ukuran dinamis sedang digunakan secara bertahap ke AWS Wilayah tempat Aurora tersedia. Tergantung Wilayah tempat klaster Anda berada, fitur ini mungkin belum tersedia. Untuk informasi selengkapnya, lihat pengumuman Apa Yang Baru.

Perbaikan prioritas tinggi:

- CVE-2019-3822
- CVE-2019-2537
- CVE-2018-2787
- CVE-2018-2784
- CVE-2018-2645
- CVE-2018-2640

Perbaikan ketersediaan:

- Memperbaiki masalah pada manajer kunci di mana kondisi balapan dapat menyebabkan sebuah kunci dimiliki bersama oleh dua transaksi, yang menyebabkan basis data memulai ulang.
- Memperbaiki masalah yang berkaitan dengan manajemen memori kunci transaksi dengan transaksi tulis jangka panjang yang menyebabkan basis data memulai ulang.

- Memperbaiki kondisi balapan di manajer kunci yang mengakibatkan basis data memulai ulang atau melakukan failover selama rollback transaksi.
- Memperbaiki masalah selama peningkatan dari 5.6 ke 5.7 saat `innodb_file_format` berubah di tabel dengan DDL Cepat diaktifkan.
- Memperbaiki beberapa masalah di mana mesin mungkin memulai ulang selama patching zero-downtime sambil memeriksa titik quiesced dalam aktivitas basis data untuk patching.
- Memperbaiki masalah yang terkait dengan pemulihan DDL yang memengaruhi mulai ulang instans DB saat memulihkan operasi `DROP TRIGGER` yang terganggu.
- Memperbaiki bug yang dapat menyebabkan tidak tersedianya basis data jika terjadi crash selama pelaksanaan operasi partisi tertentu. Secara khusus, gangguan operasi `ALTER TABLE` yang memodifikasi jenis partisi atau jumlah partisi di dalam tabel.
- Perbaiki nilai default dari `table_open_cache` pada 16XL dan 24XL yang dapat menyebabkan failover berulang dan pemanfaatan CPU yang tinggi pada kelas Large besar (R4/R5-16XL, R5-12XL, R5-24XL). Hal ini berdampak pada 1.21.x dan 1.22.x.

Basis data global:

- Isi data yang hilang dalam tampilan `INFORMATION_SCHEMA.REPLICA_HOST_STATUS` MySQL di AWS Wilayah primer dan sekunder dalam database global Aurora.
- Memperbaiki kegagalan kueri tak terduga yang dapat terjadi di Wilayah sekunder DB Global akibat pengumpulan sampah catatan UNDO di Wilayah primer, setelah masalah konektivitas jaringan sementara antara Wilayah primer dan sekunder.

Kueri paralel:

- Memperbaiki masalah di mana kueri paralel dapat menyebabkan kueri yang berlangsung lama memberikan hasil kosong.
- Memperbaiki masalah di mana kueri pada tabel kecil di replika baca Aurora memakan waktu lebih dari satu detik.
- Memperbaiki masalah yang mungkin menyebabkan mulai ulang saat kueri paralel dan pernyataan DML berjalan secara bersamaan di bawah beban kerja berat.

Perbaikan umum:

- Memperbaiki masalah di mana kueri yang menggunakan indeks spasial mungkin mengembalikan hasil parsial jika indeks spasial dibuat pada tabel yang sudah memiliki nilai spasial besar.
- Peningkatan panjang maksimum yang diizinkan untuk variabel sistem Audit `server_audit_incl_users` dan `server_audit_excl_users` dari 1024 byte hingga 2000 byte.
- Memperbaiki masalah di mana replika binlog yang terhubung ke primer binlog Aurora MySQL mungkin menampilkan data yang tidak lengkap saat primer binlog Aurora MySQL memuat data dari S3 di bawah `statement` `binlog_format`.
- Mematuhi perilaku komunitas untuk memetakan binlog_format `mixed` ke `row`, bukan ke `statement` untuk memuat data.
- Memperbaiki masalah yang menyebabkan replikasi binlog berhenti berfungsi ketika pengguna menutup koneksi dan sesi menggunakan tabel sementara.
- Memperbaiki waktu respons kueri yang melibatkan tabel sementara MyISAM.
- Memperbaiki penerbitan izin saat pekerja binlog menjalankan fungsi lambda asli.
- Memperbaiki masalah pada replika baca Aurora saat mencoba melakukan kueri atau merotasi log lambat atau log umum.
- Memperbaiki masalah yang merusak replikasi logis ketika `binlog_checksum` parameter diatur ke nilai yang berbeda pada master dan replika.
- Memperbaiki masalah replika baca yang mungkin melihat sekilas hasil parsial transaksi yang baru saja diterapkan pada penulis.
- Menyertakan informasi transaksi yang dibatalkan di `show engine innodb status` saat deadlock diatasi.

Integrasi perbaikan bug MySQL Community Edition

- Peristiwa binlog dengan `ALTER TABLE ADD COLUMN ALGORITHM=QUICK` akan ditulis ulang sebagai `ALGORITHM=DEFAULT` agar kompatibel dengan edisi komunitas.
- BUG #22350047: JIKA KLIEN DIMATIKAN SETELAH ROLLBACK KE SAVEPOINT STMTS SEBELUMNYA DITERAPKAN
- Bug #29915479: MENJALANKAN COM_REGISTER_SLAVE TANPA COM_BINLOG_DUMP DAPAT BERDAMPAK PADA KELUAR SERVER
- Bug #30441969: BUG #29723340: MYSQL SERVER CRASH SETELAH KUERI SQL DENGAN ? AST DATA

- Bug #30628268: CRASH KEHABISAN MEMORI
- Bug #27081349: PERILAKU YANG TAK DIHARAPKAN SAAT MENGHAPUS DENGAN FUNGSI SPASIAL
- Bug #27230859: PERILAKU YANG TAK DIHARAPKAN SAAT PENANGANAN POLIGON TIDAK VALID"
- Bug #27081349: PERILAKU YANG TAK DIHARAPKAN SAAT MENGHAPUS DENGAN SPASIAL"
- Bug #26935001: TABEL ALTERNATIF AUTO_INCREMENT MENCOBA MEMBACA INDEKS DARI RUANG TABEL YANG DIHAPUS
- Bug #29770705: SERVER CRASH SAAT MELAKUKAN SELECT DENGAN KLAUSA WHERE KHUSUS
- Bug #27659490: SELECT MENGGUNAKAN DYNAMIC RANGE DAN INDEX MERGE MENGGUNAKAN TERLALU BANYAK MEMORI(OOM)
- Bug #24786290: REPLIKASI RUSAK SETELAH BUG #74145 TERJADI DI MASTER
- Bug #27703912: PENGGUNAAN MEMORI BERLEBIH DENGAN BANYAK PERSIAPAN
- Bug #20527363: POTONG KERUSAKAN TABEL SEMENTARA:! DICT_ TF2 _FLAG_IS_SET (TABEL, DICT_ _SEMENTARA) TF2
- Bug#23103937 PS_TRUNCATE_ALL_TABLES() TIDAK DAPAT DIGUNAKAN DALAM MODE SUPER_READ_ONLY
- Bug #25053286: MENGGUNAKAN TAMPILAN DENGAN KONDISI DALAM PROSEDUR MENYEBABKAN PERILAKU YANG TIDAK BENAR (diperbaiki di 5.6.36)
- Bug #25586773: PERILAKU YANG TIDAK BENAR UNTUK CREATE TABLE SELECT IN A LOOP IN SP (diperbaiki di 5.6.39)
- Bug #27407480: AUTOMATIC_SP_PRIVILEGES MEMBUTUHKAN HAK ISTIMEWA UNTUK TABEL MYSQL.USER
- Bug #26997096: Nilai `relay_log_space` tidak diperbarui secara sinkron sehingga nilainya terkadang jauh lebih tinggi daripada ruang disk aktual yang digunakan oleh log relay.
- Bug#15831300 SLAVE_TYPE_CONVERSIONS=ALL_NON_LOSSY TIDAK BEKERJA SEBAGAIMANA DIHARAPKAN
- SSL Bug backport Bug #17087862, Bug #20551271
- Bug #16894092: REGRESI KINERJA DALAM 5.6.6+ UNTUK DIMASUKKAN KE ... SELECT … FROM (diperbaiki di 5.6.15).
- Masukkan perbaikan bug yang terkait dengan `SLAVE_TYPE_CONVERSIONS`.

Pembaruan mesin basis data Aurora MySQL 2021-06-03 (versi 1.22.5) (Dihentikan)

Versi: 1.22.5

Aurora MySQL 1.22.5 tersedia secara umum. Aurora MySQL versi 1.* kompatibel dengan MySQL 5.6 dan Aurora MySQL versi 2.* kompatibel dengan MySQL 5.7.

Versi mesin ini dijadwalkan akan dihentikan pada 28 Februari 2023. Untuk informasi selengkapnya, lihat Mempersiapkan akhir masa pakai Amazon Aurora Edisi Kompatibel MySQL versi 1.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Untuk membuat klaster dengan Aurora MySQL versi lama, tentukan versi mesin melalui RDS Console, AWS CLI, atau API Amazon RDS.

> **Note**
>
> Versi ini ditetapkan sebagai rilis dukungan jangka panjang (LTS). Untuk informasi selengkapnya, lihat Rilis dukungan jangka panjang (LTS) Aurora MySQL di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan ketersediaan:

- Menyelesaikan masalah yang dapat menyebabkan basis data terhenti, kemudian memulai ulang atau melakukan failover karena konflik serentak antara beberapa thread pembersihan internal.
- Menyelesaikan masalah yang dapat menyebabkan klaster menjadi tidak tersedia jika basis data memulai ulang saat menahan transaksi XA dalam status siap, lalu memulai ulang lagi sebelum transaksi tersebut dijalankan atau dibatalkan. Sebelum perbaikan ini, Anda dapat mengatasi masalah tersebut dengan memulihkan klaster ke titik waktu tertentu sebelum mulai ulang pertama.

- Menyelesaikan masalah yang dapat menyebabkan pembersihan InnoDB diblokir jika basis data memulai ulang saat memproses pernyataan DDL. Akibatnya, daftar riwayat InnoDB akan bertambah panjang dan volume penyimpanan klaster akan terus bertambah hingga penuh, sehingga basis data menjadi tidak tersedia. Sebelum perbaikan ini, Anda dapat mengatasi masalah tersebut dengan memulai ulang basis data lagi untuk membuka blokir pembersihan.

Pembaruan mesin basis data Aurora MySQL 2021-03-04 (versi 1.22.4) (Dihentikan)

Versi: 1.22.4

Aurora MySQL 1.22.4 tersedia secara umum. Aurora MySQL versi 1.* kompatibel dengan MySQL 5.6 dan Aurora MySQL versi 2.* kompatibel dengan MySQL 5.7.

Versi mesin ini dijadwalkan akan dihentikan pada 28 Februari 2023. Untuk informasi selengkapnya, lihat Mempersiapkan akhir masa pakai Amazon Aurora Edisi Kompatibel MySQL versi 1.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.* dan 3.02.*.

Untuk membuat cluster dengan versi lama Aurora MySQL, tentukan versi engine melalui RDS Console, AWS CLI, atau Amazon RDS API.

> **Note**
>
> Versi ini ditetapkan sebagai rilis dukungan jangka panjang (LTS). Untuk informasi selengkapnya, lihat Rilis dukungan jangka panjang (LTS) Aurora MySQL di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan keamanan:

Perbaikan dan penyempurnaan lain untuk penanganan fine-tune di lingkungan terkelola. Di bawah ini adalah beberapa perbaikan CVE tambahan:

- CVE-2020-14867
- CVE-2020-14812
- CVE-2020-14793
- CVE-2020-14769
- CVE-2020-14765
- CVE-2020-14672
- CVE-2020-1971

Perbaikan ketersediaan:

- Memperbaiki masalah yang dapat memicu basis data memulai ulang atau melakukan failover selama perintah `kill session`. Jika Anda mengalami masalah ini, hubungi AWS dukungan untuk mengaktifkan perbaikan ini pada instans Anda.
- Memperbaiki pencatatan log biner untuk mengurangi waktu pemulihan crash dan latensi commit saat melibatkan transaksi besar.
- Memperbaiki masalah yang menyebabkan replika binlog berhenti dengan kesalahan `HA_ERR_KEY_NOT_FOUND`.

Pembaruan mesin basis data Aurora MySQL 2020-11-09 (versi 1.22.3) (Usang)

Versi: 1.22.3

Aurora MySQL 1.22.3 tersedia secara umum. Aurora MySQL versi 1.* kompatibel dengan MySQL 5.6 dan Aurora MySQL versi 2.* kompatibel dengan MySQL 5.7.

Versi mesin ini dijadwalkan akan dihentikan pada 28 Februari 2023. Untuk informasi selengkapnya, lihat Mempersiapkan akhir masa pakai Amazon Aurora Edisi Kompatibel MySQL versi 1.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.* dan 3.02.*.

Untuk membuat cluster dengan versi lama Aurora MySQL, tentukan versi engine melalui RDS Console, AWS CLI, atau Amazon RDS API.

> **Note**
>
> Versi ini ditetapkan sebagai rilis dukungan jangka panjang (LTS). Untuk informasi selengkapnya, lihat Rilis dukungan jangka panjang (LTS) Aurora MySQL di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan keamanan:

Perbaikan dan penyempurnaan lain untuk penanganan fine-tune di lingkungan terkelola. Di bawah ini adalah beberapa perbaikan CVE tambahan:

- CVE-2020-14559
- CVE-2020-14539
- CVE-2020-2579
- CVE-2020-2812
- CVE-2020-2780
- CVE-2020-2763

Perubahan tidak kompatibel:

Versi ini memperkenalkan perubahan izin yang memengaruhi perilaku perintah `mysqldump`. Pengguna harus memiliki hak istimewa `PROCESS` untuk mengakses tabel `INFORMATION_SCHEMA.FILES`. Untuk menjalankan perintah `mysqldump` tanpa perubahan apa pun, berikan hak istimewa `PROCESS` untuk pengguna basis data yang terhubung dengan perintah `mysqldump`. Anda juga dapat menjalankan perintah `mysqldump` dengan opsi `--no-tablespaces`. Dengan opsi tersebut, output `mysqldump` tidak menyertakan pernyataan `CREATE LOGFILE`

GROUP atau CREATE TABLESPACE apa pun. Dalam hal ini, perintah mysqldump tidak mengakses tabel INFORMATION_SCHEMA.FILES, dan Anda tidak perlu memberikan izin PROCESS.

Perbaikan ketersediaan:

- Memperbaiki masalah yang mungkin menyebabkan server memulai ulang selama pemulihan pernyataan DDL yang tidak di-commit.
- Memperbaiki kondisi balapan di manajer kunci yang dapat menyebabkan server memulai ulang.
- Memperbaiki masalah yang mungkin menyebabkan agen pemantau memulai ulang server selama pemulihan transaksi besar

Perbaikan umum:

- Mengubah perilaku untuk memetakan MIXED binlog_format ke ROW alih-alih STATEMENT saat menjalankan LOAD DATA FROM INFILE | S3.
- Memperbaiki masalah di mana replika binlog yang terhubung ke primer binlog Aurora MySQL mungkin menampilkan data yang tidak lengkap ketika primer yang dijalankan LOAD DATA FROM S3 dan binlog_format diatur ke STATEMENT.

Integrasi perbaikan bug MySQL Community Edition

- Bug #26654685: Indeks ID rusak yang ditemukan selama pemeriksaan kunci asing memunculkan pernyataan
- Bug #15831300: Secara default, ketika mempromosikan integer dari tipe yang lebih kecil pada master ke tipe yang lebih besar pada slave (misalnya, dari kolom SMALLINT pada master ke kolom BIGINT pada slave), nilai yang dipromosikan diperlakukan seolah-olah ditandatangani. Dalam kasus seperti itu, perilaku ini dapat dimodifikasi atau ditimpa menggunakan ALL_SIGNED, ALL_UNSIGNED baik salah satu maupun keduanya dalam set nilai yang ditetapkan untuk variabel sistem server slave_type_conversions. Untuk informasi selengkapnya, lihat Replikasi berbasis baris: promosi dan demosi atribut, serta deskripsi variabel.
- Bug #17449901: Dengan foreign_key_checks=0, InnoDB mengizinkan penghapusan indeks yang diperlukan oleh batasan kunci asing, sehingga tabel menjadi tidak konsisten dan menyebabkan kegagalan pemeriksaan kunci asing yang terjadi di pemuatan tabel. InnoDB sekarang mencegah penghapusan indeks yang diperlukan oleh batasan kunci asing, sekalipun dengan foreign_key_checks=0. Batasan kunci asing harus dibuang sebelum menghapus indeks kunci asing.

- BUG #20768847: ALTER TABLE ... Operasi DROP INDEX pada tabel dengan dependensi kunci asing memunculkan pernyataan.

Pembaruan mesin basis data Aurora MySQL 2020-03-05 (versi 1.22.2) (Dihentikan)

Versi: 1.22.2

Aurora MySQL 1.22.2 tersedia secara umum. Aurora MySQL versi 1.* kompatibel dengan MySQL 5.6 dan Aurora MySQL versi 2.* kompatibel dengan MySQL 5.7.

Versi mesin ini dijadwalkan akan dihentikan pada 28 Februari 2023. Untuk informasi selengkapnya, lihat Mempersiapkan akhir masa pakai Amazon Aurora Edisi Kompatibel MySQL versi 1.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.* dan 3.02.*.

Untuk membuat cluster dengan versi lama Aurora MySQL, harap tentukan versi mesin melalui Konsol RDS, AWS CLI, atau Amazon RDS API.

> **Note**
>
> Versi ini saat ini tidak tersedia di wilayah berikut: AWS GovCloud (AS-Timur) [us-gov-east-1], AWS GovCloud (AS-Barat) [us-gov-west-1]. Akan ada pengumuman terpisah setelah tersedia.
>
> Versi ini ditetapkan sebagai rilis dukungan jangka panjang (LTS). Untuk informasi selengkapnya, lihat Rilis dukungan jangka panjang (LTS) Aurora MySQL di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan prioritas tinggi:

- Memperbaiki masalah kegagalan koneksi yang tidak stabil setelah rotasi sertifikat.

- Memperbaiki masalah yang menyebabkan kloning memakan waktu lebih lama di beberapa klaster basis data dengan beban tulis yang tinggi.
- Memperbaiki masalah yang merusak replikasi logis ketika parameter `binlog_checksum` diatur ke nilai yang berbeda pada master dan replika.
- Memperbaiki masalah yang menyebabkan log lambat dan log umum mungkin tidak berotasi dengan benar pada replika baca.
- Memperbaiki masalah pada perilaku ANSI Read Committed Isolation Level.

Pembaruan mesin basis data Aurora MySQL 2019-12-23 (versi 1.22.1) (Dihentikan)

Versi: 1.22.1

Aurora MySQL 1.22.1 tersedia secara umum. Aurora MySQL versi 1.* kompatibel dengan MySQL 5.6 dan Aurora MySQL versi 2.* kompatibel dengan MySQL 5.7.

Versi mesin ini dijadwalkan akan dihentikan pada 28 Februari 2023. Untuk informasi selengkapnya, lihat Mempersiapkan akhir masa pakai Amazon Aurora Edisi Kompatibel MySQL versi 1.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.* dan 3.02.*.

Untuk membuat cluster dengan versi lama dari Aurora MySQL, harap tentukan versi mesin melalui, atau RDS API Konsol Manajemen AWS. AWS CLI Anda memiliki opsi untuk meningkatkan klaster basis data Aurora MySQL 1.* yang ada ke Aurora MySQL 1.22.1.

> **Note**
>
> Versi ini saat ini tidak tersedia di AWS Wilayah berikut: AWS GovCloud (AS-Timur) [us-gov-east-1], (AS-Barat) [-1], Tiongkok AWS GovCloud (Ningxia) [cn-barat laut us-gov-west 1], Asia Pasifik (Hong Kong) [ap-east-1], dan Timur Tengah (Bahrain) [me-south-1. Akan ada pengumuman terpisah setelah tersedia.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Prosedur untuk meningkatkan klaster DB Anda telah berubah. Untuk informasi selengkapnya, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan penting:

- Memperbaiki masalah yang mencegah pemulihan mesin yang melibatkan kunci tabel dan tabel sementara.
- Meningkatkan stabilitas log biner saat tabel sementara digunakan.

Perbaikan prioritas tinggi:

- Memperbaiki kebocoran memori yang lambat di sistem pelacakan basis data dan pencatatan log subsistem Aurora tertentu yang menurunkan memori yang dapat dilepas.

Pembaruan mesin basis data Aurora MySQL 2019-11-25 (versi 1.22.0) (Dihentikan)

Versi: 1.22.0

Aurora MySQL 1.22.0 tersedia secara umum. Aurora MySQL versi 1.* kompatibel dengan MySQL 5.6 dan Aurora MySQL versi 2.* kompatibel dengan MySQL 5.7.

Versi mesin ini dijadwalkan akan dihentikan pada 28 Februari 2023. Untuk informasi selengkapnya, lihat Mempersiapkan akhir masa pakai Amazon Aurora Edisi Kompatibel MySQL versi 1.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.* dan 3.02.*.

Untuk membuat cluster dengan versi lama dari Aurora MySQL, harap tentukan versi mesin melalui, atau RDS API Konsol Manajemen AWS. AWS CLI Anda memiliki opsi untuk meningkatkan klaster basis data Aurora MySQL 1.* yang ada ke Aurora MySQL 1.22.0.

> **Note**
>
> Versi ini saat ini tidak tersedia di AWS Wilayah berikut: AWS GovCloud (AS-Timur) [us-gov-east-1], (AS-Barat) [-1], Tiongkok AWS GovCloud (Ningxia) [cn-barat laut us-gov-west 1], Asia Pasifik (Hong Kong) [ap-east-1], Timur Tengah (Bahrain) [me-south-1], dan Amerika Selatan (São Paulo) [sa-sa-east-1]. Akan ada pengumuman terpisah setelah tersedia.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Prosedur untuk meningkatkan klaster DB Anda telah berubah. Untuk informasi selengkapnya, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Fitur baru:

- Klaster MySQL Aurora kini mendukung jenis instans r5.8xlarge, r5.16xlarge, dan r5.24xlarge.
- Binlog memiliki peningkatan baru untuk latensi waktu penerapan yang lebih baik ketika transaksi yang sangat besar dilibatkan.
- Aurora MySQL sekarang memiliki mekanisme untuk meminimalkan durasi waktu penulisan peristiwa transaksi besar ke binlog saat commit. Hal ini secara efektif mencegah pemulihan offline yang lama saat kerusakan basis data mengalami crash selama durasi tersebut. Fitur ini juga memperbaiki masalah ketika transaksi besar memblokir transaksi kecil pada commit binlog. Fitur ini dimatikan secara default dan dapat diaktifkan oleh tim layanan jika diperlukan untuk beban kerja Anda. Saat diaktifkan, fitur ini akan terpicu ketika ukuran transaksi >500MB.
- Menambahkan dukungan untuk tingkat isolasi `READ COMMITTED` ANSI pada replika baca. Tingkat isolasi ini memungkinkan kueri yang berlangsung lama pada replika baca untuk mengeksekusi tanpa memengaruhi throughput tulis yang tinggi pada simpul penulis. Untuk informasi selengkapnya, lihat Tingkat isolasi Aurora MySQL.

- Database Global sekarang memungkinkan penambahan wilayah replika hanya-baca sekunder untuk kluster basis data yang digunakan di Wilayah ini: AWS wilayah: AS Timur (Virginia N.) [us-timur-1], AS Timur (Ohio) [us-timur-2], AS Barat (California N.) [us-west-1], AS Barat (Oregon) [us-west-2], Eropa (Irlandia) [eu-west-1], Eropa (London) [eu-west-2], Eropa (Paris) [eu-west-3], Asia Pasifik (Tokyo) [ap-northeast-1], Asia Pasifik (Tokyo) [ap-northeast-1], Asia Pasifik (Seoul) [ap-northeast-2], Asia Pasifik (Singapura) [ap-southeast-1], Asia Pasifik (Sydney) [ap-southeast-2], Kanada (Tengah) [ca-central-1], Eropa (Frankfurt) [eu-central-1], dan Asia Pasifik (Mumbai) [ap-south-1].
- Fitur hot row contention sudah tersedia secara umum dan mode lab Aurora tidak perlu diatur ke ON. Fitur ini meningkatkan secara signifikan throughput untuk beban kerja dengan banyak transaksi yang bersaing untuk baris di halaman yang sama.
- Versi ini telah memperbarui file zona waktu untuk mendukung pembaruan zona waktu terbaru Brasil untuk klaster baru.

Perbaikan penting:

- CVE-2019-2922
- CVE-2019-2923
- CVE-2019-2924
- CVE-2019-2910

Perbaikan prioritas tinggi:

- CVE-2019-2805
- CVE-2019-2730
- CVE-2019-2740
- CVE-2018-3064
- CVE-2018-3058
- CVE-2017-3653
- CVE-2017-3464
- CVE-2017-3244
- CVE-2016-5612
- CVE-2016-5439

- CVE-2016-0606
- CVE-2015-4904
- CVE-2015-4879
- CVE-2015-4864
- CVE-2015-4830
- CVE-2015-4826
- CVE-2015-2620
- CVE-2015-0382
- CVE-2015-0381
- CVE-2014-6555
- CVE-2014-4258
- CVE-2014-4260
- CVE-2014-2444
- CVE-2014-2436
- CVE-2013-5881
- CVE-2014-0393
- CVE-2013-5908
- CVE-2013-5807
- CVE-2013-3806
- CVE-2013-3811
- CVE-2013-3804
- CVE-2013-3807
- CVE-2013-2378
- CVE-2013-2375
- CVE-2013-1523
- CVE-2013-2381
- CVE-2012-5615
- CVE-2014-6489

- Memperbaiki masalah dalam komponen pemulihan DDL yang menghasilkan waktu henti basis data yang lama. Klaster yang menjadi tidak tersedia setelah menjalankan kueri `TRUNCATE TABLE` di tabel dengan kolom `AUTO_INCREMENT` harus diperbarui.
- Memperbaiki masalah dalam komponen pemulihan DDL yang menghasilkan waktu henti basis data yang lama. Klaster yang menjadi tidak tersedia setelah mengeksekusi kueri `DROP TABLE` pada beberapa tabel secara paralel harus diperbarui.

Perbaikan stabilitas umum:

- Memperbaiki masalah yang menyebabkan replika baca memulai ulang selama transaksi jangka panjang. Pelanggan yang mengalami mulai ulang replika yang bertepatan dengan penurunan yang lebih cepat pada memori yang dapat dilepas sebaiknya mempertimbangkan untuk meningkatkan ke versi ini.
- Memperbaiki masalah kesalahan pelaporan `ERROR 1836` ketika kueri bersarang dijalankan terhadap tabel sementara di replika baca.
- Memperbaiki kesalahan kueri paralel pada instans pembaca Aurora ketika beban kerja tulis yang berat berjalan di instans tulis Aurora.
- Memperbaiki masalah yang menyebabkan basis data yang dikonfigurasi sebagai Master Binlog memulai ulang selama beban kerja tulis yang berat sedang berjalan.
- Memperbaiki masalah ketidaktersediaan berkepanjangan saat memulai ulang mesin. Perbaikan ini mengatasi masalah dalam inisialisasi pool buffer. Masalah ini sangat jarang terjadi tetapi dapat berpotensi memengaruhi rilis apa pun yang didukung.
- Memperbaiki masalah yang menghasilkan data yang tidak konsisten di tabel `information_schema.replica_host_status`.
- Memperbaiki kondisi balapan antara kueri paralel dan jalur eksekusi standar yang menyebabkan simpul Pembaca memulai ulang secara tidak pasti.
- Meningkatkan stabilitas basis data ketika jumlah koneksi klien melebihi nilai parameter `max_connections`.
- Meningkatkan stabilitas instans pembaca dengan memblokir DDL dan kueri `LOAD FROM S3` yang tidak didukung.

Integrasi perbaikan bug MySQL Community Edition

- Bug#16346241 - KERUSAKAN SERVER DALAM ITEM_PARAM::QUERY_VAL_STR

- Bug#17733850 - NAME_CONST() CRASH DI ITEM_NAME_CONST::ITEM_NAME_CONST()
- Bug #20989615 - INNODB AUTO_INCREMENT MENGHASILKAN NILAI YANG SAMA DUA KALI
- Bug #20181776 - PENGENDALIAN AKSES TIDAK COCOK DENGAN SEBAGIAN BESAR HOST SPESIFIK SAAT MEMUAT WILDCARD
- Bug #27326796 - MYSQL CRASH DENGAN KEGAGALAN PERNYATAAN INNODB DI FILE PARS.CC PARS0
- Bug #20590013 - JIKA ANDA MEMILIKI INDEX TEKS PENUH DAN MELETAKKANNYA ANDA TIDAK LAGI DAPAT MELAKUKAN DDL ONLINE

Pembaruan mesin basis data Aurora MySQL 2019-11-25 (versi 1.21.0) (Dihentikan)

Versi: 1.21.0

Aurora MySQL 1.21.0 tersedia secara umum. Aurora MySQL versi 1.* kompatibel dengan MySQL 5.6 dan Aurora MySQL versi 2.* kompatibel dengan MySQL 5.7.

Rilis Aurora MySQL yang didukung saat ini adalah 1.14.*, 1.15.*, 1.16.*, 1.17.*, 1.18.*, 1.19.*, 1.20.*, 1.21.*, 1.22.*, 2.01.*, 2.02.*, 2.03.*, 2.04.*, 2.05.*, 2.06.*, dan 2.07.*. Untuk membuat cluster dengan versi lama dari Aurora MySQL, harap tentukan versi mesin melalui, atau RDS API Konsol Manajemen AWS. AWS CLI Anda memiliki opsi untuk meningkatkan klaster basis data Aurora MySQL 1.* yang ada ke Aurora MySQL 1.21.0.

> **Note**
>
> Versi ini saat ini tidak tersedia di AWS Wilayah berikut: AWS GovCloud (AS-Timur) [us-gov-east-1], (AS-Barat) [-1], Tiongkok AWS GovCloud (Ningxia) [cn-barat laut us-gov-west 1], Asia Pasifik (Hong Kong) [ap-east-1], Eropa (Stockholm) [eu-north-1], dan Timur Tengah (Bahrain) [me-south-1] me-south-1]. Akan ada pengumuman terpisah setelah tersedia.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Prosedur untuk meningkatkan klaster DB Anda telah berubah. Untuk informasi selengkapnya, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan penting:

- CVE-2018-0734
- CVE-2019-2534
- CVE-2018-2612
- CVE-2017-3599
- CVE-2018-2562
- CVE-2017-3329
- CVE-2018-2696
- CVE-2015-4737

Perbaikan prioritas tinggi:

- Pelanggan dengan ukuran basis data yang mendekati 64 tebibyte (TiB) sangat disarankan untuk melakukan peningkatan ke versi ini untuk menghindari waktu henti akibat bug stabilitas yang memengaruhi volume yang mendekati batas penyimpanan Aurora.

Perbaikan stabilitas umum:

- Memperbaiki kesalahan kueri paralel pada instans pembaca Aurora ketika beban kerja tulis yang berat berjalan di instans tulis Aurora.
- Memperbaiki masalah pada instans pembaca Aurora yang mengurangi memori bebas selama transaksi jangka panjang saat ada lalu lintas commit transaksi yang berat di instans penulis.
- Nilai parameter `aurora_disable_hash_join` sekarang tetap bertahan setelah memulai ulang basis data atau penggantian host.

- Memperbaiki masalah yang terkait dengan cache Pencarian Teks Penuh yang menyebabkan instans Aurora kehabisan memori. Pelanggan yang menggunakan Pencarian Teks Penuh harus melakukan peningkatan.
- Meningkatkan stabilitas basis data saat fitur sambungan hash diaktifkan dan instans memiliki memori yang rendah. Pelanggan yang menggunakan sambungan hash harus melakukan peningkatan.
- Memperbaiki masalah pada cache kueri dengan kesalahan “Terlalu banyak koneksi” yang dapat menyebabkan mulai ulang.
- Memperbaiki perhitungan memori bebas pada instans T2 dengan menyertakan pertukaran ruang memori untuk mencegah booting ulang yang tidak perlu.

Integrasi perbaikan bug MySQL Community Edition

- Bug #19929406: HANDLE_FATAL_SIGNAL (SIG=11) DI __MEMMOVE_ _KEMBALI DARI STRING: :COPY SSSE3
- Bug #17059925: Untuk pernyataan UNION, nilai yang diperiksa baris tidak dihitung dengan benar. Ini diwujudkan sebagai nilai yang terlalu besar untuk kolom `ROWS_EXAMINED` dari tabel pernyataan Skema Kinerja (seperti vents_statements_current).
- Bug #11827369: Beberapa kueri dengan subkueri bersarang `SELECT ... FROM DUAL` memunculkan pernyataan.
- Bug #16311231: Hasil yang salah dikembalikan apabila sebuah mengandung subkueri dalam klausa `IN` yang memuat operasi XOR di klausa `WHERE`.

Pembaruan mesin basis data Aurora MySQL 2020-03-05 (versi 1.20.1) (Dihentikan)

Versi: 1.20.1

Aurora MySQL 1.20.1 tersedia secara umum. Aurora MySQL versi 1.* kompatibel dengan MySQL 5.6 dan Aurora MySQL versi 2.* kompatibel dengan MySQL 5.7.

Rilis Aurora MySQL yang didukung saat ini adalah 1.14.*, 1.15.*, 1.16.*, 1.17.*, 1.18.*, 1.19.*, 1.20.*, 1.21.*, 1.22.*, 2.01.*, 2.02.*, 2.03.*, 2.04.*, 2.05.*, 2.06.*, dan 2.07.*. Anda dapat memulihkan snapshot basis data Aurora MySQL 1.* ke Aurora MySQL 1.20.1.

Untuk membuat cluster dengan versi lama Aurora MySQL, harap tentukan versi mesin melalui Konsol RDS, AWS CLI, atau Amazon RDS API.

> **Note**
>
> Versi ini saat ini tidak tersedia di wilayah berikut: AWS GovCloud (AS-Timur) [us-gov-east-1], AWS GovCloud (AS-Barat) [us-gov-west-1]. Akan ada pengumuman terpisah setelah tersedia.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan prioritas tinggi:

- Memperbaiki masalah kegagalan koneksi yang tidak stabil setelah rotasi sertifikat.
- Memperbaiki masalah yang terkait dengan konkurensi penutupan koneksi yang akan mengakibatkan failover di bawah beban kerja yang berat.

Perbaikan stabilitas umum:

- Memperbaiki kegagalan selama eksekusi kueri kompleks yang melibatkan sambungan multitabel dan agregasi yang menggunakan tabel menengah secara internal.

Pembaruan mesin basis data Aurora MySQL 2019-11-11 (versi 1.20.0) (Dihentikan)

Versi: 1.20.0

Aurora MySQL 1.20.0 tersedia secara umum. Aurora MySQL versi 1.* kompatibel dengan MySQL 5.6 dan Aurora MySQL versi 2.* kompatibel dengan MySQL 5.7.

Rilis Aurora MySQL yang didukung saat ini adalah 1.14.*, 1.15.*, 1.16.*, 1.17.*, 1.18.*, 1.19.*, 1.20.*, 2.01.*, 2.02.*, 2.03.*, dan 2.04.*. Untuk membuat cluster dengan versi lama dari Aurora MySQL, harap tentukan versi mesin melalui, atau RDS API Konsol Manajemen AWS. AWS CLI Anda memiliki

opsi untuk meningkatkan klaster basis data Aurora MySQL 1.* yang ada, hingga 1.19.5, ke Aurora MySQL 1.20.0.

> **Note**
>
> Versi ini saat ini tidak tersedia di AWS Wilayah berikut: AWS GovCloud (AS-Timur) [us-gov-east-1], (AS-Barat) [-1], Tiongkok AWS GovCloud (Ningxia) [cn-barat laut us-gov-west 1], Asia Pasifik (Hong Kong) [ap-east-1], Eropa (Stockholm) [eu-north-1], dan Timur Tengah (Bahrain) [me-south-1] me-south-1]. Akan ada pengumuman terpisah setelah tersedia.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Prosedur untuk meningkatkan klaster DB Anda telah berubah. Untuk informasi selengkapnya, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan penting:

- CVE-2018-0734
- CVE-2019-2534
- CVE-2018-2612
- CVE-2017-3599
- CVE-2018-2562
- CVE-2017-3329
- CVE-2018-2696
- CVE-2015-4737

Perbaikan prioritas tinggi:

- Pelanggan dengan ukuran basis data yang mendekati 64 tebibyte (TiB) sangat disarankan untuk melakukan peningkatan ke versi ini untuk menghindari waktu henti akibat bug stabilitas yang memengaruhi volume yang mendekati batas penyimpanan Aurora.

Perbaikan stabilitas umum:

- Memperbaiki kesalahan kueri paralel pada instans pembaca Aurora ketika beban kerja tulis yang berat berjalan di instans tulis Aurora.
- Memperbaiki masalah pada instans pembaca Aurora yang mengurangi memori bebas selama transaksi jangka panjang saat ada lalu lintas commit transaksi yang berat di instans penulis.
- Nilai parameter `aurora_disable_hash_join` sekarang tetap bertahan setelah memulai ulang basis data atau penggantian host.
- Memperbaiki masalah yang terkait dengan cache Pencarian Teks Penuh yang menyebabkan instans Aurora kehabisan memori. Pelanggan yang menggunakan Pencarian Teks Penuh harus melakukan peningkatan.
- Meningkatkan stabilitas basis data saat fitur sambungan hash diaktifkan dan instans memiliki memori yang rendah. Pelanggan yang menggunakan sambungan hash harus melakukan peningkatan.
- Memperbaiki masalah pada cache kueri dengan kesalahan “Terlalu banyak koneksi” yang dapat menyebabkan mulai ulang.
- Memperbaiki perhitungan memori bebas pada instans T2 dengan menyertakan pertukaran ruang memori untuk mencegah booting ulang yang tidak perlu.

Integrasi perbaikan bug MySQL Community Edition

- Bug #19929406: HANDLE_FATAL_SIGNAL (SIG=11) DI __MEMMOVE_ _KEMBALI DARI STRING: :COPY SSSE3
- Bug #17059925: Untuk pernyataan UNION, nilai yang diperiksa baris tidak dihitung dengan benar. Ini diwujudkan sebagai nilai yang terlalu besar untuk kolom `ROWS_EXAMINED` dari tabel pernyataan Skema Kinerja (seperti vents_statements_current).
- Bug #11827369: Beberapa kueri dengan subkueri bersarang `SELECT ... FROM DUAL` memunculkan pernyataan.
- Bug #16311231: Hasil yang salah dikembalikan apabila sebuah mengandung subkueri dalam klausa `IN` yang memuat operasi XOR di klausa `WHERE`.

Pembaruan mesin basis data Aurora MySQL 2020-03-05 (versi 1.19.6) (Dihentikan)

Versi: 1.19.6

Aurora MySQL 1.19.6 tersedia secara umum. Aurora MySQL versi 1.* kompatibel dengan MySQL 5.6 dan Aurora MySQL versi 2.* kompatibel dengan MySQL 5.7.

Versi mesin ini dijadwalkan akan dihentikan pada 28 Februari 2023. Untuk informasi selengkapnya, lihat Mempersiapkan akhir masa pakai Amazon Aurora Edisi Kompatibel MySQL versi 1.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Anda dapat memulihkan snapshot basis data Aurora MySQL 1.* ke Aurora MySQL 1.19.6.

Untuk membuat cluster dengan versi lama Aurora MySQL, harap tentukan versi mesin melalui Konsol RDS, AWS CLI, atau Amazon RDS API.

> **Note**
>
> Versi ini saat ini tidak tersedia di wilayah berikut: AWS GovCloud (AS-Timur) [us-gov-east-1], AWS GovCloud (AS-Barat) [us-gov-west-1]. Akan ada pengumuman terpisah setelah tersedia.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan prioritas tinggi:

- Memperbaiki masalah kegagalan koneksi yang tidak stabil setelah rotasi sertifikat.

Pembaruan mesin basis data Aurora MySQL 2019-09-19 (versi 1.19.5) (Dihentikan)

Versi: 1.19.5

Aurora MySQL 1.19.5 tersedia secara umum. Aurora MySQL versi 1.* kompatibel dengan MySQL 5.6 dan Aurora MySQL versi 2.* kompatibel dengan MySQL 5.7.

Versi mesin ini dijadwalkan akan dihentikan pada 28 Februari 2023. Untuk informasi selengkapnya, lihat Mempersiapkan akhir masa pakai Amazon Aurora Edisi Kompatibel MySQL versi 1.

Rilis Aurora MySQL yang saat ini didukung adalah 1.19.5, 1.19.6, 1.22.*, 1.23.*, 2.04.*, 2.07.*, 2.08.*, 2.09.*, 2.10.*, 3.01.*, dan 3.02.*.

Anda memiliki opsi untuk meningkatkan klaster basis data yang ada ke Aurora MySQL 1.19.5. Anda dapat memulihkan snapshot Aurora MySQL 1.14.*, 1.15.*, 1.16.*, 1.17.*, 1.18.*, 1.19.1, dan 1.19.2 ke Aurora MySQL 1.19.5.

Untuk menggunakan versi lama Aurora MySQL, Anda dapat membuat cluster database baru dengan menentukan versi mesin melalui, API, atau RDS. Konsol Manajemen AWS AWS CLI

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Versi ini saat ini tidak tersedia di AWS Wilayah berikut: Eropa (London) [eu-west-2] AWS GovCloud , (AS-Timur) [-1], (AS-Barat) us-gov-east [-1], Tiongkok AWS GovCloud (Ningxia) [us-gov-westcn-barat-1], dan Asia Pasifik (Hong Kong) [ap-east-1]. Akan ada pengumuman terpisah setelah tersedia.

> **Note**
>
> Prosedur untuk meningkatkan klaster DB Anda telah berubah. Untuk informasi selengkapnya, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

- Memperbaiki masalah pada instans pembaca Aurora yang mengurangi memori bebas selama transaksi jangka panjang saat ada lalu lintas commit transaksi yang berat di instans penulis.
- Memperbaiki kesalahan kueri paralel pada instans pembaca Aurora ketika beban kerja tulis yang berat berjalan di instans tulis Aurora.
- Nilai parameter `aurora_disable_hash_join` sekarang tetap dipertahankan setelah mulai ulang basis data atau penggantian host.
- Memperbaiki masalah yang terkait dengan cache Pencarian Teks Penuh yang menyebabkan instans Aurora kehabisan memori.
- Meningkatkan stabilitas basis data ketika ukuran volume mendekati batas volume 64 tebibyte (TiB) dengan mencadangkan ruang berukuran 160 GB untuk alur kerja pemulihan guna menyelesaikan tanpa failover.
- Meningkatkan stabilitas basis data saat fitur sambungan hash diaktifkan dan instans memiliki memori yang rendah.
- Memperbaiki perhitungan memori bebas dengan menyertakan pertukaran ruang memori pada instans T2 yang menyebabkan instans mengalami booting ulang sebelum waktunya.
- Memperbaiki masalah pada cache kueri dengan kesalahan “Terlalu banyak koneksi” yang dapat menyebabkan mulai ulang.

Integrasi perbaikan bug MySQL Community Edition

- CVE-2018-2696
- CVE-2015-4737
- Bug #19929406: HANDLE_FATAL_SIGNAL (SIG=11) DI __MEMMOVE_ _KEMBALI DARI STRING: :COPY SSSE3
- Bug #17059925: Untuk pernyataan UNION, nilai yang diperiksa baris tidak dihitung dengan benar. Ini diwujudkan sebagai nilai yang terlalu besar untuk kolom `ROWS_EXAMINED` dari tabel pernyataan Skema Kinerja (seperti vents_statements_current).
- Bug #11827369: Beberapa kueri dengan subkueri bersarang `SELECT ... FROM DUAL` memunculkan pernyataan.
- Bug #16311231: Hasil yang salah dikembalikan apabila sebuah mengandung subkueri dalam klausa `IN` yang memuat operasi XOR di klausa `WHERE`.

Pembaruan mesin basis data Aurora MySQL 2019-06-05 (versi 1.19.2) (Dihentikan)

Versi: 1.19.2

Aurora MySQL 1.19.2 tersedia secara umum. Semua klaster basis data Aurora MySQL baru dengan kompatibilitas MySQL 5.6, termasuk yang dipulihkan dari snapshot, dapat dibuat dengan 1.17.8, 1.19.0, 1.19.1, atau 1.19.2. Anda memiliki opsi, tetapi tidak diharuskan, untuk meningkatkan klaster basis data yang ada ke Aurora MySQL 1.19.2. Untuk menggunakan versi yang lebih lama, Anda dapat membuat klaster basis data baru di Aurora MySQL 1.14.4, Aurora MySQL 1.15.1, Aurora MySQL 1.16, Aurora MySQL 1.17.8, atau Aurora MySQL 1.18. Anda dapat melakukannya menggunakan AWS CLI atau Amazon RDS API dan menentukan versi mesin.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Versi ini saat ini tidak tersedia di Wilayah AWS GovCloud (AS-Barat) [us-gov-west-1], Eropa (Stockholm) [eu-north-1], Tiongkok (Ningxia) [cn-northwest-1], dan Asia Pasifik (Hong Kong) [ap-east-1]. AWS Akan ada pengumuman terpisah setelah tersedia.

> **Note**
>
> Prosedur untuk meningkatkan klaster DB Anda telah berubah. Untuk informasi selengkapnya, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

- Memperbaiki masalah yang dapat menyebabkan kegagalan saat memuat data ke Aurora dari Amazon S3.
- Memperbaiki masalah yang dapat menyebabkan kegagalan saat mengunggah data dari Aurora ke Amazon S3.

- Memperbaiki masalah yang menyebabkan sesi zombie yang dibiarkan dalam status dimatikan.
- Memperbaiki masalah yang menyebabkan koneksi yang dibatalkan saat menangani kesalahan dalam manajemen protokol jaringan.
- Memperbaiki masalah yang dapat menyebabkan crash saat menangani tabel yang dipartisi.
- Memperbaiki masalah terkait replikasi binlog dari pembuatan pemicu.

Pembaruan mesin basis data Aurora MySQL 2019-05-09 (versi 1.19.1) (Dihentikan)

Versi: 1.19.1

Aurora MySQL 1.19.1 tersedia secara umum. Semua klaster basis data Aurora MySQL baru dengan kompatibilitas MySQL 5.6, termasuk yang dipulihkan dari snapshot, dapat dibuat dengan 1.17.8, 1.19.0, atau 1.19.1. Anda memiliki opsi, tetapi tidak diharuskan, untuk meningkatkan klaster basis data yang ada ke Aurora MySQL 1.19.1. Untuk menggunakan versi yang lebih lama, Anda dapat membuat klaster basis data baru di Aurora MySQL 1.14.4, Aurora MySQL 1.15.1, Aurora MySQL 1.16, Aurora MySQL 1.17.8, atau Aurora MySQL 1.18. Anda dapat melakukannya menggunakan AWS CLI atau Amazon RDS API dan menentukan versi mesin.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Versi ini saat ini tidak tersedia di wilayah AWS GovCloud (AS-Barat) [us-gov-west-1] dan Tiongkok (Beijing) [cn-north-1]. Akan ada pengumuman terpisah setelah tersedia.

> **Note**
>
> Prosedur untuk meningkatkan klaster DB Anda telah berubah. Untuk informasi selengkapnya, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

- Memperbaiki bug dalam replikasi binlog yang dapat menyebabkan masalah pada instans Aurora yang dikonfigurasi sebagai pekerja binlog.
- Memperbaiki kesalahan dalam menangani jenis `ALTER TABLE` perintah.
- Memperbaiki masalah terkait koneksi yang dibatalkan karena kesalahan dalam manajemen protokol jaringan.

Pembaruan mesin basis data Aurora MySQL 2019-02-07 (versi 1.19.0) (Dihentikan)

Versi: 1.19.0

Aurora MySQL 1.19.0 tersedia secara umum. Semua klaster basis data Aurora MySQL baru dengan kompatibilitas MySQL 5.6, termasuk yang dipulihkan dari snapshot, dapat dibuat dengan 1.17.8 atau 1.19.0. Anda memiliki opsi, tetapi tidak diharuskan, untuk meningkatkan klaster basis data yang ada ke Aurora MySQL 1.19.0. Untuk menggunakan versi yang lebih lama, Anda dapat membuat klaster basis data baru di Aurora MySQL 1.14.4, Aurora MySQL 1.15.1, Aurora MySQL 1.16, Aurora MySQL 1.17.8, atau Aurora MySQL 1.18.0. Anda dapat melakukannya menggunakan AWS CLI atau Amazon RDS API dan menentukan versi mesin.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Versi ini saat ini tidak tersedia di wilayah AWS GovCloud (AS-Barat) [us-gov-west-1] dan Tiongkok (Beijing) [cn-north-1]. Akan ada pengumuman terpisah setelah tersedia.

> **Note**
>
> Prosedur untuk meningkatkan klaster DB Anda telah berubah. Untuk informasi selengkapnya, lihat Meningkatkan versi kecil atau tingkat patch klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Fitur

- Pemilih Versi Aurora - Dimulai dengan Aurora MySQL 1.19.0, Anda dapat memilih salah satu dari beberapa versi Aurora yang kompatibel dengan MySQL 5.6 di konsol Amazon RDS. Untuk informasi selengkapnya, lihat Memeriksa atau menentukan versi mesin Aurora MySQL melalui AWS di Panduan Pengguna Amazon Aurora.

Perbaikan

- Memperbaiki masalah stabilitas yang terkait dengan kueri `CHECK TABLE` di Replika Aurora.
- Memperkenalkan variabel pengguna global baru `aurora_disable_hash_join` untuk menonaktifkan Sambungan Hash.
- Memperbaiki masalah stabilitas saat membuat baris output selama beberapa kali sambungan hash.
- Memperbaiki masalah yang menyebabkan kesalahan hasil karena perubahan rencana selama pemeriksaan kelayakan Sambungan Hash.
- Zero Downtime Patching didukung dengan transaksi yang berjalan lama. Peningkatan ini akan mulai berlaku saat peningkatan dari versi 1.19 ke versi yang lebih tinggi.
- Zero Downtime Patching sekarang didukung ketika binlog diaktifkan. Peningkatan ini akan mulai berlaku saat peningkatan dari versi 1.19 ke versi yang lebih tinggi.
- Memperbaiki masalah yang menyebabkan lonjakan penggunaan CPU di Replika Aurora yang tidak terkait dengan beban kerja.
- Memperbaiki kondisi balapan di manajer kunci yang mengakibatkan mulai ulang basis data.
- Memperbaiki kondisi balapan di komponen manajer kunci untuk meningkatkan stabilitas instans Aurora.
- Memperbaiki stabilitas detektor deadlock di dalam komponen manajer kunci.
- Operasi `INSERT` pada tabel dilarang jika InnoDB mendeteksi bahwa indeks rusak.
- Memperbaiki masalah stabilitas dalam DDL Cepat.
- Meningkatkan stabilitas Aurora dengan mengurangi konsumsi memori dalam pemindaian batch untuk subkueri baris tunggal.
- Memperbaiki masalah stabilitas yang terjadi setelah kunci asing dihilangkan ketika variabel sistem `foreign_key_checks` diatur ke 0.

- Memperbaiki masalah di fitur Out Of Memory Avoidance yang salah membatalkan perubahan ke nilai `table_definition_cache` yang dibuat pengguna.
- Memperbaiki masalah stabilitas dalam fitur Out Of Memory Avoidance.
- Memperbaiki masalah yang mengatur `query_time` dan `lock_time` di `slow_query_log` ke nilai sampah.
- Memperbaiki masalah stabilitas kueri paralel yang dipicu oleh penanganan yang tidak tepat terhadap kolasi string secara internal.
- Memperbaiki masalah stabilitas kueri paralel yang dipicu oleh pencarian indeks sekunder.
- Memperbaiki masalah stabilitas kueri paralel yang dipicu oleh pembaruan multitabel.

Integrasi perbaikan bug MySQL Community Edition

- BUG #32917: DETEKSI FILE TEMP-POOL ORPHAN, DAN TANGANI DENGAN MULUS
- BUG #63144 BUAT TABEL JIKA TIDAK ADA METADATA TERLALU MEMBATASI

Pembaruan mesin basis data Aurora MySQL 2018-09-20 (versi 1.18.0) (Dihentikan)

Versi: 1.18.0

Aurora MySQL 1.18.0 tersedia secara umum. Semua klaster kueri paralel Aurora MySQL baru dengan kompatibilitas MySQL 5.6, termasuk yang dipulihkan dari snapshot, akan dibuat dengan Aurora MySQL 1.18.0. Anda memiliki opsi, tetapi tidak diharuskan, untuk meningkatkan klaster kueri paralel yang ada ke Aurora MySQL 1.18.0. Anda dapat membuat klaster DB baru di Aurora MySQL 1.14.4, Aurora MySQL 1.15.1, Aurora MySQL 1.16, atau Aurora MySQL 1.17.6. Anda dapat melakukannya menggunakan AWS CLI atau Amazon RDS API dan menentukan versi mesin.

Dengan versi Aurora MySQL 1.18.0, kami menggunakan model patching klaster di mana semua simpul dalam klaster Aurora DB ditambal pada saat yang sama.

> **Important**
>
> Aurora MySQL 1.18.0 hanya berlaku untuk klaster kueri paralel Aurora. Jika Anda meningkatkan klaster 5.6.10a yang telah disediakan, versi yang dihasilkan adalah 1.17.8. Jika Anda meningkatkan klaster 5.6.10a kueri paralel, versi yang dihasilkan adalah 1.18.0.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Fitur

- Kueri Paralel tersedia dengan rilis ini, untuk klaster baru dan snapshot yang dipulihkan. Aurora MySQL parallel query adalah optimasi yang memparalelkan beberapa dan komputasi yang terlibat dalam pemrosesan kueri data-intensif I/O . Pekerjaan yang diparalelkan mencakup pengambilan baris dari penyimpanan, ekstraksi nilai kolom, dan penentuan baris yang cocok dengan ketentuan dalam klausa `WHERE` dan klausa sambungan. Pekerjaan sarat data ini didelegasikan (dalam istilah pengoptimalan basis data, ditekan) ke beberapa simpul di lapisan penyimpanan terdistribusi Aurora. Tanpa kueri paralel, setiap kueri membawa semua data yang dipindai ke simpul tunggal dalam klaster Aurora MySQL (simpul kepala) dan melakukan semua pemrosesan kueri di sana.
  - Saat fitur kueri paralel diaktifkan, mesin Aurora MySQL secara otomatis menentukan kapan kueri dapat menguntungkan, tanpa memerlukan perubahan SQL seperti petunjuk atau atribut tabel.

  Untuk informasi selengkapnya, lihat Menggunakan kueri paralel untuk Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.
- OOM Avoidance: Fitur ini memantau memori sistem dan melacak memori yang dikonsumsi oleh berbagai komponen basis data. Setelah sistem memiliki memori yang rendah, sistem melakukan sederet tindakan untuk melepaskan memori dari berbagai komponen yang dilacak dalam upaya menyimpan basis data agar tidak masuk ke status Out of Memory (OOM) dan dengan demikian menghindari mulai ulang basis data. Fitur upaya terbaik ini diaktifkan secara default untuk instans t2 dan dapat diaktifkan di kelas instans lain melalui parameter instans baru yang bernama `aurora_oom_response`. Parameter instans ini mengambil serangkaian tindakan yang dipisahkan koma yang harus dilakukan oleh instans saat memorinya rendah. Tindakan yang valid meliputi "print", "tune", "decline", "kill_query", atau kombinasinya. Setiap string kosong berarti tidak boleh ada tindakan yang diambil dan membuat fitur menjadi dinonaktifkan secara efektif. Perhatikan bahwa tindakan default untuk fitur ini adalah “print, tune”. Contoh penggunaan:
  - “print” – Hanya mencetak kueri yang mengambil memori dalam jumlah besar.
  - “tune” – Menyesuaikan cache tabel internal untuk melepas beberapa memori kembali ke sistem.
  - “decline” – Menolak kueri baru setelah memori instans rendah.
  - “kill_query” – Mematikan kueri dalam urutan konsumsi memori secara menurun hingga permukaan memori instans di atas ambang batas rendah. Pernyataan data definition language (DDL) tidak dimatikan.

- "print, tune" – Melakukan tindakan yang dijelaskan untuk “print” dan “tune”.
- "tune, decline, kill_query" – Melakukan tindakan yang dijelaskan untuk "tune", "decline", dan "kill_query".

Untuk informasi tentang out-of-memory kondisi penanganan dan saran pemecahan masalah lainnya, lihat Amazon Aurora MySQL masalah kehabisan memori di Panduan Pengguna Amazon Aurora.

Pembaruan mesin basis data Aurora MySQL 2020-03-05 (versi 1.17.9) (Dihentikan)

Versi: 1.17.9

Aurora MySQL 1.17.9 tersedia secara umum. Aurora MySQL versi 1.* kompatibel dengan MySQL 5.6 dan Aurora MySQL versi 2.* kompatibel dengan MySQL 5.7.

Rilis Aurora MySQL yang didukung saat ini adalah 1.14.*, 1.15.*, 1.16.*, 1.17.*, 1.18.*, 1.19.*, 1.20.*, 1.21.*, 1.22.*, 2.01.*, 2.02.*, 2.03.*, 2.04.*, 2.05.*, 2.06.*, dan 2.07.*. Anda dapat memulihkan snapshot basis data Aurora MySQL 1.* ke Aurora MySQL 1.17.9.

Untuk membuat cluster dengan versi lama Aurora MySQL, harap tentukan versi mesin melalui Konsol RDS, AWS CLI, atau Amazon RDS API.

> **Note**
>
> Versi ini saat ini tidak tersedia di wilayah berikut: AWS GovCloud (AS-Timur) [us-gov-east-1], AWS GovCloud (AS-Barat) [us-gov-west-1]. Akan ada pengumuman terpisah setelah tersedia.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

Perbaikan prioritas tinggi:

- Memperbaiki masalah kegagalan koneksi yang tidak stabil setelah rotasi sertifikat.

Pembaruan mesin basis data Aurora MySQL 2019-01-17 (versi 1.17.8) (Dihentikan)

Versi: 1.17.8

Aurora MySQL 1.17.9 tersedia secara umum. Semua klaster basis data Aurora MySQL baru dengan kompatibilitas MySQL 5.6, termasuk yang dipulihkan dari snapshot, akan dibuat dengan Aurora MySQL 1.17.8. Anda memiliki opsi, tetapi tidak diharuskan, untuk meningkatkan klaster basis data yang ada ke Aurora MySQL 1.17.8. Untuk menggunakan versi yang lebih lama, Anda dapat membuat klaster basis data baru di Aurora MySQL 1.14.4, 1.15.1, 1.16, atau 1.17.7. Anda dapat melakukannya menggunakan AWS CLI atau Amazon RDS API dan menentukan versi mesin.

Dengan versi Aurora MySQL 1.17.8, kami menggunakan model patching klaster di mana semua simpul dalam klaster Aurora DB ditambal pada saat yang sama.

> **Note**
>
> Versi ini saat ini tidak tersedia di wilayah AWS GovCloud (AS-Barat) [us-gov-west-1] dan Tiongkok (Beijing) [cn-north-1]. Akan ada pengumuman terpisah setelah tersedia.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

- Memperbaiki masalah kinerja yang meningkatkan penggunaan CPU di Replika Aurora setelah memulai ulang.
- Memperbaiki masalah stabilitas untuk kueri `SELECT` yang menggunakan sambungan hash.

Integrasi perbaikan bug MySQL Community Edition

- BUG #13418638 BUAT TABEL JIKA TIDAK ADA METADATA TERLALU MEMBATASI

Pembaruan mesin basis data Aurora MySQL 2018-10-08 (versi 1.17.7) (Dihentikan)

Versi: 1.17.7

Aurora MySQL 1.17.7 tersedia secara umum. Semua klaster basis data Aurora MySQL baru dengan kompatibilitas MySQL 5.6, termasuk yang dipulihkan dari snapshot, akan dibuat dengan Aurora MySQL 1.17.7. Anda memiliki opsi, tetapi tidak diharuskan, untuk meningkatkan klaster basis data yang ada ke Aurora MySQL 1.17.7. Untuk menggunakan versi yang lebih lama, Anda dapat membuat klaster basis data baru di Aurora MySQL 1.14.4, 1.15.1, 1.16, atau 1.17.6. Anda dapat melakukannya menggunakan AWS CLI atau Amazon RDS API dan menentukan versi mesin.

Dengan versi Aurora MySQL 1.17.7, kami menggunakan model patching klaster di mana semua simpul dalam klaster Aurora DB ditambal pada saat yang sama.

> **Note**
>
> Versi ini saat ini tidak tersedia di wilayah AWS GovCloud (AS-Barat) [us-gov-west-1] dan Tiongkok (Beijing) [cn-north-1]. Akan ada pengumuman terpisah setelah tersedia.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

- Variabel status InnoDB `innodb_buffer_pool_size` telah dibuat terlihat oleh publik untuk diubah oleh pelanggan.
- Memperbaiki masalah stabilitas di klaster Aurora yang terjadi saat terjadi failover.
- Peningkatan ketersediaan klaster dengan memperbaiki masalah pemulihan DDL yang terjadi setelah kegagalan operasi `TRUNCATE`.
- Memperbaiki masalah stabilitas yang terkait dengan pembaruan tabel `mysql.innodb_table_stats`, yang dipicu oleh operasi DDL.
- Memperbaiki masalah stabilitas Replika Aurora yang dipicu selama pembatalan cache kueri setelah operasi DDL.

- Memperbaiki masalah stabilitas yang dipicu oleh akses memori yang tidak valid selama pengosongan cache kamus berkala di latar belakang.

Integrasi perbaikan bug MySQL Community Edition

- Bug #16208542: Meletakkan indeks pada kolom kunci asing menyebabkan hilangnya tabel.
- Bug #76349: kebocoran memori di add_derived_key().
- Bug #16862316: Untuk tabel yang dipartisi, kueri dapat memberikan hasil yang berbeda bergantung pada apakah Penggabungan Indeks digunakan.
- Bug #17588348: Kueri yang menggunakan optimasi index_merge (lihat Optimasi penggabungan indeks) dapat memberikan hasil yang tidak valid saat menjalankan tabel yang dipartisi oleh HASH.

Pembaruan mesin basis data Aurora MySQL 2018-09-06 (versi 1.17.6) (Dihentikan)

Versi: 1.17.6

Aurora MySQL 1.17.6 tersedia secara umum. Semua klaster basis data Aurora MySQL baru dengan kompatibilitas MySQL 5.6, termasuk yang dipulihkan dari snapshot, akan dibuat dengan Aurora MySQL 1.17.6. Anda memiliki opsi, tetapi tidak diharuskan, untuk meningkatkan klaster basis data yang ada ke Aurora MySQL 1.17.6. Untuk menggunakan versi yang lebih lama, Anda dapat membuat klaster basis data baru di Aurora MySQL 1.14.4, 1.15.1, 1.16, atau 1.17.5. Anda dapat melakukannya menggunakan AWS CLI atau Amazon RDS API dan menentukan versi mesin.

Dengan versi Aurora MySQL 1.17.6, kami menggunakan model patching klaster di mana semua simpul dalam klaster Aurora DB ditambal pada saat yang sama.

> **Note**
>
> Versi ini saat ini tidak tersedia di wilayah AWS GovCloud (AS-Barat) [us-gov-west-1] dan Tiongkok (Beijing) [cn-north-1]. Akan ada pengumuman terpisah setelah tersedia.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

- Memperbaiki masalah stabilitas di Pembaca Aurora untuk kueri `SELECT` saat Penulis Aurora melakukan operasi DDL pada tabel yang sama.
- Memperbaiki masalah stabilitas yang disebabkan oleh pembuatan dan penghapusan log DDL untuk tabel sementara yang menggunakan mesin. Heap/Memory
- Memperbaiki masalah stabilitas pada pekerja binlog ketika pernyataan DDL direplikasi ketika koneksi ke Master Binlog tidak stabil.
- Memperbaiki masalah stabilitas yang dihadapi saat menulis ke log kueri yang lambat.
- Memperbaiki masalah dengan tabel status replika yang berisi informasi lag Pembaca Aurora yang salah.

Integrasi perbaikan bug MySQL Community Edition

- Untuk pernyataan ALTER TABLE yang mengubah nama atau mengubah nilai default dari kolom BINARY, perubahan dilakukan menggunakan salinan tabel dan bukan di tempatnya. (Bug #67141, Bug #14735373, Bug #69580, Bug #17024290)
- Sambungan luar antara tabel biasa dan tabel turunan yang secara implisit dikelompokkan dapat menyebabkan server keluar. (Bug #16177639)

Pembaruan mesin basis data Aurora MySQL 2018-08-14 (versi 1.17.5) (Dihentikan)

Versi: 1.17.5

Aurora MySQL 1.17.5 tersedia secara umum. Semua klaster basis data Aurora MySQL baru dengan kompatibilitas MySQL 5.6, termasuk yang dipulihkan dari snapshot, akan dibuat dengan Aurora MySQL 1.17.5. Anda memiliki opsi, tetapi tidak diharuskan, untuk meningkatkan klaster basis data yang ada ke Aurora MySQL 1.17.5. Untuk menggunakan versi yang lebih lama, Anda dapat membuat klaster basis data baru di Aurora MySQL 1.14.4, 1.15.1, 1.16, atau 1.17.4. Anda dapat melakukannya menggunakan AWS CLI atau Amazon RDS API dan menentukan versi mesin.

Dengan versi Aurora MySQL 1.17.5, kami menggunakan model patching klaster di mana semua simpul dalam klaster Aurora DB ditambal pada saat yang sama.

> **Note**
>
> Versi ini saat ini tidak tersedia di wilayah AWS GovCloud (AS-Barat) [us-gov-west-1] dan Tiongkok (Beijing) [cn-north-1]. Akan ada pengumuman terpisah setelah tersedia.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

- Memperbaiki masalah yang menyebabkan Penulis Aurora mungkin mengalami mulai ulang setelah klaster Aurora ditambal menggunakan fitur Zero-Downtime Patching.

Pembaruan mesin basis data Aurora MySQL 2018-08-07 (versi 1.17.4) (Dihentikan)

Versi: 1.17.4

Aurora MySQL 1.17.4 tersedia secara umum. Semua klaster basis data Aurora MySQL baru dengan kompatibilitas MySQL 5.6, termasuk yang dipulihkan dari snapshot, akan dibuat dengan Aurora MySQL 1.17.4. Anda memiliki opsi, tetapi tidak diharuskan, untuk meningkatkan klaster basis data yang ada ke Aurora MySQL 1.17.4. Untuk menggunakan versi yang lebih lama, Anda dapat membuat klaster basis data baru di Aurora MySQL 1.14.4, 1.15.1, 1.16, atau 1.17.3. Anda dapat melakukannya menggunakan AWS CLI atau Amazon RDS API dan menentukan versi mesin.

Dengan versi Aurora MySQL 1.17.4, kami menggunakan model patching klaster di mana semua simpul dalam klaster Aurora DB ditambal pada saat yang sama.

> **Note**
>
> Versi ini saat ini tidak tersedia di wilayah AWS GovCloud (AS-Barat) [us-gov-west-1] dan Tiongkok (Beijing) [cn-north-1]. Akan ada pengumuman terpisah setelah tersedia.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

- Perbaikan replikasi:
  - Mengurangi lalu lintas jaringan dengan tidak memindahkan catatan binlog ke replika klaster. Perbaikan ini diaktifkan secara default.
  - Mengurangi lalu lintas jaringan dengan mengompresi pesan replikasi. Perbaikan ini diaktifkan secara default untuk kelas instans 8xlarge dan 16xlarge. Instans large tersebut dapat mempertahankan volume lalu lintas tulis yang tinggi yang mengakibatkan lalu lintas jaringan yang sangat besar untuk pesan replikasi.
  - Perbaikan pada cache kueri replika.
- Memperbaiki masalah di mana `ORDER BY LOWER(col_name)` dapat menghasilkan pengurutan yang salah saat menggunakan kolasi `utf8_bin`.
- Memperbaiki masalah di mana pernyataan DDL (terutama `TRUNCATE TABLE`) dapat menyebabkan masalah pada replika Aurora, termasuk ketidakstabilan atau hilangnya tabel.
- Memperbaiki masalah di mana soket dibiarkan dalam status setengah terbuka saat simpul penyimpanan dimulai ulang.
- Parameter klaster DB baru berikut ini tersedia:
  - `aurora_enable_zdr` – Memungkinkan koneksi yang dibuka di Replika Aurora tetap aktif saat replika memulai ulang.
  - `aurora_enable_replica_log_compression` – Mengaktifkan kompresi muatan replikasi untuk meningkatkan pemanfaatan bandwidth jaringan antara master dan Replika Aurora.
  - `aurora_enable_repl_bin_log_filtering` – Mengaktifkan pemfilteran catatan replikasi yang tidak dapat digunakan oleh Replika Aurora pada master.

Pembaruan mesin basis data Aurora MySQL 2018-06-05 (versi 1.17.3) (Dihentikan)

Versi: 1.17.3

Aurora MySQL 1.17.3 tersedia secara umum. Semua klaster basis data Aurora MySQL baru dengan kompatibilitas MySQL 5.6, termasuk yang dipulihkan dari snapshot, akan dibuat dengan Aurora MySQL 1.17.3. Anda memiliki opsi, tetapi tidak diharuskan, untuk meningkatkan klaster basis data yang ada ke Aurora MySQL 1.17.3. Anda dapat membuat klaster basis data baru di Aurora MySQL 1.14.4, Aurora MySQL 1.15.1, atau Aurora MySQL 1.16. Anda dapat melakukannya menggunakan AWS CLI atau Amazon RDS API dan menentukan versi mesin.

Dengan versi Aurora MySQL 1.17.3, kami menggunakan model patching klaster di mana semua simpul dalam klaster Aurora DB ditambal pada saat yang sama.

> **Note**
>
> Versi ini saat ini tidak tersedia di wilayah AWS GovCloud (AS-Barat) [us-gov-west-1] dan Tiongkok (Beijing) [cn-north-1]. Akan ada pengumuman terpisah setelah tersedia.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

- Memperbaiki masalah di mana Replika Aurora dapat memulai ulang saat menggunakan pemulihan kursor optimis saat membaca catatan.
- Memperbaiki masalah saat Aurora Writer memulai ulang saat mencoba mematikan sesi MySQL (bunuh ""*<session id>*) dengan skema kinerja diaktifkan.
- Memperbaiki masalah yang menyebabkan Penulis Aurora mengalami mulai ulang saat menghitung ambang batas untuk pengumpulan sampah.
- Memperbaiki masalah yang menyebabkan Penulis Aurora terkadang dapat memulai ulang saat melacak kemajuan Replika Aurora di aplikasi log.
- Memperbaiki masalah pada Kueri Cache saat komitmen otomatis mati dan yang berpotensi menyebabkan pembacaan menjadi usang.

Pembaruan mesin basis data Aurora MySQL 2018-04-27 (versi 1.17.2) (Dihentikan)

Versi: 1.17.2

Aurora MySQL 1.17.2 tersedia secara umum. Semua klaster basis data Aurora MySQL baru dengan kompatibilitas MySQL 5.6, termasuk yang dipulihkan dari snapshot, akan dibuat dengan Aurora MySQL 1.17.2. Anda memiliki opsi, tetapi tidak diharuskan, untuk meningkatkan klaster basis data yang ada ke Aurora MySQL 1.17.2. Anda dapat membuat klaster basis data baru di Aurora MySQL 1.14.4, Aurora MySQL 1.15.1, atau Aurora MySQL 1.16. Anda dapat melakukannya menggunakan AWS CLI atau Amazon RDS API dan menentukan versi mesin.

Dengan versi Aurora MySQL 1.17.2, kami menggunakan model patching klaster di mana semua simpul dalam klaster Aurora DB ditambal pada saat yang sama.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

- Memperbaiki masalah yang menyebabkan mulai ulang selama operasi partisi DDL tertentu.
- Memperbaiki masalah yang menyebabkan dukungan untuk pemanggilan AWS Lambda fungsi melalui fungsi MySQL Aurora asli dinonaktifkan.
- Memperbaiki masalah dengan ketidakabsahan cache yang menyebabkan mulai ulang pada Replika Aurora.
- Memperbaiki masalah di manajer kunci yang menyebabkan mulai ulang.

Pembaruan mesin basis data Aurora MySQL 2018-03-23 (versi 1.17.1) (Dihentikan)

Versi: 1.17.1

Aurora MySQL 1.17.1 tersedia secara umum. Semua klaster basis data baru, termasuk yang dipulihkan dari snapshot, akan dibuat di Aurora MySQL 1.17.1. Anda memiliki opsi, tetapi tidak diharuskan, untuk meningkatkan klaster basis data yang ada ke Aurora MySQL 1.17.1. Anda dapat

membuat klaster DB baru di Aurora MySQL 1.15.1, Aurora MySQL 1.16, atau Aurora MySQL 1.17. Anda dapat melakukannya menggunakan AWS CLI atau Amazon RDS API dan menentukan versi mesin.

Dengan versi Aurora MySQL 1.17.1, kami menggunakan model patching klaster di mana semua simpul dalam klaster Aurora DB ditambal pada saat yang sama. Rilis ini memperbaiki beberapa masalah mesin serta serta regresi yang diketahui.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

> **Note**
>
> Terdapat masalah di versi terbaru mesin Aurora MySQL. Setelah meningkatkan ke 1.17.1, versi mesin dilaporkan secara salah sebagai `1.17`. Jika Anda meningkatkan ke 1.17.1, Anda dapat mengonfirmasi peningkatan dengan memeriksa kolom Pemeliharaan untuk klaster DB di Konsol Manajemen AWS. Jika ditampilkan `none`, maka mesin ditingkatkan ke 1.17.1.

Perbaikan

- Memperbaiki masalah dalam pemulihan log biner yang menghasilkan waktu pemulihan yang lebih lama untuk situasi dengan file indeks log biner besar yang dapat terjadi jika log biner berotasi sangat sering.
- Memperbaiki masalah dalam pengoptimal kueri yang menghasilkan rencana kueri yang tidak efisien untuk tabel yang dipartisi.
- Memperbaiki masalah dalam pengoptimal kueri yang menjadi penyebab kueri rentang menghasilkan mulai ulang pada mesin basis data.

Pembaruan mesin basis data Aurora MySQL 2018-03-13 (versi 1.17) (Dihentikan)

Versi: 1.17

Aurora MySQL 1.17 tersedia secara umum. Aurora MySQL versi 1.x hanya kompatibel dengan MySQL 5.6, dan bukan MySQL 5.7. Semua klaster basis data baru yang kompatibel dengan 5.6,

termasuk yang dipulihkan dari snapshot, akan dibuat dengan Aurora 1.17. Anda memiliki opsi, tetapi tidak diharuskan, untuk meningkatkan klaster basis data yang ada ke Aurora 1.17. Anda dapat membuat klaster DB baru di Aurora 1.14.1, Aurora 1.15.1, atau Aurora 1.16. Anda dapat melakukannya menggunakan AWS CLI atau Amazon RDS API dan menentukan versi mesin.

Dengan versi Aurora 1.17, kami menggunakan model patching klaster di mana semua simpul dalam klaster Aurora DB ditambal pada saat yang sama. Kami mendukung zero-downtime patching, yang bekerja berdasarkan upaya terbaik untuk mempertahankan koneksi klien melalui proses patching. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support.

Zero-downtime patching

Fitur zero-downtime patching (ZDP) mencoba, berdasarkan upaya terbaik, untuk menjaga koneksi klien melalui patch mesin. Untuk informasi selengkapnya tentang ZDP, lihat Menggunakan zero-downtime patching di Panduan Pengguna Amazon Aurora.

Fitur baru

- Aurora MySQL sekarang mendukung kompresi kunci, yang mengoptimalkan penggunaan memori manajer kunci. Mulai versi 1.17, Anda dapat menggunakan fitur ini tanpa mengaktifkan mode lab.

Perbaikan

- Memperbaiki masalah yang terutama terlihat pada instans dengan lebih sedikit inti di mana satu inti mungkin memiliki 100% penggunaan CPU bahkan ketika basis data sedang idle.
- Meningkatkan kinerja pengambilan log biner dari klaster Aurora.
- Memperbaiki masalah yang menyebabkan Replika Aurora berusaha untuk menuliskan statistik tabel ke penyimpanan persisten, dan mengalami crash.
- Memperbaiki masalah yang menyebabkan cache pencarian tidak berfungsi seperti yang diharapkan di Replika Aurora.
- Memperbaiki kondisi balapan di manajer kunci yang mengakibatkan mulai ulang mesin.
- Memperbaiki masalah di mana kunci yang diambil oleh transaksi hanya-baca dan commit-otomatis menyebabkan mulai ulang mesin.

- Memperbaiki masalah yang menyebabkan beberapa kueri tidak dituliskan ke log audit.
- Memperbaiki masalah pada pemulihan dari operasi pemeliharaan partisi tertentu pada failover.

Integrasi perbaikan bug MySQL

- LAST_INSERT_ID direplikasi dengan tidak benar jika filter replikasi digunakan (Bug #69861)
- Pencarian memberikan hasil yang berbeda tergantung pada pengaturan INDEX_MERGE (Bug #16862316)
- Proc kueri mengeksekusi ulang penyimpanan rutin, rencana kueri tidak efisien (Bug #16346367)
- INNODB FTS : Pernyataan dalam FTS_CACHE_APPEND_DELETED_DOC_IDS (BUG #18079671)
- Pernyataan RBT_EMPTY(INDEX_CACHE->WORDS) di ALTER TABLE CHANGE COLUMN (BUG #17536995)
- Pencarian teks penuh INNODB tidak menemukan catatan ketika titik penyimpanan dilibatkan (BUG #70333, BUG #17458835)

Pembaruan mesin basis data Aurora MySQL 2017-12-11 (versi 1.16) (Dihentikan)

Versi: 1.16

Aurora MySQL 1.16 tersedia secara umum. Semua klaster basis data baru, termasuk yang dipulihkan dari snapshot, akan dibuat dengan Aurora 1.16. Anda memiliki opsi, tetapi tidak diharuskan, untuk meningkatkan klaster basis data yang ada ke Aurora 1.16. Anda dapat membuat klaster DB baru di Aurora 1.14.1 atau Aurora 1.15.1. Anda dapat melakukannya menggunakan AWS CLI atau Amazon RDS API dan menentukan versi mesin.

Dengan versi Aurora 1.16, kami menggunakan model patching klaster di mana semua simpul dalam klaster Aurora DB ditambal pada saat yang sama. Kami mengaktifkan zero-downtime patching, yang bekerja berdasarkan upaya terbaik untuk mempertahankan koneksi klien melalui proses patching. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support.

Zero-downtime patching

Fitur zero-downtime patching (ZDP) mencoba, berdasarkan upaya terbaik, untuk menjaga koneksi klien melalui patch mesin. Untuk informasi selengkapnya tentang ZDP, lihat Menggunakan zero-downtime patching di Panduan Pengguna Amazon Aurora.

Fitur baru

- Aurora MySQL sekarang mendukung pemanggilan sinkron AWS Lambda melalui fungsi asli. `lambda_sync()` Fungsi asli `lambda_async()` juga tersedia, yang dapat digunakan sebagai alternatif untuk prosedur tersimpan yang ada untuk invokasi Lambda asinkron. Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dari klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.
- Aurora MySQL sekarang mendukung sambungan hash untuk mempercepat kueri equijoin. Pengoptimal berbasis biaya Aurora dapat secara otomatis memutuskan kapan akan menggunakan sambungan hash; Anda juga dapat memaksakan penggunaannya dalam rencana kueri. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- Aurora MySQL sekarang mendukung pemindaian batch untuk mempercepat kueri berorientasi pemindaian dalam memori secara signifikan. Fitur ini meningkatkan kinerja pemindaian penuh tabel, pemindaian penuh indeks, dan pemindaian rentang indeks berdasarkan pemrosesan batch.

Perbaikan

- Memperbaiki masalah yang menyebabkan replika baca terhenti saat menjalankan kueri di tabel yang baru saja dihapus di master.
- Memperbaiki masalah di mana memulai ulang penulis di klaster basis data dengan jumlah indeks `FULLTEXT` yang sangat besar menghasilkan pemulihan yang lebih lama dari yang diharapkan.
- Memperbaiki masalah di mana menghapus log biner menyebabkan insiden `LOST_EVENTS` di peristiwa binlog.
- Memperbaiki masalah stabilitas dengan penjadwal saat skema kinerja diaktifkan.
- Memperbaiki masalah yang menyebabkan subkueri yang menggunakan tabel sementara dapat memberikan hasil parsial.

Integrasi perbaikan bug MySQL

Tidak ada

Pembaruan mesin basis data Aurora MySQL 2017-11-20 (versi 1.15.1) (Dihentikan)

Versi: 1.15.1

Aurora MySQL 1.15.1 tersedia secara umum. Semua klaster basis data baru, termasuk yang dipulihkan dari snapshot, akan dibuat dengan Aurora 1.15.1. Anda memiliki opsi, tetapi tidak diharuskan, untuk meningkatkan klaster DB yang ada ke Aurora MySQL 1.15.1. Anda dapat membuat kluster DB baru di Aurora 1.14.1. Anda dapat melakukannya menggunakan AWS CLI atau Amazon RDS API dan menentukan versi mesin.

Dengan versi Aurora 1.15.1, kami menggunakan model patching klaster di mana semua simpul dalam klaster Aurora DB ditambal pada saat yang sama. Kami mengaktifkan zero-downtime patching, yang bekerja berdasarkan upaya terbaik untuk mempertahankan koneksi klien melalui proses patching. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Zero-downtime patching

Fitur zero-downtime patching (ZDP) mencoba, berdasarkan upaya terbaik, untuk menjaga koneksi klien melalui patch mesin. Untuk informasi selengkapnya tentang ZDP, lihat Menggunakan zero-downtime patching di Panduan Pengguna Amazon Aurora.

Perbaikan

- Memperbaiki masalah di selektor segmen adaptif untuk permintaan baca yang akan menyebabkannya memilih segmen yang sama dua kali yang menyebabkan lonjakan latensi baca di bawah keadaan tertentu.
- Memperbaiki masalah yang berasal dari pengoptimalan di Aurora MySQL untuk penjadwal thread. Masalah ini terlihat dari kesalahan yang mengkhawatirkan saat menulis ke log lambat, sementara kueri yang terkait dapat bekerja dengan baik.

- Memperbaiki masalah stabilitas replika baca pada volume besar (> 5 TB).
- Memperbaiki masalah yang menyebabkan jumlah thread pekerja meningkat terus-menerus karena jumlah koneksi yang luar biasa.
- Memperbaiki masalah dengan kunci tabel yang menyebabkan waktu tunggu semaphore lama selama beban kerja penyisipan.
- Mengembalikan perbaikan bug MySQL berikut yang disertakan di Aurora MySQL 1.15:
  - Instans MySQL yang macet saat "melakukan indeks SYNC" (Bug #73816)
  - Pernyataan RBT_EMPTY(INDEX_CACHE->WORDS) di ALTER TABLE CHANGE COLUMN (Bug #17536995)
  - Pencarian Teks Penuh InnoDB tidak menemukan catatan ketika titik penyimpanan dilibatkan (Bug #70333)

Integrasi perbaikan bug MySQL

Tidak ada

Pembaruan mesin basis data Aurora MySQL 2017-10-24 (versi 1.15) (Dihentikan)

Versi: 1.15

Aurora MySQL 1.15 tersedia secara umum. Semua klaster basis data baru, termasuk yang dipulihkan dari snapshot, akan dibuat dengan Aurora 1.15. Anda memiliki opsi, tetapi tidak diharuskan, untuk meningkatkan klaster DB yang ada ke Aurora MySQL 1.15. Anda dapat membuat klaster DB baru di Aurora 1.14.1. Anda dapat melakukannya menggunakan AWS CLI atau Amazon RDS API dan menentukan versi mesin.

Dengan versi Aurora 1.15, kami menggunakan model patching klaster di mana semua simpul dalam klaster Aurora DB ditambal pada saat yang sama. Pembaruan memerlukan mulai ulang basis data, sehingga Anda akan mengalami waktu henti selama 20 hingga 30 detik, setelah itu Anda dapat melanjutkan menggunakan klaster DB Anda. Jika klaster DB Anda saat ini menjalankan Aurora 1.14 atau Aurora 1.14.1, fitur zero-downtime patching di Aurora MySQL mungkin memungkinkan koneksi klien ke instans primer Aurora MySQL Anda tetap ada selama peningkatan, tergantung beban kerja Anda.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Zero-downtime patching

Fitur zero-downtime patching (ZDP) mencoba, berdasarkan upaya terbaik, untuk menjaga koneksi klien melalui patch mesin. Untuk informasi selengkapnya tentang ZDP, lihat Menggunakan zero-downtime patching di Panduan Pengguna Amazon Aurora.

Fitur baru

- Asynchronous Key Prefetch – Asynchronous key prefetch (AKP) adalah fitur yang ditargetkan untuk meningkatkan kinerja sambungan indeks yang tidak di-cache, dengan mengambil kunci di dalam memori terlebih dahulu sebelum kunci tersebut dibutuhkan. Kasus penggunaan utama yang ditargetkan oleh AKP adalah sambungan indeks antara tabel luar kecil dan tabel dalam besar, di mana indeks sangat selektif pada tabel yang lebih besar. Selain itu, ketika antarmuka Multi-Range Read (MRR) diaktifkan, AKP akan dimanfaatkan untuk pencarian indeks sekunder ke indeks primer. Instans yang lebih kecil yang memiliki keterbatasan memori mungkin dalam beberapa kasus dapat memanfaatkan AKP, dengan mempertimbangkan kardinalitas kunci yang benar. Untuk informasi selengkapnya, lihat Mengoptimalkan kueri sambungan terindeks Aurora dengan asynchronous key prefetch di Panduan Pengguna Amazon Aurora.
- DDL Cepat – Kami telah memperluas fitur yang dirilis di Aurora 1.13 ke operasi yang mencakup nilai standar. Dengan perluasan fitur ini, DDL Cepat berlaku untuk operasi yang menambahkan kolom kosong di akhir tabel, dengan atau tanpa nilai default. Fitur tetap dalam mode lab Aurora. Lihat informasi yang lebih lengkap di Mengubah tabel di Amazon Aurora menggunakan DDL cepat di Panduan Pengguna Amazon Aurora.

Perbaikan

- Memperbaiki kesalahan perhitungan selama optimalisasi kueri WITHIN/CONTAINS spasial yang sebelumnya menghasilkan set hasil kosong.
- Memperbaiki perintah `SHOW VARIABLE` untuk menampilkan nilai parameter `innodb_buffer_pool_size` yang diperbarui setiap kali nilai diubah dalam grup parameter.
- Meningkatkan stabilitas instans primer selama penyisipan massal ke dalam tabel yang diubah menggunakan DDL Cepat saat pengindeksan hash adaptif dinonaktifkan dan catatan yang disisipkan adalah catatan pertama halaman.

- Meningkatkan stabilitas Aurora saat pengguna mencoba menetapkan nilai parameter klaster DB server_audit_events ke **`default`**.
- Memperbaiki masalah yang menyebabkan perubahan set karakter basis data untuk pernyataan TABLE ALTER yang dijalankan di instans primer Aurora tidak direplikasi di Replika Aurora sampai dimulai ulang.
- Meningkatkan stabilitas dengan memperbaiki kondisi balapan pada instans primer yang sebelumnya mengizinkannya untuk mendaftarkan Replika Aurora bahkan jika instans primernya telah menutup volumenya sendiri.
- Meningkatkan kinerja instans primer selama pembuatan indeks di tabel besar dengan mengubah protokol penguncian untuk memungkinkan pernyataan bahasa manipulasi data (DML) serentak pembuatan indeks.
- Memperbaiki inkonsistensi metadata InnoDB selama kueri ALTER TABLE RENAME yang meningkatkan stabilitas. Contoh: Ketika kolom tabel t1(c1, c2) diubah namanya secara siklis menjadi t1(c2,c3) dalam pernyataan ALTER yang sama.
- Meningkatkan stabilitas Replika Aurora untuk skenario di mana Replika Aurora tidak memiliki beban kerja aktif dan instans primernya tidak responsif.
- Meningkatkan ketersediaan Replika Aurora untuk skenario di mana Replika Aurora memegang kunci eksplisit pada tabel dan memblokir thread replikasi dari menerapkan perubahan DDL apa pun yang diterima dari instans primer.
- Meningkatkan stabilitas instans primer saat kunci asing dan kolom ditambahkan ke tabel dari dua sesi terpisah pada saat yang sama dan DDL Cepat telah diaktifkan.
- Meningkatkan stabilitas thread pembersihan di instans primer selama beban kerja tulis yang berat dengan memblokir rekaman undo hingga selesai.
- Meningkatkan stabilitas dengan memperbaiki urutan pelepasan kunci selama proses transaksi yang menghasilkan tabel.
- Memperbaiki cacat untuk Replika Aurora di mana instans DB tidak dapat menyelesaikan startup dan memberi tahu bahwa port 3306 sudah digunakan.
- Memperbaiki kondisi race di mana kueri SELECT berjalan pada tabel information_schema tertentu (innodb_trx, innodb_lock, innodb_lock_waits) meningkatkan ketidakstabilan klaster.

Integrasi perbaikan bug MySQL

- CREATE USER menerima plugin dan perintah kata sandi, tetapi mengabaikan hash kata sandi (Bug #78033)

- Mesin partisi menambahkan bidang pada set bit baca agar dapat mengembalikan entri yang disortir dari indeks yang dipartisi. Hal ini menyebabkan buffer sambungan akan mencoba membaca bidang yang tidak diperlukan. Diperbaiki dengan tidak menambahkan semua kolom ke read_set, tetapi hanya menyortir kolom prefiks yang sudah ditetapkan di read_set. Menambahkan DBUG_ASSERT bahwa jika melakukan key_cmp, setidaknya kolom pertama harus dibaca (Bug #16367691).
- Instans MySQL yang macet saat "melakukan indeks SYNC" (Bug #73816)
- Pernyataan RBT_EMPTY(INDEX_CACHE->WORDS) di ALTER TABLE CHANGE COLUMN (Bug #17536995)
- Pencarian Teks Penuh InnoDB tidak menemukan catatan ketika titik penyimpanan dilibatkan (Bug #70333)

Pembaruan mesin basis data Aurora MySQL 2018-03-13 (versi 1.14.4) (Dihentikan)

Versi: 1.14.4

Aurora MySQL 1.14.4 tersedia secara umum. Anda dapat membuat cluster DB baru di Aurora 1.14.4, menggunakan AWS CLI atau Amazon RDS API dan menentukan versi mesin. Anda memiliki opsi, tetapi tidak diharuskan, untuk meningkatkan klaster DB yang ada ke Aurora MySQL 1.14.4.

Dengan versi Aurora 1.14.4, kami menggunakan model patching klaster di mana semua simpul dalam klaster Aurora DB ditambal pada saat yang sama. Kami mendukung zero-downtime patching, yang bekerja berdasarkan upaya terbaik untuk mempertahankan koneksi klien melalui proses patching. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Zero-downtime patching

Fitur zero-downtime patching (ZDP) mencoba, berdasarkan upaya terbaik, untuk menjaga koneksi klien melalui patch mesin. Untuk informasi selengkapnya tentang ZDP, lihat Menggunakan zero-downtime patching di Panduan Pengguna Amazon Aurora.

Fitur baru

- Aurora MySQL sekarang mendukung kelas instans db.r4.

Perbaikan

- Memperbaiki masalah di mana `LOST_EVENTS` dihasilkan saat menulis peristiwa binlog besar.

Integrasi perbaikan bug MySQL

- Peristiwa yang tidak berbahaya tidak berhasil dan tidak diuji (Bug #74683)
- NEW->OLD ASSERT FAILURE 'GTID_MODE > 0' (Bug #20436436)

Pembaruan mesin basis data Aurora MySQL: 2017-09-22 (versi 1.14.1) (Dihentikan)

Versi: 1.14.1

Aurora MySQL 1.14.1 tersedia secara umum. Semua klaster basis data baru, termasuk yang dipulihkan dari snapshot, akan dibuat di Aurora MySQL 1.14.1. Aurora MySQL 1.14.1 juga merupakan peningkatan wajib untuk klaster DB Aurora MySQL yang ada. Untuk informasi selengkapnya, lihat Pengumuman: Perpanjangan ke jadwal pemutakhiran wajib untuk Amazon Aurora di situs web Forum AWS Pengembang.

Dengan versi Aurora MySQL 1.14.1, kami menggunakan model patching klaster di mana semua simpul dalam klaster Aurora DB ditambal pada saat yang sama. Pembaruan memerlukan mulai ulang basis data, sehingga Anda akan mengalami waktu henti selama 20 hingga 30 detik, setelah itu Anda dapat melanjutkan menggunakan klaster DB Anda. Jika klaster DB Anda saat ini menjalankan versi 1.13 atau lebih tinggi, fitur patching zero-downtime di Aurora MySQL mungkin memungkinkan koneksi klien ke instans primer Aurora MySQL Anda tetap ada selama peningkatan, bergantung pada beban kerja Anda.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support.

Perbaikan

- Memperbaiki kondisi balapan yang terkait dengan penyisipan dan pembersihan untuk meningkatkan stabilitas fitur DDL Cepat, yang tetap berada dalam mode lab Aurora MySQL.

Pembaruan mesin basis data Aurora MySQL 2017-08-07 (versi 1.14) (Dihentikan)

Versi: 1.14

Aurora MySQL 1.14 tersedia secara umum. Semua klaster basis data baru, termasuk yang dipulihkan dari snapshot, akan dibuat di Aurora MySQL 1.14. Aurora MySQL 1.14 juga merupakan peningkatan wajib untuk klaster DB Aurora MySQL yang ada. Kami akan mengirimkan pengumuman terpisah dengan jadwal untuk menghentikan versi Aurora MySQL sebelumnya.

Dengan versi Aurora MySQL 1.14, kami menggunakan model patching klaster di mana semua simpul dalam klaster Aurora DB ditambal pada saat yang sama. Pembaruan memerlukan mulai ulang basis data, sehingga Anda akan mengalami waktu henti selama 20 hingga 30 detik, setelah itu Anda dapat melanjutkan menggunakan klaster DB Anda. Jika klaster DB Anda saat ini menjalankan fitur patching zerotime versi 1.13, fitur Aurora mungkin memungkinkan koneksi klien ke instans primer Aurora Anda bertahan melalui peningkatan, tergantung pada beban kerja Anda.

Jika Anda memiliki pertanyaan atau masalah, AWS Support tersedia di forum komunitas dan melalui AWS Support.

Zero-downtime patching

Fitur zero-downtime patching (ZDP) mencoba, berdasarkan upaya terbaik, untuk menjaga koneksi klien melalui patch mesin. Untuk informasi selengkapnya tentang ZDP, lihat Menggunakan zero-downtime patching di Panduan Pengguna Amazon Aurora.

Perbaikan

- Memperbaiki kesalahan “catatan tidak ditemukan” yang salah ketika catatan ditemukan dalam indeks sekunder tetapi tidak dalam indeks primer.
- Memperbaiki masalah stabilitas yang dapat terjadi karena pernyataan defensif (ditambahkan di 1.12) yang terlalu kuat dalam kasus ketika laporan individu terdiri dari 32 halaman. Situasi seperti itu dapat terjadi, misalnya, dengan nilai BLOB Large.

- Memperbaiki masalah stabilitas karena ketidakkonsistenan antara cache tablespace dan cache kamus.
- Memperbaiki masalah yang menyebabkan Replika Aurora menjadi tidak responsif setelah basis data melebihi jumlah maksimum upaya untuk terhubung dengan instans primer. Replika Aurora sekarang akan kembali lagi jika periode tidak aktif lebih dari periode waktu detak jantung yang digunakan untuk pemeriksaan kondisi oleh instans primer.
- Memperbaiki livelock yang dapat terjadi dengan konkurensi yang sangat tinggi ketika satu koneksi mencoba mendapatkan metadata lock (MDL) eksklusif saat mengeluarkan perintah, seperti `ALTER TABLE`.
- Memperbaiki masalah stabilitas di Replika Baca Aurora dengan adanya logical/parallel read ahead.
- Memperbaiki `LOAD FROM S3` dengan dua cara:
  1. Penanganan yang lebih baik dari kesalahan batas waktu Amazon S3 dengan menggunakan coba ulang SDK selain coba ulang yang sudah ada.
  2. Optimalisasi kinerja saat memuat file yang sangat besar atau file dalam jumlah besar dengan menyimpan dalam cache dan menggunakan kembali status klien.
- Memperbaiki masalah stabilitas berikut dengan DDL Cepat untuk operasi `ALTER TABLE`:
  1. Saat pernyataan `ALTER TABLE` memiliki beberapa perintah `ADD COLUMN` dan nama kolom tidak dalam urutan naik.
  2. Ketika string nama kolom akan diperbarui dan string nama yang sesuai, yang diambil dari tabel sistem internal, berbeda dengan karakter yang berujung nol (/0).
  3. Dalam operasi B-tree split tertentu.
  4. Saat tabel memiliki kunci primer panjang variabel.
- Memperbaiki masalah stabilitas dengan Replika Aurora ketika diperlukan waktu terlalu lama agar Full Text Search (FTS) index cache konsisten dengan cache dari instans primer. Hal ini dapat terjadi jika sebagian besar entri indeks FTS yang baru dibuat pada instans primer belum di-flush ke disk.
- Memperbaiki masalah stabilitas yang dapat terjadi selama pembuatan indeks.
- Infrastruktur baru yang melacak konsumsi memori per koneksi dan telemetri terkait yang akan digunakan untuk membangun strategi penghindaran Out-Of-Memory (OOM).
- Memperbaiki masalah di mana `ANALYZE TABLE` diizinkan secara salah pada Replika Aurora. Hal ini sekarang telah diblokir.
- Memperbaiki masalah stabilitas yang disebabkan oleh deadlock yang jarang terjadi sebagai akibat dari kondisi balapan antara pembersihan dan read-ahead logis.

Integrasi perbaikan bug MySQL

- Pencarian teks penuh yang digabungkan dengan tabel turunan (subkueri dalam klausa `FROM`) menyebabkan server keluar. Sekarang, jika operasi teks penuh tergantung pada tabel turunan, server akan menghasilkan kesalahan yang menunjukkan bahwa pencarian teks penuh tidak dapat dilakukan pada tabel yang termaterialisasi. (Bug #68751, Bug #16539903)

Pembaruan mesin basis data Aurora MySQL: 2017-05-15 (versi 1.13) (Dihentikan)

Versi: 1.13

> **Note**
>
> Kami mengaktifkan fitur baru - SELECT INTO OUTFILE S3 - di Aurora MySQL versi 1.13 setelah rilis awal, dan telah memperbarui catatan rilis untuk mencerminkan perubahan tersebut.

Aurora MySQL 1.13 tersedia secara umum. Semua klaster basis data baru, termasuk yang dipulihkan dari snapshot, akan dibuat di Aurora MySQL 1.13. Anda memiliki opsi, tetapi tidak diharuskan, untuk meningkatkan klaster basis data yang ada ke Aurora MySQL 1.13. Dengan versi Aurora 1.13, kami menggunakan model patching klaster di mana semua simpul dalam klaster Aurora DB ditambal pada saat yang sama. Kami mengaktifkan zero-downtime patching, yang bekerja berdasarkan upaya terbaik untuk mempertahankan koneksi klien melalui proses patching. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Zero-downtime patching

Fitur zero-downtime patching (ZDP) mencoba, berdasarkan upaya terbaik, untuk menjaga koneksi klien melalui patch mesin. Untuk informasi selengkapnya tentang ZDP, lihat Menggunakan zero-downtime patching di Panduan Pengguna Amazon Aurora.

Fitur baru

- SELECT INTO OUTFILE S3 – Aurora MySQL sekarang memungkinkan Anda mengunggah hasil kueri ke satu file atau lebih dalam bucket Amazon S3. Untuk informasi selengkapnya, lihat

Menyimpan data dari klaster DB Amazon Aurora MySQL ke dalam file teks di bucket Amazon S3 di Panduan Pengguna Amazon Aurora.

Perbaikan:

- Implementasi pemenggalan file log format CSV pada saat startup mesin untuk menghindari waktu pemulihan yang lama. Tabel `general_log_backup`, `general_log`, `slow_log_backup`, dan `slow_log` sekarang tidak bertahan setelah basis data dimulai ulang.
- Memperbaiki masalah yang menyebabkan kegagalan migrasi basis data bernama **`test`**.
- Meningkatkan stabilitas di pengumpul sampah manajer kunci dengan menggunakan kembali segmen kunci yang benar.
- Meningkatkan stablitas manajer kunci dengan menghilangkan pernyataan yang tidak valid selama algoritma deteksi deadlock.
- Mengaktifkan ulang replikasi asinkron, dan memperbaiki masalah terkait yang menyebabkan lag replika yang tidak benar yang dilaporkan di bawah beban kerja tanpa muatan atau hanya-baca. Peningkatan alur replikasi yang diperkenalkan di versi 1.10. Perbaikan ini diperkenalkan untuk menerapkan pembaruan log stream pada cache buffer Replika Aurora, yang membantu meningkatkan kinerja baca dan stabilitas di Replika Aurora.
- Memperbaiki masalah di mana autocommit=OFF menyebabkan pemblokiran peristiwa terjadwal dan transaksi yang lama dibiarkan terbuka hingga server melakukan boot ulang.
- Memperbaiki masalah di mana log kueri umum, audit, dan lambat tidak dapat mencatat log kueri yang ditangani oleh penerapan asinkron.
- Meningkatkan kinerja fitur logical read ahead (LRA) hingga 2,5 kali. Hal ini dilakukan dengan memungkinkan pengambilan awal untuk melanjutkan di seluruh halaman menengah di B-tree.
- Menambahkan validasi parameter untuk variabel audit untuk memangkas ruang yang tidak perlu.
- Memperbaiki regresi, yang diperkenalkan di Aurora MySQL versi 1.11, di mana kueri dapat mengembalikan hasil yang salah saat menggunakan opsi SQL_CALC_FOUND_ROWS dan menginvokasi fungsi FOUND_ROWS().
- Memperbaiki masalah stabilitas saat daftar Kunci Metadata salah dibentuk.
- Meningkatkan stabilitas ketika sql_mode diatur ke PAD_CHAR_TO_FULL_LENGTH dan perintah `SHOW FUNCTION STATUS WHERE Db='`*`string`*`'` dijalankan.
- Memperbaiki kasus yang jarang terjadi ketika instans tidak muncul setelah peningkatan versi Aurora karena pemeriksaan konsistensi volume yang salah.

- Memperbaiki masalah kinerja, yang diperkenalkan di Aurora MySQL versi 1.12, di mana kinerja penulis Aurora berkurang ketika pengguna memiliki banyak tabel.
- Memperbaiki masalah stabilitas saat penulis Aurora dikonfigurasi sebagai pekerja binlog dan jumlah koneksi mendekati 16.000.
- Memperbaiki masalah langka yang menyebabkan Replika Aurora dapat memulai ulang ketika koneksi diblokir dan menunggu untuk Kunci Metadata ketika menjalankan DDL pada master Aurora.

Integrasi perbaikan bug MySQL

- Dengan tabel InnoDB kosong, tidak mungkin menurunkan nilai auto_increment menggunakan pernyataan ALTER TABLE, meskipun tabel kosong. (Bug #69882)
- MATCH() ... Kueri AGAINST yang menggunakan string panjang sebagai argumen untuk AGAINST() dapat mengakibatkan kesalahan saat dijalankan pada tabel InnoDB dengan indeks pencarian teks penuh. (Bug #17640261)
- Penanganan SQL_CALC_FOUND_ROWS yang dikombinasikan dengan ORDER BY dan LIMIT dapat menyebabkan hasil yang salah untuk FOUND_ROWS(). (Bug #68458, Bug # 16383173)
- ALTER TABLE tidak memungkinkan perubahan null kolom jika ada kunci asing. (Bug #77591)

Pembaruan mesin basis data Aurora MySQL: 2017-04-05 (versi 1.12) (Dihentikan)

Versi: 1.12

Aurora MySQL 1.12 sekarang menjadi versi pilihan untuk pembuatan klaster DB baru, termasuk pemulihan dari snapshot.

Ini bukan peningkatan wajib untuk klaster yang ada. Anda akan memiliki opsi untuk meningkatkan kluster yang ada ke versi 1.12 setelah kami menyelesaikan patch di seluruh armada menjadi 1.11 (lihat catatan rilis Aurora 1.11 dan pengumuman forum terkait). Dengan versi Aurora 1.12, kami menggunakan model patching klaster di mana semua simpul dalam klaster Aurora DB ditambal pada saat yang sama. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Fitur baru

- DDL Cepat – Aurora MySQL sekarang memungkinkan Anda untuk mengeksekusi operasi ALTER TABLE tbl_name ADD COLUMN col_name column_definition hampir secara seketika. Operasi tersebut akan diselesaikan tanpa perlu menyalin tabel dan tanpa memengaruhi pernyataan DML lainnya secara signifikan. Karena tidak menggunakan penyimpanan sementara untuk salinan tabel, pernyataan DDL menjadi praktis bahkan untuk tabel besar di kelas instans kecil. DDL Cepat saat ini hanya mendukung penambahan kolom yang dapat dikosongkan, tanpa nilai default, di akhir tabel. Fitur ini saat ini tersedia dalam mode lab Aurora. Lihat informasi yang lebih lengkap di Mengubah tabel di Amazon Aurora menggunakan DDL cepat di Panduan Pengguna Amazon Aurora.
- Menunjukkan status volume - Kami telah menambahkan perintah pemantauan baru, SHOW VOLUME STATUS, untuk menampilkan jumlah simpul dan disk dalam satu volume. Lihat informasi yang lebih lengkap di Menampilkan status volume untuk klaster DB Aurora MySQL dalam Panduan Pengguna Amazon Aurora.

Perbaikan

- Menerapkan perubahan kompresi kunci untuk lebih mengurangi alokasi memori per objek kunci. Perbaikan ini tersedia dalam mode lab.
- Memperbaiki masalah di mana metrik `trx_active_transactions` menurun dengan cepat meskipun basis data tidak digunakan.
- Memperbaiki pesan kesalahan yang tidak valid terkait sintaks kueri injeksi kesalahan saat menyimulasikan kegagalan dalam disk dan simpul.
- Memperbaiki beberapa masalah terkait dengan kondisi race dan dead latch di manajer kunci.
- Memperbaiki masalah yang menyebabkan luapan buffer di optimalisasi kueri.
- Memperbaiki masalah stabilitas pada replika baca Aurora ketika simpul penyimpanan yang mendasari memiliki memori rendah yang tersedia.
- Memperbaiki masalah di mana koneksi idle bertahan melewati pengaturan parameter `wait_timeout`.
- Memperbaiki masalah yang `query_cache_size` mengembalikan nilai tak terduga setelah me-boot ulang instans.
- Memperbaiki masalah kinerja yang merupakan hasil dari thread diagnostik yang terlalu sering memeriksa jaringan apabila penulisan tidak mencapai penyimpanan.

Integrasi perbaikan bug MySQL

- Memuat ulang tabel yang dikosongkan dalam keadaan kosong menyebabkan nilai AUTO_INCREMENT direset. (Bug #21454472, Bug #77743)
- Catatan indeks tidak ditemukan pada rollback karena inkonsistensi di dalam struktur purge_node_t. Inkonsistensi menimbulkan peringatan dan pesan kesalahan seperti “error in sec index entry update”, “unable to purge a record”, dan “tried to purge sec index entry not marked for deletion”. (Bug #19138298, Bug #70214, Bug #21126772, Bug #21065746)
- Perhitungan ukuran tumpukan yang salah untuk operasi qsort menyebabkan luapan tumpukan. (Bug #73979)
- Catatan tidak ditemukan dalam indeks pada saat rollback. (Bug #70214, Bug #72419)
- ALTER TABLE tambahkan kolom TIMESTAMP saat memperbarui CURRENT_TIMESTAMP menyisipkan ZERO-datas (Bug #17392)

Pembaruan mesin basis data Aurora MySQL: 2017-02-23 (versi 1.11) (Dihentikan)

Versi: 1.11

Kami akan menambal semua klaster DB Aurora MySQL dengan versi terbaru dalam waktu singkat setelah rilis. Klaster DB di-patch menggunakan prosedur legacy dengan waktu henti sekitar 5-30 detik.

Patching terjadi selama jendela pemeliharaan sistem yang telah Anda tentukan untuk setiap instans basis data Anda. Anda dapat melihat atau mengubah jadwal ini menggunakan Konsol Manajemen AWS. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Atau, Anda dapat menerapkan patch segera di Konsol Manajemen AWS dengan memilih cluster DB, memilih Tindakan Cluster, dan kemudian memilih Upgrade Now.

Dengan versi Aurora MySQL 1.11, kami menggunakan model patching klaster di mana semua simpul dalam klaster Aurora DB ditambal pada saat yang sama.

Fitur baru

- Opsi MANIFEST untuk LOAD DATA FROM S3 – LOAD DATA FROM S3 dirilis di versi 1.8. Opsi untuk perintah ini telah diperluas, dan Anda sekarang dapat menentukan daftar file yang akan dimuat ke klaster Aurora DB dari Amazon S3 dengan menggunakan file manifes. Ini memudahkan untuk memuat data dari file tertentu di satu atau lebih lokasi, alih-alih memuat data dari satu file dengan menggunakan opsi FILE atau memuat data dari beberapa file yang memiliki lokasi dan prefiks yang sama dengan menggunakan opsi PREFIX. Format file manifes sama dengan yang digunakan oleh Amazon Redshift. Untuk informasi selengkapnya tentang penggunaan LOAD DATA FROM S3 dengan opsi MANIFEST, lihat Menggunakan manifes untuk menentukan file data yang akan dimuat di Panduan Pengguna Amazon Aurora.
- Pengindeksan spasial diaktifkan secara default – Fitur ini dirilis dalam mode lab di versi 1.10, dan sekarang diaktifkan secara default. Pengindeksan spasial meningkatkan kinerja kueri pada set data besar untuk kueri yang menggunakan data spasial. Untuk informasi selengkapnya tentang penggunaan pengindeksan spasial, lihat Amazon Aurora MySQL dan data spasial di Panduan Pengguna Amazon Aurora.
- Perubahan pengaturan waktu Audit Lanjutan – Fitur ini dirilis pada versi 1.10.1 untuk menyediakan fasilitas berkinerja tinggi untuk audit aktivitas basis data. Dalam rilis ini, presisi stempel waktu log audit telah diubah dari satu detik menjadi satu mikrodetik. Stempel waktu yang lebih akurat memungkinkan Anda untuk lebih memahami kapan peristiwa audit terjadi. Untuk informasi selengkapnya tentang audit, lihat Menggunakan Audit Lanjutan dengan klaster DB Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

- Memodifikasi parameter `thread_handling` untuk mencegah Anda menyetelnya ke apa pun selain **`multiple-connections-per-thread`**, yang merupakan satu-satunya model yang didukung dengan kumpulan thread Aurora.
- Memperbaiki masalah yang disebabkan saat Anda menetapkan parameter `buffer_pool_size` atau `query_cache_size` menjadi lebih besar dari total memori klaster DB. Dalam keadaan ini, Aurora menetapkan parameter yang dimodifikasi ke nilai default, sehingga klaster DB dapat memulai dan tidak crash.
- Memperbaiki masalah di cache kueri yang membuat transaksi mendapatkan hasil baca usang jika tabel tidak valid di transaksi lain.

- Memperbaiki masalah di mana file binlog yang ditandai untuk penghapusan tidak langsung dihapus.
- Memperbaiki masalah di mana basis data yang dibuat dengan nama **`tmp`** diperlakukan sebagai basis data sistem yang disimpan dalam penyimpanan sementara dan tidak dipertahankan pada penyimpanan terdistribusi Aurora.
- Mengubah perilaku SHOW TABLES untuk mengecualikan tabel sistem internal tertentu. Perubahan ini membantu menghindari failover yang tidak perlu yang disebabkan oleh mysqldump mengunci semua file yang tercantum dalam SHOW TABLES, yang pada gilirannya mencegah penulisan di tabel sistem internal, yang menyebabkan failover.
- Memperbaiki masalah saat Replika Aurora salah dimulai ulang saat membuat tabel sementara dari kueri yang memanggil fungsi yang argumennya adalah kolom tabel InnoDB.
- Memperbaiki masalah yang terkait dengan konflik kunci metadata di simpul Replika Aurora yang menyebabkan Replika Aurora berada di belakang klaster DB primer dan akhirnya dimulai ulang.
- Memperbaiki dead latch di alur replikasi pada simpul pembaca, yang menyebabkan Replika Aurora tetringgal di belakang dan akhirnya memulai ulang.
- Memperbaiki masalah yang menyebabkan ketidakteraturan Replika Aurora terlalu banyak volume terenkripsi lebih besar dari 1 terabyte (TB).
- Peningkatan deteksi dead latch Replika Aurora dengan menggunakan cara yang lebih baik untuk membaca waktu jam sistem.
- Memperbaiki masalah di mana Replika Aurora dapat memulai ulang dua kali, bukannya sekali, setelah pembatalan pendaftaran oleh penulis.
- Memperbaiki masalah kinerja kueri lambat pada Replika Aurora yang terjadi saat statistik sementara menyebabkan perbedaan pada kolom indeks non-unik.
- Memperbaiki masalah di mana Replika Aurora dapat macet ketika pernyataan DDL direplikasi pada Replika Aurora pada saat bersamaan ketika Replika Aurora memproses kueri terkait.
- Mengubah perbaikan alur replikasi yang diperkenalkan di versi 1.10 dari diaktifkan secara default menjadi dinonaktifkan secara default. Peningkatan ini diperkenalkan untuk menerapkan pembaruan log stream pada cache buffer Replika Aurora, dan meskipun fitur ini membantu meningkatkan kinerja baca dan stabilitas di Replika Aurora, fitur ini meningkatkan lag replika di beban kerja tertentu.
- Memperbaiki masalah di mana kemunculan simultan pernyataan DDL yang sedang berlangsung dan Parallel Read Ahead yang tertunda di tabel yang sama menyebabkan kegagalan pernyataan selama fase commit transaksi DDL.
- Meningkatkan log umum dan log kueri lambat agar tetap bertahan saat memulai ulang klaster DB.

- Memperbaiki out-of-memory masalah untuk kueri tertentu yang berjalan lama dengan mengurangi konsumsi memori dalam modul ACL.
- Memperbaiki masalah mulai ulang yang terjadi saat tabel memiliki indeks non-spasial, terdapat predikat spasial dalam kueri, perencana memilih untuk menggunakan indeks non-spasial, dan perencana secara tidak tepat menekan kondisi spasial ke indeks.
- Memperbaiki masalah saat cluster DB dimulai ulang ketika ada penghapusan, pembaruan, atau pembersihan objek geospasial yang sangat besar yang disimpan secara eksternal (seperti). LOBs
- Memperbaiki masalah di mana simulasi kesalahan yang menggunakan ALTER SYSTEM SIMULATE ... FOR INTERVAL tidak berfungsi dengan baik.
- Memperbaiki masalah stabilitas yang disebabkan oleh pernyataan yang tidak valid tentang invarian yang salah di manajer kunci.
- Menonaktifkan dua perbaikan berikut pada Pencarian Teks Lengkap InnoDB yang diperkenalkan di versi 1.10 karena keduanya memunculkan masalah stabilitas untuk beberapa beban kerja yang berat:
  - Memperbarui cache hanya setelah permintaan baca ke Replika Aurora untuk meningkatkan kecepatan replikasi cache indeks pencarian teks penuh.
  - Melimpahkan tugas sinkronisasi cache ke thread terpisah segera setelah ukuran cache melewati 10% dari ukuran total, untuk menghindari kueri MySQL terhenti terlalu lama selama sinkronisasi cache FTS ke disk. (Bugs #22516559, #73816).

Integrasi perbaikan bug MySQL

- Menjalankan ALTER table DROP foreign key secara bersamaan dengan operasi DROP lain menyebabkan tabel menghilang. (Bug #16095573)
- Beberapa kueri INFORMATION_SCHEMA yang menggunakan ORDER BY tidak menggunakan optimalisasi filesort seperti sebelumnya. (Bug #16423536)
- FOUND_ROWS () mengembalikan jumlah baris yang salah pada tabel. (Bug #68458)
- Server mengalami kegagalan, bukan memunculkan pesan kesalahan saat terlalu banyak tabel sementara terbuka. (Bug #18948649)

Pembaruan mesin basis data Aurora MySQL: 2017-01-12 (versi 1.10.1) (Dihentikan)

Versi: 1.10.1

Aurora MySQL versi 1.10.1 adalah versi opt-in dan tidak digunakan untuk menambal instans basis data Anda. Versi ini tersedia untuk membuat instans Aurora baru dan untuk meningkatkan instans yang ada. Anda dapat menerapkan patch dengan memilih klaster dalam Konsol Amazon RDS, dengan memilih Tindakan Klaster, lalu memilih Tingkatkan Sekarang. Patching memerlukan mulai ulang basis data dengan waktu henti yang biasanya berlangsung selama 5-30 detik, kemudian Anda dapat melanjutkan menggunakan klaster DB Anda. Patch ini menggunakan model patching klaster di mana semua simpul dalam klaster Aurora ditambal pada saat yang sama.

Fitur baru

- Audit Lanjutan – Aurora MySQL menyediakan fitur Audit Lanjutan berkinerja tinggi, yang dapat Anda gunakan untuk mengaudit aktivitas basis data. Untuk informasi selengkapnya tentang pengaktifan dan penggunaan Audit Lanjutan, lihat Menggunakan Audit Lanjutan dengan klaster DB Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

- Memperbaiki masalah pada pengindeksan spasial saat membuat kolom dan menambahkan indeks dalam pernyataan yang sama.
- Memperbaiki masalah yang menyebabkan statistik spasial tidak bertahan di seluruh mulai ulang klaster DB.

Pembaruan mesin basis data Aurora MySQL: 2016-12-14 (versi 1.10) (Dihentikan)

Versi: 1.10

Fitur baru

- Zero downtime patch – Fitur ini memungkinkan instans DB untuk di-patch tanpa waktu henti. Artinya, peningkatan basis data dilakukan tanpa memutuskan sambungan aplikasi klien, atau mem-

boot ulang basis data. Pendekatan ini meningkatkan ketersediaan klaster DB Aurora Anda selama jadwal pemeliharaan. Perhatikan bahwa data sementara seperti itu di dalam skema kinerja direset selama proses peningkatan versi. Fitur ini berlaku untuk patch yang dikirim oleh layanan selama jadwal pemeliharaan serta patch yang diinisiasi pengguna.

Saat patch diinisiasi, layanan memastikan tidak ada kunci, transaksi, atau tabel sementara yang terbuka, kemudian menunggu jadwal yang sesuai sehingga basis data dapat di-patch dan dimulai ulang. Sesi aplikasi dipertahankan, meskipun ada penurunan throughput saat patch sedang berlangsung (selama sekitar 5 detik). Jika tidak ada jadwal yang cocok, maka patching disetel ke perilaku patching standar default.

Patching nol waktu henti dilakukan berdasarkan upaya terbaik, tergantung pada batasan tertentu sebagaimana dijelaskan sebagai berikut:

- Fitur ini saat ini berlaku untuk patching klaster DB simpul tunggal atau instans tulis dalam klaster DB multisimpul.
- Koneksi SSL tidak didukung dalam hubungannya dengan fitur ini. Jika ada koneksi SSL aktif, Amazon Aurora MySQL tidak akan melakukan patch nol waktu henti, dan sebaliknya akan mencoba kembali secara berkala untuk melihat apakah koneksi SSL telah diakhiri. Jika ada, patching nol waktu henti akan dilanjutkan. Jika koneksi SSL tetap ada setelah lebih dari beberapa detik, patching dengan waktu henti standar berlanjut.
- Fitur ini tersedia di Aurora release 1.10 dan seterusnya. Selanjutnya, kami akan mengidentifikasi rilis atau patch yang tidak dapat diterapkan dengan menggunakan patching nol waktu henti.
- Fitur ini tidak berlaku jika replikasi berdasarkan pencatatan log biner aktif.

- Pengindeksan spasial – Pengindeksan spasial meningkatkan kinerja kueri pada set data besar untuk kueri yang menggunakan data spasial. Untuk informasi selengkapnya tentang penggunaan pengindeksan spasial, lihat Amazon Aurora MySQL dan data spasial di Panduan Pengguna Amazon Aurora.

  Fitur ini dinonaktifkan secara default dan dapat diaktifkan dengan mengaktifkan mode lab Aurora. Untuk selengkapnya, lihat Mode lab Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

- Perbaikan alur replikasi – Aurora MySQL sekarang menggunakan mekanisme yang ditingkatkan untuk menerapkan pembaruan log stream ke cache buffer Replika Aurora. Fitur ini meningkatkan performa baca dan stabilitas pada Replika Aurora jika terdapat beban kerja yang berat pada master serta beban baca yang signifikan pada Replika. Fitur ini diaktifkan secara default.

- Peningkatan throughput untuk beban kerja dengan pembacaan dalam cache – Aurora MySQL kini menggunakan algoritma serentak bebas latch untuk menerapkan tampilan baca, yang akan menghasilkan throughput yang lebih baik untuk kueri baca yang dilayani oleh cache buffer. Sebagai hasil dari ini dan peningkatan lainnya, Amazon Aurora MySQL dapat mencapai throughput hingga 625K pembacaan per detik dibandingkan dengan 164K pembacaan per detik oleh MySQL 5.7 untuk beban kerja Select-only. SysBench
- Peningkatan throughput untuk beban kerja dengan kontensi baris panas – Aurora MySQL menggunakan rilis kunci baru yang meningkatkan kinerja, terutama ketika ada kontensi halaman panas (yaitu, banyak transaksi yang bersaing untuk baris di halaman yang sama). Dalam pengujian dengan tolok ukur TPC-C, hal ini dapat menghasilkan peningkatan throughput hingga 16x dalam transaksi per menit dibandingkan dengan MySQL 5.7. Fitur ini dinonaktifkan secara default dan dapat diaktifkan dengan mengaktifkan mode lab Aurora. Untuk selengkapnya, lihat Mode lab Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

- Kecepatan replikasi cache indeks pencarian teks penuh telah ditingkatkan dengan memperbarui cache hanya setelah permintaan baca ke Replika Aurora. Pendekatan ini menghindari pembacaan dari disk oleh thread replikasi.
- Memperbaiki masalah yang menyebabkan ketidakabsahan cache kamus tidak berfungsi di Replika Aurora untuk tabel yang memiliki karakter khusus dalam nama basis data atau nama tabel.
- Memperbaiki masalah `STUCK IO` selama migrasi data untuk simpul penyimpanan terdistribusi saat manajemen panas penyimpanan diaktifkan.
- Memperbaiki masalah di manajer kunci di mana pemeriksaan pernyataan gagal untuk thread kunci transaksi ketika menyiapkan untuk rollback atau commit transaksi.
- Memperbaiki masalah saat membuka tabel kamus yang rusak dengan memperbarui jumlah referensi ke entri tabel kamus dengan benar.
- Memperbaiki bug yang menyebabkan titik pembacaan minimum klaster DB dapat ditahan oleh Replika Aurora yang lambat.
- Memperbaiki potensi kebocoran memori di cache kueri.
- Memperbaiki bug yang menyebabkan Replika Aurora menempatkan kunci tingkat baris pada tabel saat kueri digunakan dalam pernyataan `IF` prosedur yang disimpan.

Integrasi perbaikan bug MySQL

- UNION tabel yang diperoleh memberikan hasil yang salah, IT '1=0/false'-clauses. (Bug #69471)
- Server mengalami crash pada ITEM_FUNC_GROUP_CONCAT::FIX_FIELDS pada eksekusi ke-2 dari prosedur yang disimpan. (Bug #20755389)
- Mencegah kueri MySQL terhenti terlalu lama selama sinkronisasi cache FTS ke disk dengan mengalihkan tugas sinkronisasi cache ke thread terpisah, segera setelah ukuran cache melewati 10% dari ukuran total. (Bug #22516559, #73816)

Pembaruan mesin basis data Aurora MySQL 2016-11-10 (versi 1.9.0, 1.9.1) (Dihentikan)

Versi: 1.9.0, 1.9.1

Fitur baru

- Peningkatan build indeks – Implementasi untuk membangun indeks sekunder kini beroperasi dengan membangun indeks dengan cara yang dari bawah ke atas, yang menghilangkan pemisahan halaman yang tidak perlu. Ini dapat mengurangi waktu yang diperlukan untuk membuat indeks atau membangun ulang tabel hingga 75% (berdasarkan Kelas instans DB `db.r3.8xlarge`). Fitur ini berada dalam mode lab di Aurora MySQL versi 1.7 dan diaktifkan secara default di Aurora versi 1.9 dan di atasnya. Untuk selengkapnya, lihat Mode lab Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Kompresi kunci (mode lab) – Implementasi ini secara signifikan mengurangi jumlah memori yang dikonsumsi manajer kunci hingga 66%. Manajer kunci dapat memperoleh lebih banyak kunci baris tanpa menemui pengecualian out-of-memory. Fitur ini dinonaktifkan secara default dan dapat diaktifkan dengan mengaktifkan mode lab Aurora. Untuk selengkapnya, lihat Mode lab Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Skema kinerja – Aurora MySQL sekarang menyertakan dukungan untuk skema kinerja dengan dampak minimal pada kinerja. Dalam pengujian kami menggunakan SysBench, mengaktifkan skema kinerja dapat menurunkan kinerja MySQL hingga 60%.

  SysBench pengujian cluster Aurora DB menunjukkan dampak pada kinerja yang 4x kurang dari MySQL. Menjalankan kelas instans `db.r3.8xlarge` DB menghasilkan 100K SQL writes/sec dan lebih dari 550K SQL reads/sec, bahkan dengan skema kinerja diaktifkan.

- Perbaikan perselisihan baris panas – Fitur ini mengurangi pemanfaatan CPU dan meningkatkan throughput ketika sejumlah kecil baris panas diakses oleh banyak koneksi. Fitur ini juga menghilangkan `error 188` saat terdapat persaingan baris panas.
- out-of-memoryPenanganan yang ditingkatkan - Ketika pernyataan SQL penguncian yang tidak penting dijalankan dan kumpulan memori yang dicadangkan dilanggar, Aurora memaksa pengembalian pernyataan SQL tersebut. Fitur ini membebaskan memori dan mencegah kerusakan mesin karena pengecualian. out-of-memory
- Pemilih baca cerdas - Implementasi ini meningkatkan latensi baca dengan memilih segmen penyimpanan optimal di antara segmen yang berbeda untuk setiap pembacaan, menghasilkan throughput baca yang lebih baik. SysBench pengujian telah menunjukkan peningkatan kinerja hingga 27% untuk beban kerja tulis.

Perbaikan

- Memperbaiki masalah yang dihadapi oleh Replika Aurora yang memiliki kunci bersama selama penyalaan mesin.
- Memperbaiki kemungkinan crash pada Replika Aurora ketika kursor tampilan baca dalam sistem pembersihan adalah NULL.

Pembaruan mesin basis data Aurora MySQL: 2016-10-26 (versi 1.8.1) (Dihentikan)

Versi: 1.8.1

Perbaikan

- Memperbaiki masalah yang menyebabkan kegagalan pada penyisipan massal yang menggunakan pemicu yang menginvokasi prosedur AWS Lambda .
- Memperbaiki masalah yang menyebabkan migrasi katalog gagal saat commit otomatis mati secara global.
- Menyelesaikan kegagalan koneksi ke Aurora saat menggunakan SSL dan meningkatkan grup Diffie-Hellman untuk menangani serangan. LogJam

Integrasi perbaikan bug MySQL

- OpenSSL mengubah parameter panjang kunci Diffie-Hellman karena masalah tersebut. LogJam (Bug #18367167)

Pembaruan mesin basis data Aurora MySQL: 2016-10-18 (versi 1.8) (Dihentikan)

Versi: 1.8

Fitur baru

- AWS Lambda integrasi — Anda sekarang dapat secara asinkron memanggil fungsi AWS Lambda dari cluster Aurora DB menggunakan prosedur. `mysql.lambda_async` Untuk informasi selengkapnya, lihat Menginvokasi fungsi Lambda dari klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.
- Muat data dari Amazon S3 – Sekarang Anda dapat memuat file teks atau XML dari bucket Amazon S3 ke dalam klaster DB Aurora menggunakan perintah `LOAD DATA FROM S3` atau `LOAD XML FROM S3`. Untuk informasi selengkapnya, lihat Memuat data ke dalam klaster DB Amazon Aurora MySQL dari file teks di bucket Amazon S3 di Panduan Pengguna Amazon Aurora.
- Migrasi katalog – Aurora sekarang mempertahankan metadata katalog di dalam volume klaster untuk mendukung versioning. Ini memungkinkan migrasi katalog yang lancar di seluruh versi dan pemulihan.
- Pemeliharaan dan patching tingkat-klaster – Aurora kini mengelola pembaruan pemeliharaan untuk seluruh klaster DB. Untuk informasi selengkapnya, lihat Memelihara klaster DB Amazon Aurora di Panduan Pengguna Amazon Aurora.

Perbaikan

- Memperbaiki masalah yang menyebabkan crash Replika Aurora saat tidak memberikan kunci metadata ke tabel DDL inflight.
- Mengizinkan Replika Aurora untuk memodifikasi tabel non-InnoDB untuk memfasilitasi rotasi file CSV log lambat dan umum apabila `log_output=TABLE`.
- Memperbaiki lag saat memperbarui statistik dari instans primer ke Replika Aurora. Tanpa perbaikan ini, statistik Replika Aurora dapat keluar dari sinkronisasi dengan statistik instans primer

dan menghasilkan rencana kueri yang berbeda (dan mungkin kurang berkinerja) pada Replika Aurora.

- Memperbaiki kondisi race yang memastikan bahwa Replika Aurora tidak mendapatkan kunci.
- Memperbaiki skenario langka yang menyebabkan kegagalan saat Replika Aurora mendaftarkan atau membatalkan pendaftaran yang tidak sesuai dengan instans primer.
- Memperbaiki kondisi race yang dapat mengakibatkan deadlock pada instans `db.r3.large` saat membuka atau menutup volume.
- Memperbaiki out-of-memory masalah yang dapat terjadi karena kombinasi beban kerja tulis yang besar dan kegagalan dalam layanan Penyimpanan Terdistribusi Aurora.
- Memperbaiki masalah konsumsi CPU yang tinggi karena thread pembersihan berputar dengan adanya transaksi yang berlangsung lama.
- Memperbaiki masalah saat menjalankan kueri skema informasi untuk mendapatkan informasi tentang kunci di bawah beban berat.
- Memperbaiki masalah pada proses diagnostik yang, meskipun jarang terjadi, menyebabkan penulisan Aurora ke simpul penyimpanan menjadi terhenti dan memulai ulang/melakukan failover.
- Memperbaiki kondisi di mana tabel yang berhasil dibuat mungkin dihapus selama pemulihan crash jika crash tersebut terjadi saat pernyataan `CREATE TABLE [if not exists]` sedang ditangani.
- Memperbaiki kasus di mana prosedur rotasi log rusak ketika log umum dan log lambat tidak disimpan di disk menggunakan mitigasi katalog.
- Memperbaiki crash saat seorang pengguna membuat tabel sementara untuk fungsi yang ditentukan pengguna, kemudian menggunakan fungsi yang ditentukan pengguna dalam daftar kueri tertentu.
- Memperbaiki crash yang terjadi saat memutar ulang peristiwa GTID. GTID tidak didukung oleh Aurora MySQL.

Integrasi perbaikan bug MySQL:

- Saat menghapus semua indeks pada suatu kolom dengan beberapa indeks, InnoDB gagal memblokir operasi DROP INDEX ketika sebuah batasan kunci asing memerlukan sebuah indeks. (Bug #16896810)
- Mengatasi crash penambahan batasan kunci asing. (Bug #16413976)
- Memperbaiki crash saat mengambil kursor di prosedur yang disimpan, dan menganalisis atau mem-flush tabel pada saat bersamaan. (Bug # 18158639)

- Memperbaiki bug kenaikan otomatis saat pengguna mengubah tabel untuk mengubah nilai AUTO_INCREMENT menjadi kurang dari nilai kolom peningkatan otomatis maksimum. (Bug # 16310273)

Pembaruan mesin basis data Aurora MySQL: 2016-09-20 (versi 1.7.1) (Dihentikan)

Versi: 1.7.1

Perbaikan

- Memperbaiki masalah di mana Replika Aurora mengalami crash jika cache pencarian teks lengkap InnoDB penuh.
- Memperbaiki masalah di mana mesin basis data mengalami crash jika ada thread pekerja di dalam thread pool menunggu dirinya sendiri.
- Memperbaiki masalah ketika Replika Aurora macet jika kunci metadata di tabel menyebabkan deadlock.
- Memperbaiki masalah yang menyebabkan mesin basis data crash akibat kondisi race antara dua thread pekerja di dalam thread pool.
- Memperbaiki masalah di mana failover yang tidak diperlukan terjadi di bawah beban berat jika agen pemantau tidak mendeteksi kemajuan operasi tulis ke subsistem penyimpanan terdistribusi.

Pembaruan mesin basis data Aurora MySQL: 2016-08-30 (versi 1.7.0) (Dihentikan)

Versi: 1.7.0

Fitur baru

- Penjadwal sadar NUMA – Penjadwal tugas untuk mesin Aurora MySQL sekarang menyadari Non-Uniform Memory Access (NUMA). Ini meminimalkan perselisihan soket lintas-CPU yang menghasilkan throughput kinerja yang lebih baik untuk kelas instans DB `db.r3.8xlarge`.
- Read-ahead paralel beroperasi secara asinkron di latar belakang – Read-ahead paralel telah direvisi untuk meningkatkan kinerja dengan menggunakan thread khusus untuk mengurangi perselisihan thread.

- Peningkatan build indeks (mode lab) – Implementasi untuk membangun indeks sekunder kini beroperasi dengan membangun indeks dengan cara yang dari bawah ke atas, yang menghilangkan pemisahan halaman yang tidak perlu. Ini dapat mengurangi waktu yang diperlukan untuk membuat indeks atau membangun ulang tabel. Fitur ini dinonaktifkan secara default dan dapat diaktifkan dengan mengaktifkan mode lab Aurora. Untuk selengkapnya, lihat Mode lab Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Perbaikan

- Memperbaiki masalah yang membuat koneksi membutuhkan waktu lama jika terjadi lonjakan dalam jumlah koneksi yang diminta untuk sebuah instans.
- Memperbaiki masalah di mana terjadi crash jika ALTER TABLE dijalankan pada tabel yang dipartisi yang tidak menggunakan InnoDB.
- Memperbaiki masalah di mana beban kerja tulis dapat menyebabkan failover.
- Memperbaiki pernyataan yang keliru yang menyebabkan kegagalan jika RENAME TABLE dijalankan di tabel yang dipartisi.
- Meningkatkan stabilitas saat melakukan rollback transaksi selama beban kerja insert-heavy.
- Memperbaiki masalah yang menyebabkan indeks pencarian teks penuh tidak dapat diatasi di Replika Aurora.

Integrasi perbaikan bug MySQL

- Meningkatkan skalabilitas dengan mempartisi kunci LOCK_grant. (Port WL #8355)
- Kursor pembukaan pada SELECT dalam prosedur yang disimpan menyebabkan segfault. (Port Bug #16499751)
- MySQL memberikan hasil yang salah dengan penggunaan khusus. (Bug #11751794)
- Crash di GET_SEL_ARG_FOR_KEYPART – yang disebabkan oleh patch untuk bug #11751794. (Bug #16208709)
- Hasil yang salah untuk kueri sederhana dengan GROUP BY. (Bug #17909656)
- Baris tambahan pada kueri semijoin dengan predikat rentang. (Bug #16221623)
- Penambahan klausa ORDER BY setelah subkueri IN dapat menyebabkan baris duplikat dikembalikan. (Bug #16308085)

- Crash dengan penjelasan untuk kueri dengan pemindaian longgar untuk GROUP BY, MyISAM. (Bug #16222245)
- Pemindaian indeks longgar dengan predikat int yang dikutip menghasilkan data acak. (Bug #16394084)
- Jika optimalisasi menggunakan pemindaian indeks longgar, server dapat keluar saat mencoba membuat tabel sementara. (Bug #16436567)
- COUNT(DISTINCT) seharusnya tidak menghitung nilai NULL, tetapi dihitung saat pengoptimal menggunakan pemindaian indeks longgar. (Bug #17222452)
- Jika sebuah kueri memiliki MIN()/MAX() dan aggregate_function(DISTINCT) (misalnya, SUM(DISTINCT)) dan dijalankan menggunakan pemindaian indeks longgar, nilai hasil MIN()/MAX() diatur dengan tidak tepat. (Bug #17217128)

Pembaruan mesin basis data Aurora MySQL: 2016-06-01 (versi 1.6.5) (Dihentikan)

Versi: 1.6.5

Fitur baru

- Penyimpanan Log Biner yang efisien – Penyimpanan log biner yang efisien sekarang diaktifkan secara default untuk semua klaster DB Aurora MySQL, dan tidak dapat dikonfigurasi. Penyimpanan log biner yang efisien diperkenalkan pada pembaruan April 2016. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL: 2016-04-06 (versi 1.6) (Dihentikan).

Perbaikan

- Peningkatan stabilitas untuk Aurora Replika ketika instans primer sedang mengalami beban kerja berat.
- Meningkatkan stabilitas untuk Replika Aurora ketika menjalankan kueri pada tabel yang dipartisi dan tabel dengan karakter khusus pada nama tabel.
- Memperbaiki masalah koneksi saat menggunakan koneksi aman.

Integrasi perbaikan bug MySQL

- SLAVE CAN'T CONTINUE REPLICATION AFTER MASTER'S CRASH RECOVERY (Port Bug #17632285)

Pembaruan mesin basis data Aurora MySQL: 2016-04-06 (versi 1.6) (Dihentikan)

Versi: 1.6

Pembaruan ini mencakup perbaikan berikut:

Fitur baru

- Read-ahead paralel – Read-ahead paralel sekarang diaktifkan secara default untuk semua klaster DB Aurora MySQL, dan tidak dapat dikonfigurasi. Read-ahead paralel diperkenalkan pada pembaruan Desember 2015. Untuk informasi selengkapnya, lihat Pembaruan mesin basis data Aurora MySQL: 2015-12-03 (versi 1.4) (Dihentikan).

  Selain mengaktifkan read-ahead paralel secara default, rilis ini mencakup perbaikan berikut untuk read-ahead paralel:

  - Meningkatkan logika sehingga read-ahead paralel menjadi kurang agresif, yang bermanfaat saat klaster DB Anda menghadapi banyak beban kerja paralel.
  - Meningkatkan stabilitas pada tabel yang lebih kecil.
- Penyimpanan Log Biner yang efisien (mode lab)— File log biner MySQL sekarang disimpan lebih efisien di Aurora MySQL. Implementasi penyimpanan baru memungkinkan file log biner dihapus lebih awal dan meningkatkan kinerja sistem untuk instans dalam klaster DB Aurora MySQL yang merupakan master replikasi log biner.

  Untuk memungkinkan penyimpanan log biner yang efisien, atur parameter `aurora_lab_mode` ke 1 dalam kelompok parameter untuk instans primer Anda atau Replika Aurora. Parameter `aurora_lab_mode` adalah parameter tingkat instans yang ada di kelompok parameter `default.aurora5.6` secara default. Untuk informasi tentang memodifikasi grup parameter DB, lihat Memodifikasi parameter dalam grup parameter DB di Panduan Pengguna Amazon Aurora. Untuk informasi tentang grup parameter dan Aurora MySQL, lihat Parameter konfigurasi Aurora MySQL di Panduan Pengguna Amazon Aurora.

Hanya aktifkan penyimpanan log biner yang efisien untuk instans di klaster DB Aurora MySQL yang merupakan instans master replikasi log biner MySQL.

- Variabel sistem AURORA_VERSION – Anda sekarang bisa mendapatkan versi Aurora dari klaster DB Aurora MySQL Anda dengan membuat kueri variabel sistem `AURORA_VERSION`.

  Untuk mendapatkan versi Aurora, gunakan salah satu kueri berikut:

  ```
  select AURORA_VERSION();
  select @@aurora_version;
  show variables like '%version';
  ```

  Anda juga dapat melihat versi Aurora di Konsol Manajemen AWS saat Anda memodifikasi cluster DB, atau dengan memanggil describe-db-engine-versions AWS CLI perintah atau operasi API Deskripsikan DBEngine Versi.

- Metrik penggunaan memori manajer kunci – Informasi tentang penggunaan memori manajer kunci kini tersedia sebagai metrik.

  Untuk mendapatkan metrik penggunaan memori manajer kunci, gunakan salah satu kueri berikut:

  ```
  show global status where variable_name in ('aurora_lockmgr_memory_used');
  select * from INFORMATION_SCHEMA.GLOBAL_STATUS where variable_name in
   ('aurora_lockmgr_memory_used');
  ```

Perbaikan

- Meningkatkan stabilitasi selama pemulihan transaksi XA dan binlog.
- Memperbaiki masalah memori yang disebabkan oleh sejumlah besar koneksi.
- Meningkatkan akurasi dalam metrik berikut: `Read Throughput`, `Read IOPS`, `Read Latency`, `Write Throughput`, `Write IOPS`, `Write Latency`, dan `Disk Queue Depth`.
- Memperbaiki masalah stabilitas yang menyebabkan mulai ulang yang lambat untuk instans besar setelah crash.
- Meningkatkan konkurensi dalam kamus data terkait mekanisme sinkronisasi dan pengosongan cache.

- Peningkatan stabilitas dan kinerja untuk Replika Aurora:
  - Memperbaiki masalah stabilitas untuk Replika Aurora selama beban kerja tulis yang berat atau burst untuk instans primer.
  - Peningkatan lag replika untuk kelas instans db.r3.4xlarge dan db.r3.8xlarge.
  - Peningkatan kinerja dengan mengurangi susunan antara aplikasi penyalinan ulang log RDS dan pembacaan bersamaan pada Replika Aurora.
  - Memperbaiki masalah untuk statistik penyegaran pada Replika Aurora untuk statistik yang baru dibuat atau diperbarui.
  - Peningkatan stabilitas untuk Replika Aurora ketika ada banyak transaksi pada instans primer dan pembacaan serentak pada Replika Aurora di seluruh data yang sama.
  - Meningkatkan stabilitas untuk Replika Aurora saat menjalankan pernyataan `UPDATE` dan `DELETE` dengan pernyataan `JOIN`.
  - Meningkatkan stabilitas untuk Replika Aurora saat menjalankan pernyataan `INSERT ... SELECT`.

Integrasi perbaikan bug MySQL

- BACKPORT Bug #18694052 FIX FOR ASSERTION `!M_ORDERED_REC_BUFFER' FAILED TO 5.6 (Port Bug #18305270)
- SEGV IN MEMCPY(), HA_PARTITION::POSITION (Port Bug # 18383840)
- WRONG RESULTS WITH PARTITIONING,INDEX_MERGE AND NO PK (Port Bug # 18167648)
- FLUSH TABLES FOR EXPORT: ASSERTION IN HA_PARTITION::EXTRA (Port Bug # 16943907)
- SERVER CRASH IN VIRTUAL HA_ROWS HANDLER::MULTI_RANGE_READ_INFO_CONST (Port Bug # 16164031)
- RANGE OPTIMIZER CRASHES IN SEL_ARG::RB_INSERT() (Port Bug # 16241773)

Pembaruan mesin basis data Aurora MySQL: 2016-01-11 (versi 1.5) (Dihentikan)

Versi: 1.5

Pembaruan ini mencakup perbaikan berikut:

Perbaikan

- Memperbaiki jeda 10 detik operasi tulis untuk instans idle selama deployment penyimpanan Aurora.
- Read-ahead logis sekarang bekerja ketika `innodb_file_per_table` diatur ke No. Untuk informasi selengkapnya tentang read-ahead logis, lihat Pembaruan mesin basis data Aurora MySQL: 2015-12-03 (versi 1.4) (Dihentikan).
- Memperbaiki masalah pada Replikas Aurora yang terhubung kembali ke instans primer. Peningkatan ini juga memperbaiki masalah ketika Anda menentukan nilai yang besar untuk parameter `quantity` saat menguji kegagalan Replika Aurora menggunakan kueri injeksi kesalahan. Untuk informasi selengkapnya, lihat Menguji kegagalan replika Aurora di Panduan Pengguna Amazon Aurora.
- Memperbaiki pemantauan Replika Aurora yang tertinggal dan memulai kembali.
- Memperbaiki masalah yang menyebabkan Replika Aurora menjadi lambat, dibatalkan pendaftarannya, lalu dimulai ulang.
- Memperbaiki masalah saat Anda menjalankan perintah `show innodb status` saat deadlock.
- Memperbaiki masalah dengan failover untuk instans yang lebih besar selama throughput tulis yang tinggi.

Integrasi perbaikan bug MySQL

- Mengatasi perbaikan yang belum selesai dalam pencarian teks penuh MySQL yang memengaruhi tabel dengan nama basis data yang dimulai dengan sebuah digit. (Port Bug #17607956)

Pembaruan mesin basis data Aurora MySQL: 2015-12-03 (versi 1.4) (Dihentikan)

Versi: 1.4

Pembaruan ini mencakup perbaikan berikut:

Fitur baru

- Sisipan Cepat – Mempercepat penyisipan paralel yang diurutkan berdasarkan kunci primer. Untuk informasi selengkapnya, lihat peningkatan kinerja Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.
- Performa baca set data besar – Aurora MySQL secara otomatis mendeteksi beban kerja IO yang berat dan meluncurkan lebih banyak thread untuk meningkatkan kinerja klaster DB. Penjadwal Aurora mengamati aktivitas IO dan memutuskan untuk menyesuaikan jumlah thread yang optimal di dalam sistem, dengan cepat menyesuaikan antara beban kerja berat IO dan CPU yang berat dengan beban kerja rendah.
- Parallel read-ahead – Meningkatkan kinerja pemindaian B-Tree yang terlalu besar untuk memori yang tersedia pada instans primer Anda atau Replika Aurora (termasuk kueri rentang). Read-ahead paralel secara otomatis mendeteksi pola baca halaman dan mengambil halaman di awal ke dalam cache buffer sebelum diperlukan. Read-ahead paralel bekerja dari beberapa tabel di waktu yang sama dengan transaksi yang sama.

Perbaikan:

- Memperbaiki masalah ketersediaan basis data Aurora selama deployment penyimpanan Aurora.
- Menerapkan batas `max_connection` dengan benar.
- Meningkatkan pembersihan binlog di mana Aurora adalah master binlog dan basis data memulai ulang setelah muatan data yang berat.
- Memperbaiki masalah manajemen memori dengan cache tabel.
- Menambahkan dukungan untuk halaman besar dalam cache buffer memori bersama untuk pemulihan yang lebih cepat.
- Memperbaiki masalah penyimpanan lokal thread yang tidak diinisialisasi.
- Mengizinkan 16K koneksi secara default.
- Pool thread dinamis untuk beban kerja berat IO.
- Memperbaiki masalah dengan tampilan yang tidak divalidasi dengan benar yang melibatkan UNION dalam cache kueri.
- Memperbaiki masalah stabilitas dengan thread statistik kamus.
- Memperbaiki kebocoran memori dalam subsistem kamus yang terkait dengan pengosongan data.
- Memperbaiki masalah latensi baca tinggi di Replika Aurora saat ada beban tulis yang sangat rendah di master.

- Memperbaiki masalah stabilitas pada Replika Aurora ketika melakukan operasi pada tabel terpartisi DDL seperti ALTER TABLE … REORGANIZE PARTITION pada master.
- Memperbaiki masalah stabilitas di Replika Aurora selama pertumbuhan volume.
- Memperbaiki masalah kinerja untuk pemindaian pada indeks tanpa klaster di Replika Aurora.
- Memperbaiki masalah stabilitas yang membuat lag Replika Aurora dan akhirnya tidak terdaftar dan dimulai kembali.

Integrasi perbaikan bug MySQL

- SEGV in FTSPARSE(). (Bug #16446108)
- Kamus data InnoDB tidak diperbarui saat mengubah nama kolom. (Bug #19465984)
- Crash FTS setelah mengubah nama tabel ke basis data yang berbeda. (Bug #16834860)
- Gagal menyiapkan pemicu pada tabel yang terpotong menyebabkan kesalahan 1054. (Bug #18596756)
- Perubahan metadata dapat menyebabkan masalah pada eksekusi pemicu. (Bug #18684393)
- Materialisasi tidak dipilih untuk bidang UTF8 VARCHAR yang panjang. (Bug #17566396)
- Rencana eksekusi yang buruk saat ORDER BY dengan batas X. (Bug #16697792)
- Backport bug #11765744 KE 5.1, 5.5, DAN 5.6. (Bug #17083851)
- Masalah mutex di SQL/SQL_SHOW.CC menghasilkan. SIG6 Sumber cenderung FILL_VARIABLES. (Bug #20788853)
- Backport bug #18008907 ke versi 5.5+. (Bug #18903155)
- Menyesuaikan perbaikan untuk kesalahan luapan tumpukan di MySQL 5.7. (Bug #19678930)

Pembaruan mesin basis data Aurora MySQL: 2015-10-16 (versi 1.2, 1.3) (Dihentikan)

Versi: 1.2, 1.3

Pembaruan ini mencakup perbaikan berikut:

Perbaikan

- out-of-memoryMasalah yang diselesaikan di pengelola kunci baru dengan transaksi yang berjalan lama

- Mengatasi kerentanan keamanan saat mereplikasi dengan basis data non-RDS for MySQL
- Diperbarui untuk memastikan bahwa penulisan quorum mencoba ulang secara benar dengan kegagalan penyimpanan
- Diperbarui untuk melaporkan lag replika secara lebih akurat
- Meningkatkan kinerja dengan mengurangi perselisihan saat banyak transaksi bersamaan mencoba mengubah baris yang sama
- Mengatasi invalidasi cache kueri untuk tampilan yang dibuat dengan menggabungkan dua tabel
- Menonaktifkan cache kueri untuk transaksi dengan isolasi `UNCOMMITTED_READ`

Perbaikan

- Kinerja lebih baik untuk kueri katalog lambat pada cache hangat
- Meningkatkan konkurensi dalam statistik kamus
- Stabilitas yang lebih baik untuk manajer sumber daya cache kueri baru, manajemen jangkauan, file yang disimpan di penyimpanan cerdas Amazon Aurora, dan tulis batch rekaman log

Integrasi perbaikan bug MySQL

- Mematikan kueri di dalam innodb menyebabkan crash dengan pernyataan. (Bug #1608883)
- Untuk kegagalan dalam membuat thread baru untuk penjadwal peristiwa, pelaksanaan peristiwa, atau koneksi baru, tidak ada pesan yang dituliskan ke log kesalahan. (Bug #16865959)
- Jika satu koneksi mengubah basis data default dan secara bersamaan koneksi lain menjalankan SHOW PROCESSLIST, koneksi kedua dapat mengakses memori yang tidak valid ketika mencoba menampilkan memori basis data default koneksi pertama. (Bug #11765252)
- PURGE BINARY LOGS berdasarkan desain tidak menghapus file log biner yang sedang digunakan atau aktif, tetapi tidak memberikan pemberitahuan apa pun saat hal ini terjadi. (Bug #13727933)
- Untuk beberapa pernyataan, kebocoran memori dapat terjadi ketika pengoptimal menghapus klausa subkueri yang tidak diperlukan. (Bug #15875919)
- Selama shutdown, server dapat mencoba mengunci mutex yang tidak diinisialisasi. (Bug #16016493)
- Pernyataan yang disiapkan yang menggunakan GROUP_CONCAT() dan klausa ORDER BY yang menyebutkan beberapa kolom dapat menyebabkan server keluar. (Bug #16075310)

- Instrumentasi Skema Kinerja tidak ada untuk thread pekerja replika. (Bug #16083949)
- `STOP SLAVE` dapat menyebabkan deadlock saat diterbitkan secara bersamaan dengan pernyataan seperti SHOW STATUS yang mengambil nilai untuk satu atau beberapa variabel status `Slave_retried_transactions`, `Slave_heartbeat_period`, `Slave_received_heartbeats`, `Slave_last_heartbeat`, atau `Slave_running`. (Bug #16088188)
- Kueri teks penuh yang menggunakan mode Boolean dapat memberikan hasil nol pada beberapa kasus di mana istilah pencarian adalah frasa yang dikutip. (Bug #16206253)
- Upaya pengoptimal untuk menghapus klausa subkueri redundan memunculkan pernyataan saat menjalankan pernyataan yang disiapkan dengan subkueri dalam klausa ON dari sambungan dalam subkueri. (Bug #16318585)
- GROUP_CONCAT tidak stabil, crash di ITEM_SUM::CLEAN_UP_AFTER_REMOVAL. (Bug #16347450)
- Mencoba mengganti daftar stopword pencarian teks lengkap (FTS) InnoDB default dengan membuat tabel InnoDB dengan struktur yang sama seperti INFORMATION_SCHEMA.INNODB_FT_DEFAULT_STOPWORD akan mengakibatkan kesalahan. (Bug #16373868)
- Setelah thread klien pada pekerja menjalankan FLUSH TABLES WITH READ LOCK dan diikuti oleh beberapa pembaruan pada master, pekerja tersebut macet saat mengeksekusi `SHOW SLAVE STATUS`. (Bug #16387720)
- Saat menguraikan string pencarian yang dibatasi seperti "abc-def" dalam pencarian teks lengkap, InnoDB sekarang menggunakan pembatas kata yang sama dengan MyISAM. (Bug #16419661)
- Crash di FTS_AST_TERM_SET_WILDCARD. (Bug #16429306)
- SEGFAULT di FTS_AST_VISIT() untuk uji FTS RQG. (Bug # 16435855)
- Untuk build debug, ketika pengoptimal menghapus Item_ref yang menunjuk ke subkueri, hal ini menyebabkan server keluar. (Bug #16509874)
- Pencarian teks lengkap pada tabel InnoDB gagal pada pencarian frasa literal yang dikombinasikan dengan operator + atau -. (Bug #16516193)
- `START SLAVE`gagal ketika server dimulai dengan opsi -- master-info-repository =TABLE relay-log-info-repository =TABLE dan dengan komit otomatis disetel ke 0, bersama dengan. `--skip-slave-start` (Bug #16533802)
- Hasil pencarian teks lengkap (FTS) InnoDB yang sangat besar dapat menghabiskan terlalu banyak memori. (Bug #16625973)

- Dalam build debug, pernyataan dapat terjadi dalam OPT_CHECK_ORDER_BY ketika menggunakan biner secara langsung dalam string pencarian, karena biner mungkin menyertakan byte NULL dan karakter lain yang tidak bermakna. (Bug #16766016)
- Untuk beberapa pernyataan, kebocoran memori dapat terjadi ketika pengoptimal menghapus klausa subkueri yang tidak diperlukan. (Bug #16807641)
- Ada kemungkinan untuk menyebabkan deadlock setelah menerbitkan FLUSH TABLES WITH READ LOCK dengan menerbitkan `STOP SLAVE` dalam koneksi yang baru dengan pekerja, kemudian menerbitkan `SHOW SLAVE STATUS` menggunakan koneksi asli. (Bug #16856735)
- GROUP_CONCAT() dengan pemisah yang tidak valid dapat menyebabkan server keluar. (Bug #16870783)
- Server melakukan penguncian berlebihan di mutex LOCK_active_mi dan active_mi->rli->data_lock untuk pernyataan 'pola' SHOW STATUS LIKE apa pun, bahkan ketika pola tidak cocok dengan variabel status yang menggunakan mutex tersebut (`Slave_heartbeat_period`, `Slave_last_heartbeat`, `Slave_received_heartbeats`, `Slave_retried_transactions`, `Slave_running`). (Bug #16904035)
- Pencarian teks penuh menggunakan modifier MODE BOOLEAN akan menyebabkan kegagalan pernyataan. (Bug #16927092)
- Pencarian teks lengkap pada tabel InnoDB gagal pada pencarian yang menggunakan operator boolean +. (Bug #17280122)
- Deadlock 4 arah: zombies, purging binlogs, show processlist, show binlogs. (Bug #17283409)
- Saat thread SQL yang menunggu kunci commit dimatikan dan dimulai ulang, ini menyebabkan transaksi dilewatkan pada pekerja. (Bug #17450876)
- Kegagalan pencarian teks penuh InnoDB akan terjadi karena token yang "tidak diakhiri". String dan panjang string harus diteruskan untuk perbandingan string. (Bug #17659310)
- Sejumlah besar tabel InnoDB yang terpartisi dapat menggunakan lebih banyak memori saat digunakan dalam MySQL 5.6 atau 5.7 dibandingkan memori yang digunakan oleh tabel yang sama yang digunakan dalam rilis Server MySQL sebelumnya. (Bug #17780517)
- Untuk kueri teks penuh, kegagalan untuk memeriksa bahwa num_token kurang dari max_proximity_item dapat menghasilkan sebuah pernyataan. (Bug #18233051)
- Kueri tertentu untuk tabel INFORMATION_SCHEMA TABLES dan COLUMNS dapat menyebabkan penggunaan memori yang berlebihan saat ada sejumlah besar tabel InnoDB yang kosong. (Bug #18592390)

- Saat menerapkan transaksi, sebuah bendera sekarang digunakan untuk memeriksa apakah thread telah dibuat, bukan memeriksa thread itu sendiri, yang menggunakan lebih banyak sumber daya, terutama ketika menjalankan server dengan master_info_repository=TABLE. (Bug #18684222)
- Jika thread klien pada pekerja menjalankan FLUSH TABLES WITH READ LOCK sementara master menjalankan DML, menjalankan `SHOW SLAVE STATUS` di klien yang sama menjadi diblokir, sehingga menyebabkan deadlock. (Bug #19843808)
- Mengurutkan berdasarkan hasil GRUP_CONCAT() dapat menyebabkan server keluar. (Bug #19880368)

Pembaruan mesin basis data Aurora MySQL: 2015-08-24 (versi 1.1) (Dihentikan)

Versi: 1.1

Pembaruan ini mencakup perbaikan berikut:

- Peningkatan stabilitas replikasi saat mereplikasi dengan basis data MySQL (replikasi binlog). Untuk informasi tentang replikasi Aurora MySQL dengan MySQL, lihat Replikasi dengan Amazon Aurora di Panduan Pengguna Amazon Aurora.
- Batas 1 gigabyte (GB) pada ukuran log relay terakumulasi untuk klaster DB Aurora MySQL yang merupakan pekerja replikasi. Hal ini meningkatkan kualitas pengelolaan file untuk klaster DB Aurora.
- Perbaikan stabilitas di bidang read-ahead, hubungan kunci asing rekursif, dan replikasi Aurora.
- Integrasi perbaikan bug MySQL.
  - Basis data InnoDB dengan nama-nama yang dimulai dengan sebuah digit menyebabkan kesalahan pengurai pencarian teks penuh (FTS). (Bug #17607956)
  - Pencarian teks penuh InnoDB gagal di basis data yang namanya dimulai dengan sebuah digit. (Bug #17161372)
  - Untuk basis data InnoDB pada Windows, ID objek pencarian teks penuh (FTS) tidak dalam format heksadesimal yang diharapkan. (Bug #16559254)
  - Regresi kode yang diperkenalkan di MySQL 5.6 berdampak negatif pada kinerja DROP TABLE dan ALTER TABLE. Hal ini dapat menyebabkan penurunan kinerja antara MySQL Server 5.5.x dan 5.6.x. (Bug #16864741)

- Menyederhanakan pencatatan log untuk mengurangi ukuran file log dan jumlah penyimpanan yang diperlukan.

Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL

Bagian berikut mengidentifikasi bug MySQL yang telah diperbaiki oleh pembaruan mesin basis data Aurora MySQL.

Topik

- Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 3.x
- Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 2.x
- Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 1.x

Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 3.x

Versi Aurora yang kompatibel dengan MySQL 8.0 berisi semua perbaikan bug MySQL hingga versi Kompatibilitas MySQL yang sesuai. Tabel berikut mengidentifikasi bug MySQL tambahan yang telah diperbaiki oleh pembaruan mesin basis data Aurora MySQL, dan di pembaruan mana bug tersebut diperbaiki.

Pembaruan mesin basis data	Versi kompatibel MySQL	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin database Aurora MySQL 2024-11-18 (versi 3.08.0, kompatibel dengan	8.0.39	3.08.0	• Memperbaiki masalah yang menyebabkan `NULL` nilai dihilangkan secara tidak benar dari kumpulan hasil untuk kueri tertentu yang memiliki keduanya `JOIN` dan operasi. `UNION` (Perbaikan Bug Komunitas #114301)

Pembaruan mesin basis data	Versi kompatibel MySQL	Versi	Bug MySQL yang diperbaiki
MySQL 8.0.39)			

Pembaruan mesin basis data	Versi kompatibel MySQL	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin database Aurora MySQL 2024-06-04 (versi 3.07.0) (Usang)	8.0.36	3.07.0	• Memperbaiki masalah di mana nilai baris cache dapat dihitung secara tidak benar, menyebabkan kegagalan selama restart database pada instance berbasis Graviton. (Perbaikan Bug Komunitas #35479763) • Memperbaiki masalah di mana beberapa contoh subkueri dalam rutinitas tersimpan tidak selalu ditangani dengan benar. (Perbaikan Bug Komunitas #35377192) • Memperbaiki masalah yang dapat menyebabkan penggunaan CPU lebih tinggi karena rotasi sertifikat TLS latar belakang (Perbaikan Bug Komunitas #34284186). • Memperbaiki masalah di mana InnoDB mengizinkan penambahan `INSTANT` kolom ke tabel dalam skema sistem MySQL di versi MySQL Aurora MySQL lebih rendah dari 3.05, yang dapat menyebabkan server tiba-tiba menutup (instance database restart) setelah upgrade ke Aurora MySQL

Pembaruan mesin basis data	Versi kompatibel MySQL	Versi	Bug MySQL yang diperbaiki
			versi 3.05.0. (Perbaikan Bug Komunitas #35625510).

Pembaruan mesin basis data	Versi kompatibel MySQL	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin database Aurora MySQL 2024-03-07 (versi 3.06.0) (Usang)	8.0.34	3.06.0	• Memperbaiki masalah di mana nilai baris cache dapat dihitung secara tidak benar, menyebabkan kegagalan selama restart database pada instance Graviton. (Perbaikan Bug Komunitas #35479763) • Memperbaiki masalah di mana beberapa contoh subkueri dalam rutinitas tersimpan tidak selalu ditangani dengan benar. (Perbaikan Bug Komunitas #35377192) • Memperbaiki masalah yang dapat menyebabkan penggunaan CPU lebih tinggi karena rotasi sertifikat TLS latar belakang. (Perbaikan Bug Komunitas #34284186) • Memperbaiki masalah di mana InnoDB mengizinkan penambahan `INSTANT` kolom ke tabel dalam skema sistem MySQL di versi MySQL Aurora MySQL lebih rendah dari 3.05, yang dapat menyebabkan server tiba-tiba menutup (instance database restart) setelah upgrade ke Aurora MySQL

Pembaruan mesin basis data	Versi kompatibel MySQL	Versi	Bug MySQL yang diperbaiki
			versi 3.05.0. (Perbaikan Bug Komunitas #35625510)
Pembaruan mesin database Aurora MySQL 2024-01-31 (versi 3.05.2) (Usang)	8.0.32	3.05.2	• Memperbaiki masalah saat `records_in_range` melakukan pembacaan disk dalam jumlah berlebihan untuk `INSERT` operasi, yang menyebabkan penurunan kinerja secara bertahap. (Perbaikan Bug Komunitas #34976138)
Pembaruan mesin database Aurora MySQL 2023-11-21 (versi 3.05.1) (Usang)	8.0.32	3.05.1	• Memperbaiki masalah di InnoDB ketika, jika tabel MySQL dalam skema sistem memiliki kolom yang ditambahkan `INSTANT ADD` antara Aurora MySQL versi 3.01 melalui Aurora MySQL versi 3.04, dan setelah Aurora MySQL ditingkatkan ke versi 3.05.0, pada tabel ini akan mengakibatkan server tiba-tiba ditutup. DMLs (Perbaikan Bug Komunitas #35625510)

Pembaruan mesin basis data	Versi kompatibel MySQL	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin database Aurora MySQL 2023-10-25 (versi 3.05.0) (Usang)	8.0.32	3.05.0	• Memperbaiki masalah yang dapat menyebabkan penggunaan CPU yang lebih tinggi karena rotasi sertifikat TLS di latar belakang (Perbaikan Bug Komunitas #34284186)
Pembaruan mesin database Aurora MySQL 2024-03-15 (versi 3.04.2, kompatibel dengan MySQL 8.0.28)	8.0.28	3.04.2	• Memperbaiki masalah di mana nilai baris cache dapat dihitung secara tidak benar, menyebabkan kegagalan selama restart database pada instance berbasis Graviton. (Perbaikan Bug Komunitas #35479763) • Berulang menjalankan rutinitas yang disimpan, memiliki sebagai subquery pernyataan SELECT yang berisi beberapa `AND`/`OR`,, atau `XOR` kondisi, menyebabkan konsumsi berlebihan dan mungkin akhirnya kelelahan memori virtual. (Perbaikan Bug Komunitas #33852530)

Pembaruan mesin basis data	Versi kompatibel MySQL	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin database Aurora MySQL 2023-11-13 (versi 3.04.1, kompatibel dengan MySQL 8.0.28)	8.0.28	3.04.1	• Memperbaiki masalah yang dapat menyebabkan penggunaan CPU yang lebih tinggi karena rotasi sertifikat TLS di latar belakang (Perbaikan Bug Komunitas #34284186)

Pembaruan mesin basis data	Versi kompatibel MySQL	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin database Aurora MySQL 2023-07-31 (versi 3.04.0), kompatibel dengan MySQL 8.0.28)	8.0.28	3.04.0	• Memperbaiki masalah di mana blok buffer yang berisi halaman tabel sementara intrinsik direlokasi selama penjelajahan halaman, sehingga menyebabkan kegagalan pernyataan (Bug# 33715694) • InnoDB: Mencegah operasi DDL online mengakses out-of-bounds memori (Bug # 34750489, Bug # 108925) • Memperbaiki masalah yang terkadang dapat menghasilkan hasil kueri yang salah saat memproses pernyataan SQL kompleks yang terdiri dari beberapa Ekspresi Tabel Umum bersarang (CTEs) (Bug # 34572040, Bug # 34634469, Bug # 33856374)
Pembaruan mesin database Aurora MySQL 2023-12-08 (versi 3.03.3) (Usang)	8.0.26	3.03.3	• Memperbaiki masalah yang dapat menyebabkan penggunaan CPU yang lebih tinggi karena rotasi sertifikat TLS di latar belakang (Perbaikan Bug Komunitas #34284186)

Pembaruan mesin basis data	Versi kompatibel MySQL	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin database Aurora MySQL 2023-08-29 (versi 3.03.2) (Usang)	8.0.26	3.03.2	• Memperbaiki masalah yang terkadang dapat menghasilkan hasil kueri yang salah saat memproses pernyataan SQL kompleks yang terdiri dari beberapa Ekspresi Tabel Umum bersarang (CTEs) (Bug #34572040, Bug #34634469, Bug #33856374) • InnoDB: Kondisi balapan (race condition) antara thread yang mencoba melepaskan dan memulai statistik untuk tabel yang sama menimbulkan kegagalan pernyataan (Bug #33135425) • InnoDB: Mencegah operasi DDL online mengakses out-of-bounds memori (Bug #34750489, Bug #108925)
Pembaruan mesin database Aurora MySQL 2023-05-11 (versi 3.03.1) (Usang)	8.0.26	3.03.1	• Memperbaiki masalah di mana blok buffer yang berisi halaman tabel sementara intrinsik direlokasi selama penjelajahan halaman, sehingga menyebabkan kegagalan pernyataan (Bug #33715694)

Pembaruan mesin basis data	Versi kompatibel MySQL	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin database Aurora MySQL 2023-03-01 (versi 3.03.0) (Usang)	8.0.26	3.03.0	• Memperbaiki masalah di mana beberapa jenis kolom, termasuk JSON dan TEXT, terkadang kehabisan buffer urutan jika ukurannya minimal tidak 15 kali lipat dari baris terbesar dalam urutan. Sekarang buffer urutan hanya perlu 15 kali lebih besar dari kunci urutan terbesar. (Bug #103325, Bug #105532, Bug #32738705, Bug #33501541) • Memperbaiki masalah di mana InnoDB tidak selalu menangani beberapa nama resmi untuk partisi tabel dengan benar. (Bug #23762382) • Memperbaiki masalah yang, dalam kondisi tertentu, dapat mengembalikan hasil yang salah karena perhitungan properti nullability yang tidak akurat saat menjalankan kueri dengan kondisi OR. (Bug #34060289) • Memperbaiki masalah yang, dalam kondisi tertentu, dapat mengembalikan hasil yang salah ketika dua kondisi berikut terpenuhi: • tabel turunan digabungkan ke dalam blok kueri luar.

Pembaruan mesin basis data	Versi kompatibel MySQL	Versi	Bug MySQL yang diperbaiki
			• kueri mencakup sambungan kiri dan subkueri IN. (Bug #34060289) • Memperbaiki masalah di mana nilai AUTO_INCREMENT yang salah dihasilkan ketika nilai kolom integer maksimum terlampaui. Kesalahan tersebut disebabkan nilai kolom maksimum yang tidak dipertimbangkan. Nilai AUTO_INCREMENT yang valid sebelumnya seharusnya dikembalikan dalam kasus ini, sehingga menyebabkan kesalahan kunci duplikat. (Bug #87926, Bug #26906787) • Memperbaiki masalah yang tidak memungkinkan pencabutan hak istimewa DROP pada Skema Kinerja. (Bug #33578113) • Memperbaiki masalah di mana prosedur tersimpan yang berisi pernyataan IF menggunakan EXISTS, yang bertindak pada satu atau beberapa tabel yang dihapus dan dibuat ulang di antara eksekusi, tidak dieksekusi dengan benar untuk invokasi

Pembaruan mesin basis data	Versi kompatibel MySQL	Versi	Bug MySQL yang diperbaiki
			berikutnya setelah invokasi pertama. (Bug #32855634). • Memperbaiki masalah di mana kueri yang mereferensikan tampilan di subkueri dan blok kueri luar dapat menyebabkan mulai ulang yang tidak terduga (Bug#32324234)

Pembaruan mesin basis data	Versi kompatibel MySQL	Versi	Bug MySQL yang diperbaiki

Pembaruan mesin database Aurora MySQL 2022-11-18 (versi 3.02.2) (Usang)	8.0.23	3.02.2	• Memperbaiki masalah yang, dalam kondisi tertentu, dapat mengembalikan hasil yang salah karena perhitungan properti nullability yang tidak akurat saat menjalankan kueri dengan kondisi OR. (Bug #34060289)
			• Memperbaiki masalah yang, dalam kondisi tertentu, dapat mengembalikan hasil yang salah ketika dua kondisi berikut terpenuhi:
			  • Tabel turunan digabungkan ke dalam blok kueri luar.
			  • Kueri mencakup sambungan kiri dan subkueri IN. (Bug #34060289)
			• Memperbaiki masalah yang tidak memungkinkan pencabutan hak istimewa DROP pada Skema Kinerja. (Bug #33578113)
			• Memperbaiki masalah di mana prosedur tersimpan yang berisi pernyataan IF menggunakan EXISTS, yang bertindak pada satu atau beberapa tabel yang dihapus dan dibuat ulang di antara eksekusi, tidak dieksekusi dengan benar untuk

Pembaruan mesin basis data	Versi kompatibel MySQL	Versi	Bug MySQL yang diperbaiki
			invokasi berikutnya setelah invokasi pertama. (MySQL Bug #32855634). • Nilai AUTO_INCREMENT yang salah dihasilkan ketika nilai kolom integer maksimum terlampaui. Kesalahan tersebut disebabkan nilai kolom maksimum yang tidak dipertimbangkan. Nilai AUTO_INCREMENT yang valid sebelumnya seharusnya dikembalikan dalam kasus ini, sehingga menyebabkan kesalahan kunci duplikat. (Bug #87926, Bug #26906787) • Memperbaiki masalah yang dapat menyebabkan kegagalan saat memutakhirkan kluster basis data Aurora MySQL versi 1 (Kompatibel dengan MySQL 5.6) yang berisi tabel buatan pengguna dengan tabel tertentu. IDs Penugasan tabel ini IDs dapat mengakibatkan tabel kamus data yang bertentangan IDs saat meningkatkan dari Aurora MySQL versi 2 (Kompatibel dengan MySQL 5.7) ke Aurora MySQL versi 3 (Kompatibel dengan MySQL 8.0). (Bug #33919635)

Pembaruan mesin basis data	Versi kompatibel MySQL	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin database Aurora MySQL 2022-04-20 (versi 3.02.0) (Usang)	8.0.23	3.02.0	Memperbaiki penanganan yang tidak tepat pada tabel sementara yang digunakan untuk kursor di dalam prosedur tersimpan yang dapat mengakibatkan perilaku server yang tidak terduga. (Bug #32416811)
Pembaruan mesin database Aurora MySQL 2022-04-15 (versi 3.01.1) (Usang)	8.0.23	3.01.1	Memperbaiki penanganan yang tidak tepat pada tabel sementara yang digunakan untuk kursor di dalam prosedur tersimpan yang dapat mengakibatkan perilaku server yang tidak terduga. (Bug #32416811)

Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 2.x

MySQL 5.7 versi kompatibel Aurora berisi semua perbaikan bug MySQL melalui MySQL 5.7.44. Tabel berikut mengidentifikasi bug MySQL tambahan yang telah diperbaiki oleh pembaruan mesin basis data Aurora MySQL, dan di pembaruan mana bug tersebut diperbaiki.

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin database Aurora MySQL 2024-07-09 (versi 2.12.3, kompatibel dengan MySQL 5.7.44) - Versi Dukungan Diperpanjang RDS	2.12.3	• Memperbaiki masalah saat tabel sementara terikat pada pemicu saat menjalankan pernyataan dapat menyebabkan restart mesin DB yang tidak terduga. • Memperbaiki cacat yang dapat menyebabkan server keluar ketika tabel tunggal `UPDATE` dan `DELETE` pernyataan menggunakan ekspresi yang diindeks dijalankan sebagai pernyataan yang disiapkan. (Bug #29257254)
Pembaruan mesin database Aurora MySQL 2023-12-28 (versi 2.12.1, kompatibel dengan MySQL 5.7.40) - Versi Dukungan Diperpanjang RDS	2.12.1	• Memperbaiki masalah yang dapat menyebabkan koneksi jarak jauh yang ada dan baru terhenti saat dijalankan bersamaan dengan pernyataan `SHOW PROCESSLIST` (Bug Komunitas #34857411) • Replikasi: Beberapa peristiwa log biner tidak selalu ditangani dengan benar (Bug #34617506) • Pemrosesan tetap token karakter tunggal oleh plugin pengurai Pencarian Teks Lengkap (FTS) (Bug #35432973)
Pembaruan mesin database Aurora MySQL 2023-07-25 (versi 2.12.0, kompatibel dengan MySQL 5.7.40) - Versi Dukungan Diperpanjang RDS	2.12.0	• Memperbaiki masalah yang dapat menyebabkan penggunaan CPU yang lebih tinggi karena rotasi sertifikat TLS di latar belakang. (Perbaikan Bug Komunitas #34284186)
Pembaruan mesin database Aurora MySQL 2023-10-17 (versi 2.11.4, kompatibel dengan MySQL 5.7.12) -	2.11.4	• Replikasi: Beberapa peristiwa log biner tidak selalu ditangani dengan benar. (Bug #34617506) • Memperbaiki masalah yang dapat menyebabkan penggunaan CPU yang lebih tinggi karena rotasi sertifikat TLS di latar belakang. (Perbaikan Bug Komunitas #34284186)

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
Versi Dukungan Diperpanjang RDS		• Dalam pernyataan yang disiapkan, beberapa jenis subkueri dapat menyebabkan keluar server. (Bug #33100586)

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin database Aurora MySQL 2022-10-25 (versi 2.11.0, kompatibel dengan MySQL 5.7.12) - Versi Dukungan Diperpanjang RDS	2.11.0	• Memperbaiki masalah di mana kode untuk membaca informasi kumpulan karakter dari tabel peristiwa pernyataan Skema Kinerja (misalnya, events_statements_current) tidak mencegah penulisan serentak ke informasi kumpulan karakter tersebut. Akibatnya, kumpulan karakter teks kueri SQL bisa menjadi tidak valid, yang dapat mengakibatkan server keluar. Dengan perbaikan ini, kumpulan karakter yang tidak valid menyebabkan pemotongan kolom SQL_TEXT dan mencegah server keluar. (Bug #23540008) • InnoDB: Backport perbaikan untuk Community Bug #25189192, Bug #84038. Memperbaiki masalah di mana operasi RENAME TABLE yang memindahkan tabel ke skema yang berbeda, InnoDB gagal memperbarui tabel kamus data INNODB_SYS_DATAFILES. Hal ini mengakibatkan kesalahan saat mulai ulang yang menunjukkan bahwa file data ruang tabel tidak dapat ditemukan. • InnoDB: Memperbaiki masalah di mana server menghapus indeks kunci asing yang didefinisikan secara internal ketika menambahkan indeks baru dan mencoba menggunakan indeks sekunder yang ditentukan pada kolom yang dihasilkan secara virtual sebagai indeks kunci asing, sehingga menyebabkan server keluar. InnoDB sekarang mengizinkan batasan kunci asing untuk mereferensikan indeks sekunder yang didefinisikan pada kolom yang dihasilkan secara virtual. (Bug #23533396) • Memperbaiki masalah di mana dua sesi secara bersamaan mengeksekusi operasi INSERT... ON DUPLICATE KEY UPDATE menghasilkan deadlock. Selama rollback sebagian tuple, sesi lain dapat memperbaruinya. Perbaikan untuk bug ini mengembalikan perbaikan untuk Bug #11758237, Bug #17604730, dan Bug #20040791. (Bug #25966845) • Memperbaiki masalah di mana hak istimewa EXECUTE dan ALTER RUTINE tidak diberikan dengan benar kepada

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
		pembuat rutin bahkan dengan automatic_sp_privileges diaktifkan. (Bug #27407480) • Backport perbaikan untuk Bug Komunitas #24671968: Memperbaiki masalah di mana kueri dapat memberikan hasil yang salah jika klausa WHERE berisi subkueri dependen, tabel memiliki indeks sekunder pada kolom dalam daftar tertentu yang diikuti oleh kolom dalam subkueri, dan `GROUP BY` atau `DISTINCT` mengizinkan kueri untuk menggunakan Pemindaian Indeks Longgar. • Memperbaiki masalah di mana replikasi rusak jika pernyataan penghapusan multi-tabel dikeluarkan terhadap beberapa tabel dengan kunci asing. (Bug #80821) • Memperbaiki masalah di mana dalam kasus khusus, kesalahan slave tertentu tidak diabaikan bahkan dengan slave_skip_errors diaktifkan. Dalam kasus ketika pembukaan dan penguncian tabel gagal atau ketika konversi bidang gagal pada server yang menjalankan replikasi berbasis baris, kesalahan tersebut dianggap kritis dan status slave_skip_errors diabaikan. Perbaikan ini memastikan bahwa dengan slave_skip_errors diaktifkan, semua kesalahan yang dilaporkan selama penerapan transaksi ditangani dengan benar. (Bug #70640, Bug #17653275) • Memperbaiki masalah di mana pernyataan `SET PASSWORD` direplikasi dari master MySQL 5.6 ke slave MySQL 5.7, atau dari master MySQL 5.7 dengan variabel sistem log_built in_as_identified_by_password diatur ke ON ke slave MySQL 5.7, hash kata sandi itu sendiri juga di-hash sebelum disimpan di slave. Masalah tersebut telah diperbaiki dan hash kata sandi yang direplikasi disimpan seperti aslinya diteruskan ke slave. (Bug#24687073) • Memperbaiki masalah di mana serialisasi nilai JSON yang terdiri dari sub-dokumen besar yang dibungkus dalam banyak tingkatan array JSON, objek, atau keduanya, terkadang

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
		membutuhkan waktu yang terlalu lama untuk diselesaikan. (Bug #23031146) • Pernyataan yang tidak dapat diuraikan (karena, misalnya, kesalahan sintaks) tidak lagi ditulis ke log kueri lambat. (Bug #33732907)
Pembaruan mesin basis data Aurora MySQL 2022-11-01 (versi 2.10.3) (Dihentikan)	2.10.3	• Memperbaiki masalah di mana kode untuk membaca informasi kumpulan karakter dari tabel peristiwa pernyataan Skema Kinerja (misalnya, events_statements_current) tidak mencegah penulisan serentak ke informasi kumpulan karakter tersebut. Akibatnya, kumpulan karakter teks kueri SQL bisa menjadi tidak valid, yang dapat mengakibatkan server keluar. Dengan perbaikan ini, kumpulan karakter yang tidak valid menyebabkan pemotongan kolom SQL_TEXT dan mencegah server keluar. (Bug #23540008) • Memperbaiki masalah ketika UPDATE memerlukan tabel sementara yang memiliki kunci primer lebih besar dari 1024 byte dan tabel tersebut dibuat menggunakan InnoDB, server dapat keluar. (Bug #25153670

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin basis data Aurora MySQL 2022-01-26 (versi 2.10.2) (Dihentikan)	2.10.2	• Memperbaiki masalah di InnoDB di mana kesalahan dalam kode yang terkait dengan statistik tabel menimbulkan pernyataan dalam file sumber dict0stats.cc (http://dict0stats.cc/). (Bug #24585978) • Indeks sekunder di atas kolom virtual menjadi rusak saat indeks tersebut dibuat secara online. Untuk pernyataan UPDATE (https://dev.mysql.com/doc/refman/5.7/en/update.html), kami memperbaikinya sebagai berikut: Jika nilai kolom virtual dari catatan indeks diatur ke NULL, maka kami menghasilkan nilai ini dari catatan indeks cluster. (Bug #30556595) • ASSERTION "!OTHER_LOCK" IN LOCK_REC_ADD_TO_QUEUE (Bug #29195848) • HANDLE_FATAL_SIGNAL (SIG=11) DI __STRCHR_ (Bug #28653104) SSE2 • Memperbaiki masalah di mana gangguan kueri selama waktu tunggu kunci dapat menyebabkan kesalahan di InnoDB. (Bug #28068293) • Transaksi yang berseling terkadang dapat menyebabkan deadlock pada pengaplikasi replika saat tingkat isolasi transaksi diatur ke REPEATABLE READ. (Bug #25040331) • Memperbaiki masalah yang dapat menyebabkan replika binlog macet karena batas waktu tunggu kunci.(Bug #27189701)
Pembaruan mesin basis data Aurora MySQL 2021-10-21 (versi 2.10.1) (Dihentikan)	2.10.1	CURRENT_TIMESTAMP MENGHASILKAN NOL PADA PEMICU. (Bug #25209512)

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin basis data Aurora MySQL 2021-05-25 (versi 2.10.0) (Dihentikan)	2.10.0	• Transaksi yang berseling terkadang dapat menyebabkan deadlock pada pengaplikasi replika saat tingkat isolasi transaksi diatur ke REPEATABLE READ. (Bug #25040331) • Ketika prosedur tersimpan berisi pernyataan yang merujuk ke suatu tampilan yang pada gilirannya merujuk ke tampilan lain, prosedur tidak dapat berhasil diinvokasi lebih dari sekali. (Bug #87858, Bug #26864199) • Untuk kueri dengan banyak kondisi OR, pengoptimal sekarang lebih hemat memori dan lebih kecil kemungkinannya untuk melebihi batas memori yang diberlakukan oleh variabel sistem range_optimizer_max_mem_size. Selain itu, nilai default untuk variabel tersebut telah dinaikkan dari 1.536.000 menjadi 8.388.608. (Bug #79450, Bug #22283790) • Replikasi: Di fungsi `next_event()`, yang dipanggil oleh thread SQL replika untuk membaca peristiwa berikutnya dari log relay, thread SQL tersebut tidak melepaskan `relaylog.log_lock` yang diperolehnya saat mengalami kesalahan (misalnya, karena log relay tertutup), sehingga menyebabkan hang pada semua thread lain yang menunggu untuk mendapatkan kunci pada log relay. Dengan perbaikan ini, kunci dilepaskan sebelum thread SQL meninggalkan fungsi dalam situasi tersebut. (Bug #21697821) • Memperbaiki kerusakan memori untuk `ALTER TABLE` dengan kolom virtual. (Bug #24961167; Bug #24960450) • Replikasi: Replika multithread tidak dapat dikonfigurasi dengan ukuran antrean kecil menggunakan slave_pending_jobs_size_max jika diperlukan untuk memproses transaksi yang lebih besar dari ukuran tersebut. Setiap paket yang lebih besar daripada slave_pending_jobs_size_max ditolak dengan kesalahan `ER_MTS_EVENT_BIGGER_PENDING_JOBS_SIZE_MAX`, bahkan jika paket lebih kecil dari batas yang ditetapkan oleh slave_max_allowed_

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
		packet. Dengan perbaikan ini, slave_pending_jobs_size_max menjadi batas lembut, bukan batas keras. Jika ukuran paket melebihi slave_pending_jobs_size_max tapi kurang dari slave_max_allowed_packet, transaksi ditangguhkan sampai semua pekerja replika memiliki antrean kosong, dan kemudian diproses. Semua transaksi berikutnya ditangguhkan sampai transaksi besar selesai. Oleh karena itu, ukuran antrean untuk pekerja replika dapat dibatasi sambil tetap memungkinkan transaksi sesekali yang lebih besar. (Bug #21280753, Bug #77406) • Replikasi: Ketika menggunakan replika multithread, kesalahan applier menampilkan data ID pekerja yang tidak konsisten dengan data yang dieksternalkan dalam tabel replikasi Skema Kinerja. (Bug #25231367) • Replikasi: Pada replika replikasi berbasis GTID yang berjalan dengan -GTID-mode=on, -log-bin=off, dan menggunakan -, ketika kesalahan ditemukan yang harus diabaikan tidak diperbarui dengan benar, menyebabkan slave-skip-errors sinkronisasi yang longgar dengan. `Exec_Master_Log_Pos Exec_Master_Log_Pos Read_master_log_pos` Jika `GTID_NEXT` tidak ditentukan, replika tidak akan pernah memperbarui status GTID ketika dibatalkan dari transaksi pernyataan tunggal. `Exec_Master_Log_Pos` tidak akan diperbarui karena meskipun transaksi selesai, status GTID akan menunjukkan sebaliknya. Perbaikan ini menghilangkan batasan pembaruan status GTID saat transaksi dibatalkan hanya jika `GTID_NEXT` ditentukan. (Bug #22268777) • Replikasi: Pernyataan yang gagal sebagian tidak menggunakan secara benar GTID yang ditentukan atau dibuat secara otomatis saat pencatatan log biner dinonaktifkan. Perbaikan ini memastikan bahwa DROP TABLE yang gagal sebagian, DROP USER yang gagal sebagian, atau DROP VIEW yang gagal sebagian menggunakan GTID masing-masing yang

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
		relevan dan menyimpannya ke dalam tabel `@@GLOBAL.GTID_EXECUTED` dan `mysql.gtid_executed` ketika pencatatan log biner dinonaktifkan. (Bug #21686749) • Replikasi: Replika yang menjalankan MySQL 5.7 tidak dapat terhubung ke sumber MySQL 5.5 karena kesalahan saat mengambil server_uuid, yang bukan bagian dari MySQL 5.5. Hal ini disebabkan oleh perubahan pada metode pengambilan `server_uuid` . (Bug #22748612) • Replikasi Binlog: Mekanisme melewatkan transaksi GTID tidak berfungsi dengan baik untuk transaksi XA sebelum perbaikan ini. Server memiliki mekanisme untuk melewati (secara diam-diam) transaksi GTID jika transaksi tersebut sudah dijalankan sebelumnya. (BUG#25041920) • Pernyataan XA ROLLBACK yang gagal karena ID transaksi yang diberikan salah, dapat direkam dalam log biner dengan ID transaksi yang benar, oleh karena itu dapat ditindaklanjuti dengan replika replikasi. Pemeriksaan sekarang dilakukan untuk situasi kesalahan sebelum pencatatan log biner terjadi, dan pernyataan `ROLLBACK` XA yang gagal tidak dicatat. (Bug #26618925) • Replikasi: Jika replika disiapkan menggunakan pernyataan CHANGE MASTER TO yang tidak menentukan nama file log sumber dan posisi log sumber, lalu matikan sebelum START SLAVE dikeluarkan, lalu restart dengan opsi - relay-log-recovery set, replikasi tidak dimulai. Hal ini terjadi karena thread penerima belum dimulai sebelum pemulihan log relay diupayakan, jadi tidak ada peristiwa rotasi log yang tersedia di log relay untuk memberikan nama file log sumber dan posisi log sumber. Dalam situasi ini, replika sekarang melompati pemulihan log relay dan mencatat log peringatan, lalu melanjutkan untuk memulai replikasi. (Bug #28996606, Bug #93397)

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
		• Replikasi: Dalam replikasi berbasis baris, sebuah pesan yang salah menampilkan panjang bidang dikembalikan saat mereplikasi dari tabel dengan kolom `utf8mb3` ke tabel definisi yang sama di mana kolom didefinisikan dengan kumpulan karakter `utf8mb4`. (Bug #25135304, Bug #83918) • Replikasi: Ketika pernyataan RESET SLAVE dikeluarkan pada replika replikasi yang sedang digunakan, file log relai yang ada dibersihkan, tetapi file log relai baru pengganti dibuat sebelum kumpulan yang diterima GTIDs untuk saluran telah dihapus. GTIDs Oleh karena itu, set GTID sebelumny a ditulis ke file log relai baru sebagai `PREVIOUS_GTIDS` acara, menyebabkan kesalahan fatal dalam replikasi yang menyatakan bahwa replika memiliki GTIDs lebih dari sumber, meskipun set gtid_execute untuk kedua server kosong. Sekarang, ketika `RESET SLAVE` dikeluarkan, set yang GTIDs diterima dihapus sebelum file log relai baru dihasilkan, sehingga situasi ini tidak terjadi. (Bug #27411175) • Replikasi: GTIDs Dengan digunakan untuk replikasi, transaksi termasuk pernyataan yang menyebabkan kesalahan penguraian (ER_PARSE_ERROR) tidak dapat dilewati secara manual dengan metode yang disarankan untuk menyuntik kan transaksi kosong atau penggantian dengan GTID yang sama. Tindakan ini seharusnya menghasilkan replika yang mengidentifikasi GTID seperti yang sudah digunakan, dan oleh karena itu melewatkan transaksi yang tidak diinginkan yang membagikan GTID-nya. Namun, dalam kasus kesalahan penguraian, karena pernyataan diuraikan sebelum GTID diperiksa untuk melihat apakah perlu dilewati, thread applier replikasi berhenti karena kesalahan penguraian, meskipun tujuannya adalah agar transaksi dilewati juga. Dengan perbaikan ini, thread applier replikasi sekarang mengabaik an kesalahan penguraian jika transaksi yang bersangku tan perlu dilewati karena GTID sudah digunakan. Perhatika

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki

		n bahwa perubahan perilaku ini tidak berlaku dalam kasus beban kerja yang terdiri dari output log biner yang dihasilkan oleh `mysqlbinlog` . Dalam situasi tersebut, akan ada risiko bahwa transaksi dengan kesalahan penguraian yang segera mengikuti transaksi yang dilewati juga akan secara diam-diam dilewati, ketika transaksi tersebut akan meningkatkan kesalahan. (Bug #27638268)
		• Replikasi: Mengaktifkan thread SQL untuk GTID melewatkan transaksi parsial. (Bug #25800025)
		• Replikasi: Ketika parameter batas waktu negatif atau pecahan diberikan ke `WAIT_UNTIL_SQL_THREAD_AFTER_GTIDS()` , server berperilaku dengan cara yang tidak terduga. Dengan perbaikan ini:
		  • Nilai batas waktu pecahan dibaca apa adanya, tanpa pembulatan.
		  • Nilai batas waktu negatif ditolak dengan kesalahan jika server menggunakan mode SQL yang ketat; jika server tidak dalam mode SQL yang ketat, nilainya membuat fungsi segera mengembalikan `NULL` tanpa menunggu apa pun dan kemudian mengeluarkan peringatan. (Bug #24976304, Bug #83537)
		• Replikasi: Jika fungsi `WAIT_FOR_EXECUTED_GTID_SET()` digunakan dengan nilai batas waktu termasuk bagian pecahan (misalnya, 1,5), kesalahan dalam logika casting berarti batas waktu dibulatkan ke bawah ke seluruh detik terdekat, dan ke nol untuk nilai kurang dari 1 detik (misalnya , 0,1). Logika casting sekarang telah diperbaiki sehingga nilai batas waktu diterapkan seperti yang ditentukan semula tanpa pembulatan. Terima kasih kepada Dirkjan Bussink atas kontribusinya. (Bug #29324564, Bug #94247)
		• Dengan GTIDs diaktifkan, XA COMMIT pada transaksi XA yang terputus dalam transaksi multi-pernyataan menimbulkan pernyataan. (Bug #22173903)

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
		• Replikasi: Sebuah pernyataan diangkat dalam build debug jika pernyataan XA ROLLBACK dikeluarkan untuk pengenal transaksi yang tidak diketahui ketika nilai gtid_next telah ditetapkan secara manual. Server sekarang tidak mencoba memperbarui status GTID jika pernyataan `ROLLBACK` XA gagal dengan kesalahan. (Bug #27928837, Bug #90640) • Memperbaiki masalah urutan penyortiran yang salah ketika beberapa fungsi `CASE` yang digunakan dalam klausa `ORDER BY`. (Bug#22810883) • Beberapa kueri yang menggunakan pengurutan dapat mengakses kolom yang tidak diinisialisasi selama pengoptim alan dan menyebabkan server keluar. (Bug #27389294)
Pembaruan mesin basis data Aurora MySQL 2021-11-12 (versi 2.09.3) (Dihentikan)	2.09.3	• ASSERTION !M_PREBUILT->TRX->CHECK_FOREIGNS. (Bug #23533396) • Replikasi:* Masalah penguncian di fungsi WAIT_FOR_ EXECUTED_GTID_SET() dapat menyebabkan server macet dalam keadaan tertentu. Masalah tersebut telah diperbaiki. (Bug #29550513)

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin basis data Aurora MySQL 2020-12-11 (versi 2.09.1) (Dihentikan)	2.09.1	• Replikasi: Transaksi yang berseling terkadang dapat menyebabkan deadlock pada pengaplikasi slave saat tingkat isolasi transaksi diatur ke REPEATABLE READ. (Bug #25040331) • Untuk tabel dengan kolom TIMESTAMP atau DATETIME yang memiliki nilai default CURRENT_TIMESTAMP, kolom tersebut dapat diinisialisasi ke `0000-00-00 00:00:00` jika tabel memiliki pemicu `BEFORE INSERT`. (Bug #25209512, Bug #84077) • Untuk pernyataan INSERT yang menerima pembuatan nilai oleh daftar `VALUES` untuk baris kedua atau setelahny a menggunakan subkueri yang berisi sambungan, server dapat keluar setelah gagal menyelesaikan hak istimewa yang diperlukan. (Bug #23762382)
Pembaruan mesin basis data Aurora MySQL 2020-11-12 (versi 2.08.3) (Dihentikan)	2.08.3	• Bug #23762382 - PENYISIPAN KUERI NILAI DENGAN SAMBUNGAN DALAM PILIHAN MENYEBABKAN PERILAKU YANG SALAH. • Bug #25209512 - CURRENT_TIMESTAMP MENGHASILKAN NOL SAAT DIPICU.

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin basis data Aurora MySQL 2020-06-02 (versi 2.08.0) (Dihentikan)	2.08.0	• Bug #25289359: Kunci cache teks lengkap yang diambil saat data disinkronkan tidak dilepaskan jika ukuran cache teks lengkap melebihi batas ukuran cache teks lengkap. • Bug #29138644: Mengubah waktu sistem secara manual saat server MySQL sedang berjalan menyebabkan penundaan thread pembersih halaman. • Bug #25222337: Nama bidang kolom virtual NULL dalam indeks virtual menyebabkan server keluar selama perbandingan nama bidang yang terjadi saat pengisian kolom virtual yang dipengaruhi oleh batasan kunci asing. • Bug #25053286: Menjalankan prosedur tersimpan dengan kueri yang mengakses suatu tampilan dapat mengalokasikan memori yang tidak dibebaskan hingga sesi berakhir. • Bug #25586773: Menjalankan prosedur tersimpan dengan pernyataan yang menciptakan tabel dari konten pernyataan SELECT tertentu dapat menyebabkan kebocoran memori. • Bug #28834208: Selama penerapan log, setelah operasi OPTIMIZE TABLE, InnoDB tidak mengisi kolom virtual sebelum memeriksa pembaruan indeks kolom virtual. • Bug #26666274: Infinite loop dalam kontainer buffer skema kinerja karena overflow integer 32-bit yang tidak ditandai.
Pembaruan mesin basis data Aurora MySQL 2022-06-16 (versi 2.07.8) (Dihentikan)	2.07.8	Ketika UPDATE memerlukan tabel sementara yang memiliki kunci primer lebih besar dari 1024 byte dan tabel tersebut dibuat menggunakan InnoDB, server dapat keluar. (Bug #25153670)

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin basis data Aurora MySQL 2021-09-02 (versi 2.07.6) (Dihentikan)	2.07.6	• DIPERLUKAN TERLALU BANYAK WAKTU UNTUK MENYISIPKAN CATATAN UKURAN 644K. (Bug#23031146)
Pembaruan mesin basis data Aurora MySQL 2021-03-04 (versi 2.07.4) (Dihentikan)	2.07.4	• Memperbaiki masalah dalam pengurai ngram Teks lengkap saat menangani token yang berisi ' ' (spasi), '%', atau ','. Pelanggan harus membangun kembali indeks FTS mereka jika menggunakan pengurai ngram. (Bug #25873310) • Memperbaiki masalah yang dapat menyebabkan mulai ulang mesin selama eksekusi kueri dengan tampilan SQL bersarang . (Bug #27214153, Bug #26864199)

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin basis data Aurora MySQL 2020-11-10 (versi 2.07.3) (Dihentikan)	2.07.3	• InnoDB: Transaksi XA bersamaan yang berhasil berjalan ke tahap persiapan XA pada master bertentangan saat diputar ulang pada slave, sehingga tunggu kunci mengalami waktu habis di thread pengaplikasi. Pertentangan tersebut disebabkan oleh rentang kunci GAP yang berbeda ketika transaksi diputar ulang secara seri pada slave. Untuk mencegah jenis pertentangan ini, kunci GAP yang diambil oleh transaksi XA pada tingkat isolasi READ COMMITTED sekarang dilepaskan (dan sudah tidak diwariskan) saat transaksi XA mencapai tahap persiapan. (Bug #27189701, Bug #25866046) • InnoDB: Kunci gap diambil secara tidak perlu selama validasi kunci asing saat menggunakan tingkat isolasi READ COMMITTED. (Bug #25082593) • Replikasi: Saat menggunakan transaksi XA, jika tunggu kunci mengalami waktu habis atau deadlock terjadi untuk thread pengaplikasi (SQL) pada slave replikasi, coba ulang otomatis tidak berfungsi. Penyebabnya adalah meskipun thread SQL akan melakukan rollback, thread tersebut tidak akan melakukan rollback transaksi XA. Ini berarti ketika transaksi dicoba kembali, peristiwa pertama adalah XA START yang tidak valid karena transaksi XA sudah berlangsung, yang menyebabkan kesalahan XAER_RMFAIL. (Bug #24764800) • Replikasi: Transaksi yang berseling terkadang dapat menyebabkan deadlock pada pengaplikasi slave saat tingkat isolasi transaksi diatur ke REPEATABLE READ. (Bug #25040331) • Replikasi: Nilai dikembalikan oleh pernyataan SHOW SLAVE STATUS untuk total ukuran gabungan dari semua file log relay yang ada (Relay_Log_Space) dapat menjadi jauh lebih besar daripada ruang disk yang digunakan oleh file log relay tersebut. I/O Utas tidak mengunci variabel saat memperbarui nilainya, sehingga utas SQL dapat secara otomatis menghapus file log relai dan menulis nilai yang dikurangi

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
		sebelum I/O utas selesai memperbarui nilainya. Thread I/O kemudian menulis perhitungan ukuran aslinya, dengan mengabaikan pembaruan thread SQL dan oleh karena itu menambahkan kembali ruang untuk file yang dihapus. Nilai Relay_Log_Space sekarang dikunci selama pembaruan untuk mencegah pembaruan bersamaan dan memastikan perhitungan yang akurat. (Bug #26997096, Bug #87832) • Untuk pernyataan INSERT yang menerima pembuatan nilai oleh daftar VALUES untuk baris kedua atau setelahnya menggunakan subkueri yang berisi sambungan, server dapat keluar setelah gagal menyelesaikan hak istimewa yang diperlukan. (Bug #23762382) • Untuk tabel dengan kolom TIMESTAMP atau DATETIME yang memiliki nilai default CURRENT_TIMESTAMP, kolom tersebut dapat diinisialisasi ke `0000-00-00 00:00:00` jika tabel memiliki pemicu `BEFORE INSERT`. (Bug #25209512, Bug #84077) • Keluarnya server dapat disebabkan oleh upaya serentak beberapa thread yang mendaftarkan dan membatalkan pendaftaran objek Skema Kinerja metadata. (Bug #26502135) • Menjalankan prosedur tersimpan yang berisi pernyataan yang menciptakan tabel dari konten pernyataan SELECT tertentu dapat menyebabkan kebocoran memori. (Bug #25586773) • Menjalankan prosedur tersimpan dengan kueri yang mengakses suatu tampilan dapat mengalokasikan memori yang tidak dibebaskan hingga sesi berakhir. (Bug #25053286) • Kasus materialisasi subkueri tertentu dapat menyebabkan keluarnya server. Kueri ini sekarang menghasilkan pesan kesalahan yang menunjukkan bahwa materialisasi dinonaktifkan. (Bug #26402045) • Kueri dengan banyak sambungan tersisa akan lambat jika buffering sambungan digunakan (misalnya, menggunak

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
		an algoritma loop bersarang blok). (Bug #18898433, Bug #72854) • Pengoptimal melewatkan kolom kedua dalam indeks komposit saat mengeksekusi sambungan bagian dalam dengan klausa `LIKE` di kolom kedua. (Bug #26402045)
Pembaruan mesin basis data Aurora MySQL 2020-04-17 (versi 2.07.2) (Dihentikan)	2.07.2	• Bug #23104498: Memperbaiki masalah dalam Skema Kinerja dalam melaporkan total memori yang digunakan. (https://github.com/mysql/mysql-server/commit/20b6840df5452f47313c6f9a6ca075bfbc00a96b) • Bug #22551677: Memperbaiki masalah di Skema Kinerja yang dapat menyebabkan kerusakan mesin basis data saat mencoba untuk melakukannya secara offline.(https://github.com/mysql/mysql-server/commit/05e2386eccd32b6b444b900c9f8a87a1d8d531e9) • Bug #23550835, Bug #23298025, Bug #81464: Memperbaiki masalah di Skema Kinerja yang menyebabkan crash pada mesin basis data karena melebihi kapasitas buffer internal. (https://github.com/mysql/mysql-server/commit/b4287f93857bf2f99b18fd06f555bbe5b12debfc, https://github.com/mysql/mysql-server/commit/b4287f93857bf2f99b18fd06f555bbe5b12debfc)

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin basis data Aurora MySQL 2019-11-25 (versi 2.07.0) (Dihentikan)	2.07.0	• Bug #26251621: PERILAKU YANG SALAH DENGAN TRIGGER DAN GCOL • Bug #22574695: ASSERTION `!TABLE || (!TABLE-> READ_SET || BITMAP_IS_SET(TABLE->READ_SET, FIEL • Bug #25966845: INSERT PADA KUNCI DUPLIKAT MENGHASILKAN DEADLOCK • Bug #23070734: TRUNCATE TABLES BERSAMAAN MENYEBABKAN MACET • Bug #26191879: FOREIGN KEY CASCADES MENGGUNAKAN MEMORI BERLEBIHAN • Bug #20989615: INNODB AUTO_INCREMENT MENGHASILKAN NILAI YANG SAMA DUA KALI
Pembaruan mesin basis data Aurora MySQL 2019-11-11 (versi 2.05.0) (Dihentikan)	2.05.0	• Bug #23054591: PURGE BINARY LOGS TO sedang membaca seluruh file binlog dan menyebabkan macet MySql

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin basis data Aurora MySQL 2020-08-14 (versi 2.04.9) (Dihentikan)	2.04.9	• Bug #23070734, Bug #80060: Pemotongan Bersamaan menyebabkan macet TABLEs • Bug #23103937: PS_TRUNCATE_ALL_TABLES() TIDAK DAPAT DIGUNAKAN DALAM MODE SUPER_READ_ONLY • Bug#22551677: Saat membuat server offline, kondisi balapan dalam Skema Kinerja dapat menyebabkan server keluar. • Bug #27082268: Sinkronisasi sync FTS tidak valid. • BUG #12589870: Memperbaiki masalah yang menyebabkan mulai ulang dengan pernyataan multikueri saat cache kueri diaktifkan. • Bug #26402045: Kasus materialisasi subkueri tertentu dapat menyebabkan keluarnya server. Kueri ini sekarang menghasilkan pesan kesalahan yang menunjukkan bahwa materialisasi dinonaktifkan. • Bug #18898433: Kueri dengan banyak sambungan tersisa akan lambat jika buffering sambungan digunakan (misalnya, menggunakan algoritma loop bersarang blok). • Bug #25222337: Nama bidang kolom virtual NULL dalam indeks virtual menyebabkan server keluar selama perbandingan nama bidang yang terjadi saat pengisian kolom virtual yang dipengaruhi oleh batasan kunci asing. (https://github.com/mysql/mysql-server/commit/273d5c9d7072c63b6c47dbef6963d7dc491d5131) • Bug #25053286: Menjalankan prosedur tersimpan dengan kueri yang mengakses suatu tampilan dapat mengalokasikan memori yang tidak dibebaskan hingga sesi berakhir. (https://github.com/mysql/mysql-server/commit/d7b37d4d141a95f577916448650c429f0d6e193d) • Bug #25586773: Menjalankan prosedur tersimpan yang berisi pernyataan yang membuat tabel dari isi pernyataan SELECT (https://dev.mysql.com/doc/refman/5.7/en/select.html) tertentu dapat mengakibatkan kebocoran memori. (https://github.co

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
		m/mysql/mysql-server/commit/88301e5adab65f6750f66af284be410c4369d0c1) • Bug #26666274: LOOP TANPA HENTI DI KONTAINER BUFFER SKEMA KINERJA. • Bug #23550835, Bug #23298025, Bug #81464: Tabel Skema Kinerja SELECT ketika buffer internal penuh dapat menyebabkan server keluar.
Pembaruan mesin basis data Aurora MySQL 2019-09-19 (versi 2.04.6) (Dihentikan)	2.04.6	• Bug #23054591: PURGE BINARY LOGS TO sedang membaca seluruh file binlog dan menyebabkan macet MySql
Pembaruan mesin basis data Aurora MySQL 2019-05-02 (versi 2.04.2) (Dihentikan)	2.04.2	Bug #24829050 - OPTIMALISASI INDEX_MERGE_INTERSECTION MENYEBABKAN HASIL KUERI YANG SALAH
Pembaruan mesin basis data Aurora MySQL 2018-10-11 (versi 2.03) (Dihentikan)	2.03	• PINDAI BALIK DI TABEL TERPARTISI MENJALANKAN PERINTAH ICP BERDASARKAN DESC (Bug #24929748). • JSON_OBJECT MEMBUAT KODE JSON TIDAK VALID (Bug#26867509). • MEMASUKKAN DATA JSON BESAR MEMBUTUHKAN WAKTU SANGAT LAMA (Bug #22843444). • TABEL PARTISI MENGGUNAKAN LEBIH BANYAK MEMORI DALAM 5.7 DARIPADA 5.6 (Bug #25080442).

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin basis data Aurora MySQL 2018-09-21 (versi 2.02.4) (Dihentikan)	2.02.4	• BUG#13651665 INNODB MAY BE UNABLE TO LOAD TABLE DEFINITION AFTER RENAME • BUG#21371070 INNODB: CANNOT ALLOCATE 0 BYTES. • BUG#21378944 FTS ASSERT ENC.SRC_ILIST_PTR != NULL, FTS_OPTIMIZE_WORD(), OPTIMIZE TABLE • BUG#21508537 ASSERTION FAILURE UT_A(!VICTIM_TRX->READ_ONLY) • BUG#21983865 UNEXPECTED DEADLOCK WITH INNODB_AUTOINC_LOCK_MODE=0 • BUG#22679185 INVALID INNODB FTS DOC ID DURING INSERT • BUG#22899305 GCOLS: ASSERTION: !(COL->PRTYPE & 256). • BUG#22956469 MEMORY LEAK INTRODUCED IN 5.7.8 IN MEMORY/INNODB/OS0FILE • BUG#22996488 CRASH IN FTS_SYNC_INDEX WHEN DOING DDL IN A LOOP • BUG#23014521 GCOL:INNODB: ASSERTION: !IS_V • BUG#23021168 REPLICATION STOPS AFTER TRX IS ROLLED BACK ASYNC • BUG#23052231 ASSERTION: ADD_AUTOINC < DICT_TABLE_GET_N_USER_COLS • BUG#23149683 ROTATE INNODB MASTER KEY WITH KEYRING_OKV_CONF_DIR MISSING: SIGSEGV; SIGNAL 11 • BUG#23762382 INSERT VALUES QUERY WITH JOIN IN A SELECT CAUSES INCORRECT BEHAVIOR • BUG#25209512 CURRENT_TIMESTAMP PRODUCES ZEROS IN TRIGGER

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
		• BUG#26626277 BUG IN "INSERT... ON DUPLICATE KEY UPDATE" QUERY • BUG#26734162 INCORRECT BEHAVIOR WITH INSERT OF BLOB + ON DUPLICATE KEY UPDATE • BUG#27460607 INCORRECT WHEN INSERT SELECT's SOURCE TABLE IS EMPTY
Pembaruan mesin basis data Aurora MySQL 2018-05-03 (versi 2.02) (Dihentikan)	2.02.0	Sambungan kiri menampilkan hasil yang salah pada sisi luar (Bug #22833364)

Bug MySQL yang diperbaiki oleh pembaruan mesin basis data Aurora MySQL 1.x

Versi Aurora yang kompatibel dengan MySQL 5.6 berisi semua perbaikan bug MySQL melalui MySQL 5.6.10. Tabel berikut mengidentifikasi bug MySQL tambahan yang telah diperbaiki oleh pembaruan mesin basis data Aurora MySQL, dan di pembaruan mana bug tersebut diperbaiki.

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin basis data Aurora MySQL 2021-03-18 (versi 1.23.2) (Dihentikan)	1.23.2	• Replikasi: Selama pernyataan SHOW BINLOG EVENTS dijalankan, setiap transaksi paralel diblokir. Perbaikan ini memastikan proses SHOW BINLOG EVENTS sekarang hanya memperoleh sebuah kunci selama menghitung posisi akhir file, oleh karena itu transaksi paralel tidak diblokir dalam jangka waktu lama. (Bug #76618, Bug #20928790)
Pembaruan mesin basis data Aurora MySQL 2020-09-0	1.23.0	• Peristiwa binlog dengan ALTER TABLE ADD COLUMN ALGORITHM=QUICK akan ditulis ulang sebagai ALGORITHM =DEFAULT agar kompatibel dengan edisi komunitas.

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
2 (versi 1.23.0) (Dihentikan)		• BUG #22350047: JIKA KLIEN DIMATIKAN SETELAH ROLLBACK KE SAVEPOINT STMTS SEBELUMNYA DITERAPKAN • Bug #29915479: MENJALANKAN COM_REGISTER_SLAVE TANPA COM_BINLOG_DUMP DAPAT BERDAMPAK PADA KELUAR SERVER • Bug #30441969: BUG #29723340: MYSQL SERVER CRASH SETELAH KUERI SQL DENGAN ?AST DATA • Bug #30628268: CRASH KEHABISAN MEMORI • Bug #27081349: PERILAKU YANG TAK DIHARAPKAN SAAT MENGHAPUS DENGAN FUNGSI SPASIAL • Bug #27230859: PERILAKU YANG TAK DIHARAPKAN SAAT PENANGANAN POLIGON TIDAK VALID" • Bug #27081349: PERILAKU YANG TAK DIHARAPKAN SAAT MENGHAPUS DENGAN SPASIAL" • Bug #26935001: TABEL ALTERNATIF AUTO_INCREMENT MENCOBA MEMBACA INDEKS DARI RUANG TABEL YANG DIHAPUS • Bug #29770705: SERVER CRASH SAAT MELAKUKAN SELECT DENGAN KLAUSA WHERE KHUSUS • Bug #27659490: SELECT MENGGUNAKAN DYNAMIC RANGE DAN INDEX MERGE MENGGUNAKAN TERLALU BANYAK MEMORI(OOM) • Bug #24786290: REPLIKASI RUSAK SETELAH BUG #74145 TERJADI DI MASTER • Bug #27703912: PENGGUNAAN MEMORI BERLEBIH DENGAN BANYAK PERSIAPAN • Bug #20527363: POTONG KERUSAKAN TABEL SEMENTARA:! DICT_ TF2 _FLAG_IS_SET (TABEL, DICT_ _SEMENTARA) TF2

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
		• Bug#23103937 PS_TRUNCATE_ALL_TABLES() TIDAK DAPAT DIGUNAKAN DALAM MODE SUPER_READ_ONLY • Bug #25053286: MENGGUNAKAN TAMPILAN DENGAN KONDISI DALAM PROSEDUR MENYEBABKAN PERILAKU YANG TIDAK BENAR (diperbaiki di 5.6.36) • Bug #25586773: PERILAKU YANG TIDAK BENAR UNTUK CREATE TABLE SELECT IN A LOOP IN SP (diperbaiki di 5.6.39) • Bug #27407480: AUTOMATIC_SP_PRIVILEGES MEMBUTUHKAN HAK ISTIMEWA UNTUK TABEL MYSQL.USER • Bug #26997096: Nilai `relay_log_space` tidak diperbarui secara sinkron sehingga nilainya terkadang jauh lebih tinggi daripada ruang disk aktual yang digunakan oleh log relay. • Bug#15831300 SLAVE_TYPE_CONVERSIONS=ALL_ NON_LOSSY TIDAK BEKERJA SEBAGAIMANA DIHARAPKAN • SSL Bug backport Bug #17087862, Bug #20551271 • Bug #16894092: REGRESI KINERJA DALAM 5.6.6+ UNTUK DIMASUKKAN KE ... SELECT … FROM (diperbaiki di 5.6.15). • Masukkan perbaikan bug yang terkait dengan `SLAVE_TYPE_CONVERSIONS` .

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin basis data Aurora MySQL 2020-11-09 (versi 1.22.3) (Usang)	1.22.3	• Bug #26654685: Indeks ID rusak yang ditemukan selama pemeriksaan kunci asing memunculkan pernyataan • Bug #15831300: Secara default, ketika mempromosikan integer dari tipe yang lebih kecil pada master ke tipe yang lebih besar pada slave (misalnya, dari kolom SMALLINT pada master ke kolom BIGINT pada slave), nilai yang dipromosikan diperlakukan seolah-olah ditandatangani. Dalam kasus seperti itu, perilaku ini dapat dimodifikasi atau ditimpa menggunakan `ALL_SIGNED` , `ALL_UNSIGNED` baik salah satu maupun keduanya dalam set nilai yang ditetapkan untuk variabel sistem server slave_type_conversions. Untuk informasi selengkapnya, lihat Replikasi berbasis baris: promosi dan demosi atribut, serta deskripsi variabel. • Bug #17449901: Dengan `foreign_key_checks=0` , InnoDB mengizinkan penghapusan indeks yang diperlukan oleh batasan kunci asing, sehingga tabel menjadi tidak konsisten dan menyebabkan kegagalan pemeriksaan kunci asing yang terjadi di pemuatan tabel. InnoDB sekarang mencegah penghapusan indeks yang diperlukan oleh batasan kunci asing, sekalipun dengan foreign_key_checks=0. Batasan kunci asing harus dibuang sebelum menghapus indeks kunci asing. • BUG #20768847: ALTER TABLE ... Operasi DROP INDEX pada tabel dengan dependensi kunci asing memunculkan pernyataan.

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin basis data Aurora MySQL 2019-11-25 (versi 1.22.0) (Dihentikan)	1.22.0	• Bug#16346241 - KERUSAKAN SERVER DALAM ITEM_PARAM::QUERY_VAL_STR • Bug#17733850 - NAME_CONST() CRASH DI ITEM_NAME _CONST::ITEM_NAME_CONST() • Bug #20989615 - INNODB AUTO_INCREMENT MENGHASILKAN NILAI YANG SAMA DUA KALI • Bug #20181776 - PENGENDALIAN AKSES TIDAK COCOK DENGAN SEBAGIAN BESAR HOST SPESIFIK SAAT MEMUAT WILDCARD • Bug #27326796 - MYSQL CRASH DENGAN KEGAGALAN PERNYATAAN INNODB DI FILE PARS.CC PARS0 • Bug #20590013 - JIKA ANDA MEMILIKI INDEX TEKS PENUH DAN MELETAKKANNYA ANDA TIDAK LAGI DAPAT MELAKUKAN DDL ONLINE
Pembaruan mesin basis data Aurora MySQL 2019-11-25 (versi 1.21.0) (Dihentikan)	1.21.0	• Bug #19929406: HANDLE_FATAL_SIGNAL (SIG=11) DI __MEMMOVE_ _KEMBALI DARI STRING: :COPY SSSE3 • Bug #17059925: Untuk pernyataan UNION, nilai yang diperiksa baris tidak dihitung dengan benar. Ini diwujudkan sebagai nilai yang terlalu besar untuk kolom `ROWS_EXAMINED` dari tabel pernyataan Skema Kinerja (seperti vents_statements_current). • Bug #11827369: Beberapa kueri dengan subkueri bersarang `SELECT ... FROM DUAL` memunculkan pernyataan. • Bug #16311231: Hasil yang salah dikembalikan apabila sebuah mengandung subkueri dalam klausa `IN` yang memuat operasi XOR di klausa `WHERE`.

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin basis data Aurora MySQL 2019-11-11 (versi 1.20.0) (Dihentikan)	1.20.0	• Bug #19929406: HANDLE_FATAL_SIGNAL (SIG=11) DI __MEMMOVE_ _KEMBALI DARI STRING: :COPY SSSE3 • Bug #17059925: Untuk pernyataan UNION, nilai yang diperiksa baris tidak dihitung dengan benar. Ini diwujudkan sebagai nilai yang terlalu besar untuk kolom ROWS_EXAMINED dari tabel pernyataan Skema Kinerja (seperti events_statements_current). • Bug #11827369: Beberapa kueri dengan subkueri bersarang SELECT ... FROM DUAL memunculkan pernyataan. • Bug #16311231: Hasil yang salah dikembalikan apabila sebuah mengandung subkueri dalam klausa IN yang memuat operasi XOR di klausa WHERE.
Pembaruan mesin basis data Aurora MySQL 2019-09-19 (versi 1.19.5) (Dihentikan)	1.19.5	• CVE-2018-2696 • CVE-2015-4737 • Bug #19929406: HANDLE_FATAL_SIGNAL (SIG=11) DI __MEMMOVE_ _KEMBALI DARI STRING: :COPY SSSE3 • Bug #17059925: Untuk pernyataan UNION, nilai yang diperiksa baris tidak dihitung dengan benar. Ini diwujudkan sebagai nilai yang terlalu besar untuk kolom ROWS_EXAMINED dari tabel pernyataan Skema Kinerja (seperti events_statements_current). • Bug #11827369: Beberapa kueri dengan subkueri bersarang SELECT ... FROM DUAL memunculkan pernyataan. • Bug #16311231: Hasil yang salah dikembalikan apabila sebuah mengandung subkueri dalam klausa IN yang memuat operasi XOR di klausa WHERE.

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin basis data Aurora MySQL 2019-02-07 (versi 1.19.0) (Dihentikan)	1.19.0	• BUG #32917: DETEKSI FILE TEMP-POOL ORPHAN, DAN TANGANI DENGAN MULUS • BUG #63144 BUAT TABEL JIKA TIDAK ADA METADATA TERLALU MEMBATASI
Pembaruan mesin basis data Aurora MySQL 2019-01-17 (versi 1.17.8) (Dihentikan)	1.17.8	• BUG #13418638 BUAT TABEL JIKA TIDAK ADA METADATA TERLALU MEMBATASI
Pembaruan mesin basis data Aurora MySQL 2018-10-08 (versi 1.17.7) (Dihentikan)	1.17.7	• Meletakkan indeks pada kolom kunci asing menyebabkan tabel hilang. (Bug #16208542) • Kebocoran memori di add_derived_key(). (Bug #76349) • Untuk tabel yang dipartisi, kueri dapat memberikan hasil yang berbeda tergantung apakah Penggabungan Indeks digunakan . (Bug #16862316) • Kueri yang menggunakan optimalisasi index_merge (lihat Optimalisasi penggabungan indeks) dapat memberikan hasil yang tidak valid saat menjalankan tabel yang dipartisi oleh HASH. (Bug #17588348)
Pembaruan mesin basis data Aurora MySQL 2018-09-06 (versi 1.17.6) (Dihentikan)	1.17.6	• Untuk pernyataan ALTER TABLE yang mengubah nama atau mengubah nilai default dari kolom BINARY, perubahan dilakukan menggunakan salinan tabel dan bukan di tempatnya . (Bug #67141, Bug #14735373, Bug #69580, Bug #17024290) • Sambungan luar antara tabel biasa dan tabel turunan yang secara implisit dikelompokkan dapat menyebabkan server keluar. (Bug #16177639)

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin basis data Aurora MySQL 2018-03-13 (versi 1.17) (Dihentikan)	1.17.0	• LAST_INSERT_ID direplikasi dengan tidak benar jika filter replikasi digunakan (Bug #69861) • Pencarian memberikan hasil yang berbeda tergantung pada pengaturan INDEX_MERGE (Bug #16862316) • Proc kueri mengeksekusi ulang penyimpanan rutin, rencana kueri tidak efisien (Bug #16346367) • InnoDB FTS : Pernyataan dalam FTS_CACHE_APPEND_DELETED_DOC_IDS (Bug #18079671) • Pernyataan RBT_EMPTY(INDEX_CACHE->WORDS) di ALTER TABLE CHANGE COLUMN (Bug #17536995) • Pencarian teks penuh InnoDB tidak menemukan catatan ketika titik penyimpanan dilibatkan (Bug #70333, Bug #17458835)
Pembaruan mesin basis data Aurora MySQL 2017-11-20 (versi 1.15.1) (Dihentikan)	1.15.1	• Dikembalikan — Instans MySQL macet saat "melakukan indeks SYNC" (Bug #73816) • Dibalik — Pernyataan RBT_EMPTY(INDEX_CACHE->WORDS) di ALTER TABLE CHANGE COLUMN (Bug #17536995) • Dibalik — Pencarian teks penuh InnoDB tidak menemukan catatan ketika titik penyimpanan dilibatkan (Bug #70333)

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin basis data Aurora MySQL 2017-10-24 (versi 1.15) (Dihentikan)	1.15.0	• CREATE USER menerima plugin dan perintah kata sandi, tetapi mengabaikan hash kata sandi (Bug #78033) • Mesin partisi menambahkan bidang pada set bit baca agar dapat mengembalikan entri yang disortir dari indeks yang dipartisi. Hal ini menyebabkan buffer sambungan akan mencoba membaca bidang yang tidak diperlukan. Diperbaiki dengan tidak menambahkan semua kolom ke read_set, tetapi hanya menyortir kolom prefiks yang sudah ditetapkan di read_set. Menambahkan DBUG_ASSERT bahwa jika melakukan key_cmp, setidaknya kolom pertama harus dibaca (Bug #16367691). • Instans MySQL yang macet saat "melakukan indeks SYNC" (Bug #73816) • Pernyataan RBT_EMPTY(INDEX_CACHE->WORDS) di ALTER TABLE CHANGE COLUMN (Bug #17536995) • Pencarian Teks Penuh InnoDB tidak menemukan catatan ketika titik penyimpanan dilibatkan (Bug #70333)
Pembaruan mesin basis data Aurora MySQL 2018-03-13 (versi 1.14.4) (Dihentikan)	1.14.4	• Peristiwa yang tidak berbahaya tidak berhasil dan tidak diuji (Bug #74683) • NEW->OLD ASSERT FAILURE 'GTID_MODE > 0' (Bug #20436436)
Pembaruan mesin basis data Aurora MySQL 2017-08-07 (versi 1.14) (Dihentikan)	1.14.0	Pencarian teks penuh yang digabungkan dengan tabel turunan (subkueri dalam `FROM` klausul) menyebabkan server keluar. Sekarang, jika operasi teks penuh tergantung pada tabel turunan, server akan menghasilkan kesalahan yang menunjukkan bahwa pencarian teks penuh tidak dapat dilakukan pada tabel yang termaterialisasi. (Bug #68751, Bug #16539903)

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin basis data Aurora MySQL: 2017-05-15 (versi 1.13) (Dihentikan)	1.13.0	• Memuat ulang tabel yang dikosongkan dalam keadaan kosong menyebabkan nilai AUTO_INCREMENT direset. (Bug #21454472, Bug #77743) • Catatan indeks tidak ditemukan pada rollback karena inkonsistensi di dalam struktur purge_node_t. Inkonsistensi menimbulkan peringatan dan pesan kesalahan seperti “error in sec index entry update”, “unable to purge a record”, dan “tried to purge sec index entry not marked for deletion”. (Bug #19138298, Bug #70214, Bug #21126772, Bug #21065746) • Perhitungan ukuran tumpukan yang salah untuk operasi qsort menyebabkan luapan tumpukan. (Bug #73979) • Catatan tidak ditemukan dalam indeks pada saat rollback. (Bug #70214, Bug #72419) • ALTER TABLE tambahkan kolom TIMESTAMP saat memperbarui CURRENT_TIMESTAMP menyisipkan ZERO-datas (Bug #17392)

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin basis data Aurora MySQL: 2017-04-05 (versi 1.12) (Dihentikan)	1.12.0	• Memuat ulang tabel yang dikosongkan dalam keadaan kosong menyebabkan nilai AUTO_INCREMENT direset. (Bug #21454472, Bug #77743) • Catatan indeks tidak ditemukan pada rollback karena inkonsistensi di dalam struktur purge_node_t. Inkonsistensi menimbulkan peringatan dan pesan kesalahan seperti “error in sec index entry update”, “unable to purge a record”, dan “tried to purge sec index entry not marked for deletion”. (Bug #19138298, Bug #70214, Bug #21126772, Bug #21065746) • Perhitungan ukuran tumpukan yang salah untuk operasi qsort menyebabkan luapan tumpukan. (Bug #73979) • Catatan tidak ditemukan dalam indeks pada saat rollback. (Bug #70214, Bug #72419) • ALTER TABLE tambahkan kolom TIMESTAMP saat memperbarui CURRENT_TIMESTAMP menyisipkan ZERO-datas (Bug #17392)
Pembaruan mesin basis data Aurora MySQL: 2017-02-23 (versi 1.11) (Dihentikan)	1.11.0	• Menjalankan ALTER table DROP foreign key secara bersamaan dengan operasi DROP lain menyebabkan tabel menghilang. (Bug #16095573) • Beberapa kueri INFORMATION_SCHEMA yang menggunakan ORDER BY tidak menggunakan optimalisasi filesort seperti sebelumnya. (Bug #16423536) • FOUND_ROWS () mengembalikan jumlah baris yang salah pada tabel. (Bug #68458) • Server mengalami kegagalan, bukan memunculkan pesan kesalahan saat terlalu banyak tabel sementara terbuka. (Bug #18948649)

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin basis data Aurora MySQL: 2016-12-14 (versi 1.10) (Dihentikan)	1.10.0	• UNION tabel yang diperoleh memberikan hasil yang salah, IT '1=0/false'-clauses. (Bug #69471) • Server mengalami crash pada ITEM_FUNC_GROUP_CONCAT::FIX_FIELDS pada eksekusi ke-2 dari prosedur yang disimpan. (Bug #20755389) • Mencegah kueri MySQL terhenti terlalu lama selama sinkronisasi cache FTS ke disk dengan mengalihkan tugas sinkronisasi cache ke thread terpisah, segera setelah ukuran cache melewati 10% dari ukuran total. (Bugs #22516559, #73816)
Pembaruan mesin basis data Aurora MySQL: 2016-10-26 (versi 1.8.1) (Dihentikan)	1.8.1	• OpenSSL mengubah parameter panjang kunci Diffie-Hellman karena masalah tersebut. LogJam (Bug #18367167)
Pembaruan mesin basis data Aurora MySQL: 2016-10-18 (versi 1.8) (Dihentikan)	1.8.0	• Saat menghapus semua indeks pada suatu kolom dengan beberapa indeks, InnoDB gagal memblokir operasi DROP INDEX ketika sebuah batasan kunci asing memerlukan sebuah indeks. (Bug #16896810) • Mengatasi crash penambahan batasan kunci asing. (Bug #16413976) • Memperbaiki crash saat mengambil kursor di prosedur yang disimpan, dan menganalisis atau mem-flush tabel pada saat bersamaan. (Bug # 18158639) • Memperbaiki bug kenaikan otomatis saat pengguna mengubah tabel untuk mengubah nilai AUTO_INCREMENT menjadi kurang dari nilai kolom peningkatan otomatis maksimum. (Bug # 16310273)

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin basis data Aurora MySQL: 2016-08-30 (versi 1.7.0) (Dihentikan)	1.7.0	• Meningkatkan skalabilitas dengan mempartisi kunci LOCK_grant. (Port WL #8355) • Kursor pembukaan pada SELECT dalam prosedur yang disimpan menyebabkan segfault. (Port Bug #16499751) • MySQL memberikan hasil yang salah dengan penggunaan khusus. (Bug #11751794) • Crash di GET_SEL_ARG_FOR_KEYPART – yang disebabkan oleh patch untuk bug #11751794. (Bug #16208709) • Hasil yang salah untuk kueri sederhana dengan GROUP BY. (Bug #17909656) • Baris tambahan pada kueri semijoin dengan predikat rentang. (Bug #16221623) • Penambahan klausa ORDER BY setelah subkueri IN dapat menyebabkan baris duplikat dikembalikan. (Bug #16308085) • Crash dengan penjelasan untuk kueri dengan pemindaian longgar untuk GROUP BY, MyISAM. (Bug #16222245) • Pemindaian indeks longgar dengan predikat int yang dikutip menghasilkan data acak. (Bug #16394084) • Jika optimalisasi menggunakan pemindaian indeks longgar, server dapat keluar saat mencoba membuat tabel sementara. (Bug #16436567) • COUNT(DISTINCT) seharusnya tidak menghitung nilai NULL, tetapi dihitung saat pengoptimal menggunakan pemindaian indeks longgar. (Bug #17222452) • Jika sebuah kueri memiliki MIN()/MAX() dan aggregate _function(DISTINCT) (misalnya, SUM(DISTINCT)) dan dijalankan menggunakan pemindaian indeks longgar, nilai hasil MIN()/MAX() diatur dengan tidak tepat. (Bug #17217128)

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin basis data Aurora MySQL: 2016-06-01 (versi 1.6.5) (Dihentikan)	1.6.5	• SLAVE CAN'T CONTINUE REPLICATION AFTER MASTER'S CRASH RECOVERY (Port Bug #17632285)
Pembaruan mesin basis data Aurora MySQL: 2016-04-06 (versi 1.6) (Dihentikan)	1.6.0	• BACKPORT Bug #18694052 FIX FOR ASSERTION `!M_ORDERED_REC_BUFFER' FAILED TO 5.6 (Port Bug #18305270) • SEGV IN MEMCPY(), HA_PARTITION::POSITION (Port Bug # 18383840) • WRONG RESULTS WITH PARTITIONING,INDEX_MERGE AND NO PK (Port Bug # 18167648) • FLUSH TABLES FOR EXPORT: ASSERTION IN HA_PARTITION::EXTRA (Port Bug # 16943907) • SERVER CRASH IN VIRTUAL HA_ROWS HANDLER::MULTI_RANGE_READ_INFO_CONST (Port Bug # 16164031) • RANGE OPTIMIZER CRASHES IN SEL_ARG::RB_INSERT() (Port Bug # 16241773)
Pembaruan mesin basis data Aurora MySQL: 2016-01-11 (versi 1.5) (Dihentikan)	1.5.0	• Mengatasi perbaikan yang belum selesai dalam pencarian teks penuh MySQL yang memengaruhi tabel dengan nama basis data yang dimulai dengan sebuah digit. (Port Bug #17607956)

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin basis data Aurora MySQL: 2015-12-03 (versi 1.4) (Dihentikan)	1.4	• SEGV in FTSPARSE(). (Bug #16446108) • Kamus data InnoDB tidak diperbarui saat mengubah nama kolom. (Bug #19465984) • Crash FTS setelah mengubah nama tabel ke basis data yang berbeda. (Bug #16834860) • Gagal menyiapkan pemicu pada tabel yang terpotong menyebabkan kesalahan 1054. (Bug #18596756) • Perubahan metadata dapat menyebabkan masalah pada eksekusi pemicu. (Bug #18684393) • Materialisasi tidak dipilih untuk bidang UTF8 VARCHAR yang panjang. (Bug #17566396) • Rencana eksekusi yang buruk saat ORDER BY dengan batas X. (Bug #16697792) • Backport bug #11765744 KE 5.1, 5.5, DAN 5.6. (Bug #17083851) • Masalah mutex di SQL/SQL_SHOW.CC menghasilkan. SIG6 Sumber cenderung FILL_VARIABLES. (Bug #20788853) • Backport bug #18008907 ke versi 5.5+. (Bug #18903155) • Menyesuaikan perbaikan untuk kesalahan luapan tumpukan di MySQL 5.7. (Bug #19678930)

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
Pembaruan mesin basis data Aurora MySQL: 2015-10-16 (versi 1.2, 1.3) (Dihentikan)	1.2, 1.3	• Mematikan kueri di dalam innodb menyebabkan crash dengan pernyataan. (Bug #1608883) • Untuk kegagalan dalam membuat thread baru untuk penjadwal peristiwa, pelaksanaan peristiwa, atau koneksi baru, tidak ada pesan yang dituliskan ke log kesalahan. (Bug #16865959) • Jika satu koneksi mengubah basis data default dan secara bersamaan koneksi lain menjalankan SHOW PROCESSLIST, koneksi kedua dapat mengakses memori yang tidak valid ketika mencoba menampilkan memori basis data default koneksi pertama. (Bug #11765252) • PURGE BINARY LOGS berdasarkan desain tidak menghapus file log biner yang sedang digunakan atau aktif, tetapi tidak memberikan pemberitahuan apa pun saat hal ini terjadi. (Bug #13727933) • Untuk beberapa pernyataan, kebocoran memori dapat terjadi ketika pengoptimal menghapus klausa subkueri yang tidak diperlukan. (Bug #15875919) • Selama shutdown, server dapat mencoba mengunci mutex yang tidak diinisialisasi. (Bug #16016493) • Pernyataan yang disiapkan yang menggunakan GROUP_CONCAT() dan klausa ORDER BY yang menyebutkan beberapa kolom dapat menyebabkan server keluar. (Bug #16075310) • Instrumentasi Skema Kinerja tidak ada untuk thread pekerja replika. (Bug #16083949) • `STOP SLAVE` dapat menyebabkan deadlock saat diterbitkan secara bersamaan dengan pernyataan seperti SHOW STATUS yang mengambil nilai untuk satu atau beberapa variabel status `Slave_retried_transactions`, `Slave_heartbeat_period`, `Slave_received_hea`

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
		`rtbeats` , `Slave_last_heartbeat` , atau `Slave_running` . (Bug #16088188) • Kueri teks penuh yang menggunakan mode Boolean dapat memberikan hasil nol pada beberapa kasus di mana istilah pencarian adalah frasa yang dikutip. (Bug #16206253) • Upaya pengoptimal untuk menghapus klausa subkueri redundan memunculkan pernyataan saat menjalankan pernyataan yang disiapkan dengan subkueri dalam klausa ON dari sambungan dalam subkueri. (Bug #16318585) • GROUP_CONCAT tidak stabil, crash di ITEM_SUM: :CLEAN_UP_AFTER_REMOVAL. (Bug #16347450) • Mencoba mengganti daftar stopword pencarian teks lengkap (FTS) InnoDB default dengan membuat tabel InnoDB dengan struktur yang sama seperti INFORMATION_SCHEMA .INNODB_FT_DEFAULT_STOPWORD akan mengakibatkan kesalahan. (Bug #16373868) • Setelah thread klien pada pekerja menjalankan FLUSH TABLES WITH READ LOCK dan diikuti oleh beberapa pembaruan pada master, pekerja tersebut macet saat mengeksekusi `SHOW SLAVE STATUS`. (Bug #16387720) • Saat menguraikan string pencarian yang dibatasi seperti "abc-def" dalam pencarian teks lengkap, InnoDB sekarang menggunakan pembatas kata yang sama dengan MyISAM. (Bug #16419661) • Crash di FTS_AST_TERM_SET_WILDCARD. (Bug #16429306) • SEGFAULT di FTS_AST_VISIT() untuk uji FTS RQG. (Bug # 16435855) • Untuk build debug, ketika pengoptimal menghapus Item_ref yang menunjuk ke subkueri, hal ini menyebabkan server keluar. (Bug #16509874)

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
		• Pencarian teks lengkap pada tabel InnoDB gagal pada pencarian frasa literal yang dikombinasikan dengan operator + atau -. (Bug #16516193) • `START SLAVE` gagal ketika server dimulai dengan opsi `--master-info-repository=TABLE` relay-log-info-repository =TABLE dan dengan komit otomatis disetel ke 0, bersama dengan. `--skip-slave-start` (Bug #16533802) • Hasil pencarian teks lengkap (FTS) InnoDB yang sangat besar dapat menghabiskan terlalu banyak memori. (Bug #16625973) • Dalam build debug, pernyataan dapat terjadi dalam OPT_CHECK_ORDER_BY ketika menggunakan biner secara langsung dalam string pencarian, karena biner mungkin menyertakan byte NULL dan karakter lain yang tidak bermakna. (Bug #16766016) • Untuk beberapa pernyataan, kebocoran memori dapat terjadi ketika pengoptimal menghapus klausa subkueri yang tidak diperlukan. (Bug #16807641) • Ada kemungkinan untuk menyebabkan deadlock setelah menerbitkan FLUSH TABLES WITH READ LOCK dengan menerbitkan `STOP SLAVE` dalam koneksi yang baru dengan pekerja, kemudian menerbitkan `SHOW SLAVE STATUS` menggunakan koneksi asli. (Bug #16856735) • GROUP_CONCAT() dengan pemisah yang tidak valid dapat menyebabkan server keluar. (Bug #16870783) • Server melakukan penguncian berlebihan di mutex LOCK_active_mi dan active_mi->rli->data_lock untuk pernyataan 'pola' SHOW STATUS LIKE apa pun, bahkan ketika pola tidak cocok dengan variabel status yang menggunakan mutex tersebut (`Slave_heartbeat_period` , `Slave_last_heartbeat` , `Slave_received_heartbeats` ,

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
		`Slave_retried_transactions` , `Slave_running`). (Bug #16904035) • Pencarian teks penuh menggunakan modifier MODE BOOLEAN akan menyebabkan kegagalan pernyataan. (Bug #16927092) • Pencarian teks lengkap pada tabel InnoDB gagal pada pencarian yang menggunakan operator boolean +. (Bug #17280122) • Deadlock 4 arah: zombies, purging binlogs, show processlist, show binlogs. (Bug #17283409) • Saat thread SQL yang menunggu kunci commit dimatikan dan dimulai ulang, ini menyebabkan transaksi dilewatkan pada pekerja. (Bug #17450876) • Kegagalan pencarian teks penuh InnoDB akan terjadi karena token yang "tidak diakhiri". String dan panjang string harus diteruskan untuk perbandingan string. (Bug #17659310) • Sejumlah besar tabel InnoDB yang terpartisi dapat menggunakan lebih banyak memori saat digunakan dalam MySQL 5.6 atau 5.7 dibandingkan memori yang digunakan oleh tabel yang sama yang digunakan dalam rilis Server MySQL sebelumnya. (Bug #17780517) • Untuk kueri teks penuh, kegagalan untuk memeriksa bahwa num_token kurang dari max_proximity_item dapat menghasil kan sebuah pernyataan. (Bug #18233051) • Kueri tertentu untuk tabel INFORMATION_SCHEMA TABLES dan COLUMNS dapat menyebabkan penggunaan memori yang berlebihan saat ada sejumlah besar tabel InnoDB yang kosong. (Bug #18592390) • Saat menerapkan transaksi, sebuah bendera sekarang digunakan untuk memeriksa apakah thread telah dibuat, bukan memeriksa thread itu sendiri, yang menggunakan lebih

Pembaruan mesin basis data	Versi	Bug MySQL yang diperbaiki
		banyak sumber daya, terutama ketika menjalankan server dengan master_info_repository=TABLE. (Bug #18684222) • Jika thread klien pada pekerja menjalankan FLUSH TABLES WITH READ LOCK sementara master menjalankan DML, menjalankan `SHOW SLAVE STATUS` di klien yang sama menjadi diblokir, sehingga menyebabkan deadlock. (Bug #19843808) • Mengurutkan berdasarkan hasil GRUP_CONCAT() dapat menyebabkan server keluar. (Bug #19880368)
Pembaruan mesin basis data Aurora MySQL: 2015-08-24 (versi 1.1) (Dihentikan)	1.1	• Basis data InnoDB dengan nama-nama yang dimulai dengan sebuah digit menyebabkan kesalahan pengurai pencarian teks penuh (FTS). (Bug #17607956) • Pencarian teks penuh InnoDB gagal di basis data yang namanya dimulai dengan sebuah digit. (Bug #17161372) • Untuk basis data InnoDB pada Windows, ID objek pencarian teks penuh (FTS) tidak dalam format heksadesimal yang diharapkan. (Bug #16559254) • Regresi kode yang diperkenalkan di MySQL 5.6 berdampak negatif pada kinerja DROP TABLE dan ALTER TABLE. Hal ini dapat menyebabkan penurunan kinerja antara MySQL Server 5.5.x dan 5.6.x. (Bug #16864741)

Kerentanan keamanan yang diperbaiki di Aurora MySQL

Kerentanan dan Eksposur Umum (CVE) adalah daftar entri untuk kerentanan keamanan siber yang diketahui publik. Setiap entri berisi nomor identifikasi, deskripsi, dan setidaknya satu referensi publik.

Di halaman ini, Anda dapat menemukan daftar kerentanan keamanan yang diperbaiki di Aurora MySQL. Untuk informasi umum tentang keamanan untuk Aurora, lihat Keamanan di Amazon Aurora di Panduan Pengguna Amazon Aurora. Untuk informasi keamanan tambahan untuk Aurora MySQL, lihat Keamanan dengan Amazon Aurora MySQL di Panduan Pengguna Amazon Aurora.

Kami menyarankan Anda selalu meningkatkan versi ke rilis Aurora terbaru agar Anda terlindungi dari kerentanan yang diketahui. Anda dapat menggunakan halaman ini untuk memverifikasi apakah versi Aurora MySQL tertentu memiliki perbaikan untuk kerentanan keamanan tertentu. Jika klaster Anda tidak memiliki perbaikan keamanan, Anda dapat melihat versi Aurora MySQL mana yang harus Anda gunakan untuk perbaikan tersebut.

CVEs tetap di Aurora MySQL versi 1, 2 dan 3 juga tercantum dalam catatan rilis untuk versi itu:

- Pembaruan mesin basis data untuk Amazon Aurora MySQL versi 2
- Pembaruan mesin basis data untuk Amazon Aurora MySQL versi 2
- Pembaruan mesin basis data untuk Amazon Aurora MySQL versi 1 (Dihentikan)

> **Note**
>
> Rilis awal Aurora MySQL versi 3 mencakup semua tetap CVEs hingga komunitas MySQL 8.0.23. Untuk masa depan CVEs yang diperbaiki, cari yang tercantum di sini dan di catatan rilis Aurora MySQL versi 3.

CVEs dan versi MySQL Aurora tetap minimum

- CVE-2024-37371: 3.08.0
- CVE-2024-25062 : 3.08.0, 2.12.4
- CVE-2024-21185: 3.08.0
- CVE-2024-21179: 3.08.0
- CVE-2024-21177 : 3.08.0, 2.12.4

- CVE-2024-21173: 3.08.0
- CVE-2024-21166: 3.08.0
- CVE-2024-21165: 3.08.0
- CVE-2024-21163: 3.08.0
- CVE-2024-21162: 3.08.0
- CVE-2024-21160: 3.08.0
- CVE-2024-21159: 3.08.0
- CVE-2024-21157: 3.08.0
- CVE-2024-21142 : 3.08.0, 2.12.4
- CVE-2024-21137: 3.08.0
- CVE-2024-21135: 3.08.0
- CVE-2024-21134: 3.08.0
- CVE-2024-21130: 3.08.0
- CVE-2024-21129: 3.08.0
- CVE-2024-21127: 3.08.0
- CVE-2024-21125: 3.08.0
- CVE-2024-21097:2.12.3, 2.11.6
- CVE-2024-21096:2.12.3, 2.11.6
- CVE-2024-21069:2.12.3, 2.11.6
- CVE-2024-21062:2.12.3, 2.11.6
- CVE-2024-21057:2.12.3, 2.11.6
- CVE-2024-21055:2.12.3, 2.11.6
- CVE-2024-21054:2.12.3, 2.11.6
- CVE-2024-21047:2.12.3, 2.11.6
- CVE-2024-21013:3.08.0, 2.12.3 , 2.11.6
- CVE-2024-21009:2.12.3, 2.11.6
- CVE-2024-21008:2.12.3, 2.11.6
- CVE-2024-21000: 3.08.0
- CVE-2024-20998 : 2.12.3, 2.11.6
- CVE-2024-20996: 3.08.0

- CVE-2024-20993 : 2.12.3, 2.11.6
- CVE-2024-20963 : 2.12.2, 2.11.6, 2.11.5
- CVE-2024-0853 : 3.08.0, 3.06.1, 3.04.3, 2.12.3, 2.11.6
- CVE-2023-44487 : 3.08.0, 3.06.1, 2.12.4, 2.12.3, 2.11.6
- CVE-2023-39975 : 3.07.0, 3.06.0, 3.05.2, 3.04.2, 2.12.2, 2.11.6, 2.11.5
- CVE-2023-38546 : 3.07.0, 3.06.0, 3.04.2, 2.11.5
- CVE-2023-38545 : 3.07.0, 3.06.0, 3.05.2, 3.05.1, 3.05.0.1, 3.04.2, 3.03.3, 2.12.2, 2.12.1, 2.12.0.1, 2.11.6, 2.11.5
- CVE-2023-22084: 2.11.5
- CVE-2023-22053: 2.12.1
- CVE-2023-22028:2.12.1, 2.11.5
- CVE-2023-22026:2.12.1, 2.11.5
- CVE-2023-22015:2.12.1, 2.11.5
- CVE-2023-21963: 3.04.0, 3.03.2, 2.12.0, 2.11.3
- CVE-2023-21912 : 3.04.0, 3.03.2, 2.12.3, 2.12.0, 2.11.6, 2.11.3
- CVE-2023-21840: 2.12.0
- CVE-2023-0215: 3.04.0, 3.03.2, 2.12.0, 2.11.3
- CVE-2022-43551: 3.04.0, 3.03.2, 2.12.0, 2.11.3
- CVE-2022-37434: 3.05.0, 3.04.0, 3.03.2, 2.12.0, 2.11.3
- CVE-2022-32221: 3.03.0, 2.12.0, 2.11.1, 2.07.9
- CVE-2022-24407: 2.12.1, 2.11.4
- CVE-2022-21635: 3.04.0
- CVE-2022-21556: 3.04.0
- CVE-2022-21460: 2.11.0
- CVE-2022-21451: 3.03.0, 3.02.2, 2.11.0
- CVE-2022-21444: 3.03.0, 3.02.2, 2.11.0, 2.10.3
- CVE-2022-21417 : 2.11.0
- CVE-2022-21352: 3.04.0
- CVE-2022-21344: 2.10.3
- CVE-2022-21304: 2.11.0, 2.10.3
- CVE-2022-21303: 2.11.0

- CVE-2022-21245: 2.11.0, 2.10.3, 2.07.8
- CVE-2022-0778: 3.02.1, 2.11.0
- CVE-2021-36222: 3.02.2, 3.01.1, 2.12.0, 2.11.1, 2.11.0, 2.10.3, 2.10.2, 2.07.8
- CVE-2021-35630: 3.04.0
- CVE-2021-35624: 3.04.0, 2.10.2
- CVE-2021-35604: 2.10.2
- CVE-2021-28196: 2.11.0
- CVE-2021-23841: 2.11.0, 2.10.0, 2.09.3, 1.23.3
- CVE-2021-22946: 3.02.0, 3.01.1, 2.12.0
- CVE-2021-22926: 3.02.2, 3.01.1, 2.11.1, 2.11.0, 2.10.3, 2.10.22.07.8
- CVE-2021-3712: 2.09.3
- CVE-2021-3449: 2.11.0, 2.10.0, 2.09.3, 1.23.3
- CVE-2021-2390: 2.10.2
- CVE-2021-2389: 2.10.2
- CVE-2021-2385: 2.10.2
- CVE-2021-2356: 2.10.2
- CVE-2021-2307: 2.11.0, 2.10.1, 2.09.3, 1.23.4
- CVE-2021-2226: 2.11.0, 2.10.1, 2.09.3, 1.23.4
- CVE-2021-2202: 2.11.0
- CVE-2021-2194: 2.11.0, 2.10.1
- CVE-2021-2179: 2.11.0
- CVE-2021-2178: 2.11.0
- CVE-2021-2174: 2.11.0, 2.10.1, 2.09.3
- CVE-2021-2171: 2.11.0, 2.10.1, 2.09.3
- CVE-2021-2169: 2.12.0, 2.11.1, 2.11.0, 2.10.1, 2.09.3
- CVE-2021-2166: 2.11.0, 2.10.1, 2.09.3
- CVE-2021-2160: 2.11.0, 2.10.1, 1.23.4
- CVE-2021-2154: 2.11.0, 2.10.1, 2.09.3, 1.23.4
- CVE-2021-2144: 2.07.3
- CVE-2021-2060: 2.10.1, 2.09.3, 1.23.4

- CVE-2021-2032: 2.10.1, 2.09.3, 1.23.4
- CVE-2021-2001: 2.10.1, 2.09.3, 1.23.4
- CVE-2020-28196: 2.10.0, 2.09.3, 1.23.3
- CVE-2020-14867: 1.23.2, 1.22.4
- CVE-2020-14812: 2.09.2, 2.07.4, 1.23.2, 1.22.4
- CVE-2020-14793: 2.09.2, 2.07.4, 1.23.2, 1.22.4
- CVE-2020-14790: 2.10.0, 2.09.2, 2.07.4
- CVE-2020-14776: 2.10.0
- CVE-2020-14775: 2.09.2, 2.07.4
- CVE-2020-14769: 2.09.3, 2.09.2, 2.07.4, 1.23.2, 1.22.4
- CVE-2020-14765: 2.09.2, 2.07.4, 1.23.2, 1.22.4
- CVE-2020-14760: 2.09.2, 2.07.4
- CVE-2020-14672: 2.09.2, 2.07.4, 1.23.2, 1.22.4
- CVE-2020-14567: 2.10.0, 2.09.1, 2.08.3, 2.07.3
- CVE-2020-14559: 2.10.0, 2.09.1, 2.08.3, 2.07.3, 1.23.1, 1.22.3
- CVE-2020-14553: 2.10.0, 2.09.1, 2.08.3, 2.07.3
- CVE-2020-14547: 2.10.0, 2.09.1, 2.08.3, 2.07.3
- CVE-2020-14540: 2.10.0, 2.09.1, 2.08.3, 2.07.3
- CVE-2020-14539: 2.10.0, 1.23.1, 1.22.3
- CVE-2020-11105 : 3.07.0, 3.06.0, 3.05.2, 3.04.2, 2.12.1, 2.11.5
- CVE-2020-11104 : 3.07.0, 3.06.0, 3.05.2, 3.04.2, 2.12.1, 2.11.5
- CVE-2020-2812: 2.09.1, 2.08.3, 2.07.3, 1.22.3
- CVE-2020-2806: 2.09.1, 2.08.3, 2.07.3
- CVE-2020-2780: 2.09.1, 2.08.3, 2.07.3, 1.22.3
- CVE-2020-2765: 2.09.1, 2.08.3, 2.07.3
- CVE-2020-2763: 2.09.1, 2.08.3, 2.07.3, 1.22.3
- CVE-2020-2760: 2.09.1, 2.08.3, 2.07.3, 2.04.9
- CVE-2020-2579: 2.09.1, 2.08.3, 2.07.3, 1.22.3
- CVE-2020-1971: 2.09.2, 2.07.4, 1.23.2, 1.22.4
- CVE-2019-17543: 2.10.2, 2.09.3, 2.07.7

- CVE-2019-5443: 2.08.0, 2.04.9
- CVE-2019-3822: 2.08.0, 2.04.9
- CVE-2019-2960: 2.10.2, 2.09.3, 2.07.7
- CVE-2019-2948: 2.09.0
- CVE-2019-2924: 2.07.0, 2.04.9, 1.22.0
- CVE-2019-2923: 2.07.0, 2.04.9, 1.22.0
- CVE-2019-2922: 2.07.0, 2.04.9, 1.22.0
- CVE-2019-2911: 2.09.0, 2.04.9, 1.23.0
- CVE-2019-2910: 2.07.0, 2.04.9, 1.22.0
- CVE-2019-2805: 2.06.0, 2.04.9, 1.22.0
- CVE-2019-2791: 2.06.0, 2.04.9
- CVE-2019-2778: 2.06.0, 2.04.9
- CVE-2019-2758: 2.06.0, 2.04.9
- CVE-2019-2740: 2.07.3, 2.04.9, 1.22.0
- CVE-2019-2739: 2.06.0, 2.04.9
- CVE-2019-2731: 2.09.0
- CVE-2019-2730: 2.06.0, 2.04.9, 1.22.0
- CVE-2019-2628: 2.04.9
- CVE-2019-2581: 2.09.0
- CVE-2019-2537: 2.09.0, 1.23.0
- CVE-2019-2534: 2.05.0, 2.04.3, 1.21.0, 1.20.0, 1.19.1
- CVE-2019-2482: 2.09.0
- CVE-2019-2434: 2.09.0
- CVE-2019-2420: 2.09.0
- CVE-2018-3284: 2.09.0
- CVE-2018-3251: 2.10.0
- CVE-2018-3156: 2.10.0
- CVE-2018-3155: 2.05.0, 2.04.3
- CVE-2018-3143: 2.10.0, 1.23.2
- CVE-2018-3065: 2.09.0

- CVE-2018-3064: 2.06.0, 2.04.9, 1.22.0
- CVE-2018-3058: 2.06.0, 2.04.9, 1.22.0
- CVE-2018-3056: 2.05.0, 2.04.4
- CVE-2018-2813: 2.04.9
- CVE-2018-2787: 2.09.0, 1.23.0
- CVE-2018-2786: 2.06.0, 2.04.9
- CVE-2018-2784: 2.09.0, 1.23.0
- CVE-2018-2696: 2.05.0, 2.04.5, 1.21.0, 1.20.0, 1.19.5
- CVE-2018-2645: 2.09.0, 1.23.0
- CVE-2018-2640: 2.09.0, 1.23.0
- CVE-2018-2612: 2.05.0, 2.04.3, 1.21.0, 1.20.0, 1.19.1
- CVE-2018-2562: 2.05.0, 2.04.4, 1.21.0, 1.20.0, 1.19.2
- CVE-2018-0734: 2.05.0, 2.04.3, 1.21.0, 1.20.0, 1.19.1
- CVE-2017-3653: 2.06.0, 2.04.9, 1.22.0
- CVE-2017-3599: 2.05.0, 2.04.3, 1.21.0, 1.20.0, 1.19.1
- CVE-2017-3465: 2.06.0, 2.04.9
- CVE-2017-3464: 1.22.0, 2.04.9
- CVE-2017-3455: 2.06.0, 2.04.9
- CVE-2017-3329: 2.05.0, 2.04.4, 1.21.0, 1.20.0, 1.19.2
- CVE-2017-3244: 2.06.0, 2.04.9, 1.22.0
- CVE-2016-8287: 2.07.2
- CVE-2016-5634: 2.07.2
- CVE-2016-5612: 2.06.0, 2.04.9, 1.22.0
- CVE-2016-5440: 2.10.0
- CVE-2016-5439: 1.22.0, 2.03.3
- CVE-2016-5436: 2.04.9, 2.03.3
- CVE-2016-3518: 2.04.5
- CVE-2016-3495: 2.03.2
- CVE-2016-3486: 2.02.2
- CVE-2016-0606: 1.22.0

- CVE-2015-4904: 1.22.0
- CVE-2015-4879: 1.22.0
- CVE-2015-4864: 1.22.0
- CVE-2015-4830: 1.22.0
- CVE-2015-4826: 1.22.0
- CVE-2015-4737: 1.21.0, 1.20.0, 1.19.5
- CVE-2015-2620: 1.22.0
- CVE-2015-0382: 1.22.0
- CVE-2015-0381: 1.22.0
- CVE-2014-6555: 1.22.0
- CVE-2014-6489: 1.22.0
- CVE-2014-4260: 1.22.0
- CVE-2014-4258: 1.22.0
- CVE-2014-2444: 1.22.0
- CVE-2014-2436: 1.22.0
- CVE-2014-0393: 1.22.0
- CVE-2013-5908: 1.22.0
- CVE-2013-5881: 1.22.0
- CVE-2013-5807: 1.22.0
- CVE-2013-3811: 1.22.0
- CVE-2013-3807: 1.22.0
- CVE-2013-3806: 1.22.0
- CVE-2013-3804: 1.22.0
- CVE-2013-2381: 1.22.0
- CVE-2013-2378: 1.22.0
- CVE-2013-2375: 1.22.0
- CVE-2013-1523: 1.22.0
- CVE-2012-5615: 1.22.0

Riwayat dokumen untuk Catatan Rilis Aurora MySQL

Tabel berikut menjelaskan rilis dokumentasi untuk Catatan Rilis Aurora MySQL.

Perubahan	Deskripsi	Tanggal
Aurora MySQL versi 3.12.0, kompatibel dengan MySQL 8.0.44	Aurora MySQL versi 3.12.0 tersedia. Versi ini kompatibel dengan MySQL 8.0.44. Untuk detail selengkapnya, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0.	Februari 17, 2026
Aurora MySQL versi 3.10.3, kompatibel dengan MySQL 8.0.42	Aurora MySQL versi 3.10.3 tersedia. Versi ini kompatibel dengan MySQL 8.0.42. Untuk informasi selengkapnya, lihat pembaruan mesin database Aurora MySQL 2026-01-02 (versi 3.10.3, kompatibel dengan MySQL 8.0.42).	Januari 2, 2026
Aurora MySQL versi 3.04.6, kompatibel dengan MySQL 8.0.28	Aurora MySQL versi 3.04.6 tersedia. Versi ini kompatibel dengan MySQL 8.0.28. Untuk detail selengkapnya, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0.	Januari 2, 2026
Aurora MySQL versi 3.11.1, kompatibel dengan MySQL 8.0.43	Aurora MySQL versi 3.11.1 tersedia. Versi ini kompatibel dengan MySQL 8.0.43. Untuk informasi selengkapnya, lihat pembaruan mesin database Aurora MySQL 2025-12-1	Desember 16, 2025

	6 (versi 3.11.1, kompatibel dengan MySQL 8.0.43).	
Aurora MySQL versi 3.10.2, kompatibel dengan MySQL 8.0.42	Aurora MySQL versi 3.10.2 tersedia. Versi ini kompatibel dengan MySQL 8.0.42. Untuk informasi selengkapnya, lihat pembaruan mesin database Aurora MySQL 2025-11-20 (versi 3.10.2, kompatibel dengan MySQL 8.0.42).	November 20, 2025
Aurora MySQL versi 3.11.0, kompatibel dengan MySQL 8.0.43	Aurora MySQL versi 3.11.0 tersedia. Versi ini kompatibel dengan MySQL 8.0.43. Untuk informasi selengkapnya, lihat pembaruan mesin database Aurora MySQL 2025-11-13 (versi 3.11.0, kompatibel dengan MySQL 8.0.43).	November 13, 2025
Aurora MySQL versi 3.10.1, kompatibel dengan MySQL 8.0.42	Aurora MySQL versi 3.10.1 tersedia. Versi ini kompatibel dengan MySQL 8.0.42. Untuk detail selengkapnya, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0.	September 30, 2025
Aurora MySQL versi 3.10.0, kompatibel dengan MySQL 8.0.42	Aurora MySQL versi 3.10.0 tersedia. Versi ini kompatibel dengan MySQL 8.0.42. Untuk detail selengkapnya, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0.	Juli 31, 2025

Aurora MySQL versi 3.09.0, kompatibel dengan MySQL 8.0.40	Aurora MySQL versi 3.09.0 tersedia. Versi ini kompatibel dengan MySQL 8.0.40. Untuk detail selengkapnya, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0.	14 Mei 2025
Aurora MySQL versi 3.04.4, kompatibel dengan MySQL 8.0.28	Aurora MySQL versi 3.04.4 tersedia. Versi ini kompatibel dengan MySQL 8.0.28. Untuk detail selengkapnya, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0.	5 Mei 2025
Aurora MySQL versi 2.12.5, kompatibel dengan MySQL 5.7.44	Aurora MySQL versi 2.12.5 tersedia. Versi ini kompatibel dengan MySQL 5.7.44. Untuk detail selengkapnya, lihat Aurora MySQL versi 2 yang kompatibel dengan MySQL 5.7.	April 9, 2025
Aurora MySQL versi 3.08.2, kompatibel dengan MySQL 8.0.39	Aurora MySQL versi 3.08.2 tersedia. Versi ini kompatibel dengan MySQL 8.0.39. Untuk detail selengkapnya, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0.	Februari 17, 2025

Aurora MySQL versi 3.08.1, kompatibel dengan MySQL 8.0.39	Aurora MySQL versi 3.08.1 tersedia. Versi ini kompatibel dengan MySQL 8.0.39. Untuk detail selengkapnya, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0.	Februari 17, 2025
Aurora MySQL versi 3.08.0, kompatibel dengan MySQL 8.0.39	Aurora MySQL versi 3.08.0 tersedia. Versi ini kompatibel dengan MySQL 8.0.39. Untuk detail selengkapnya, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0.	November 18, 2024
Aurora MySQL versi 2.12.4, kompatibel dengan MySQL 5.7.44	Aurora MySQL versi 2.12.4 tersedia. Versi ini kompatibel dengan MySQL 5.7.44. Untuk detail selengkapnya, lihat Aurora MySQL versi 2 yang kompatibel dengan MySQL 5.7.	Oktober 23, 2024
Aurora MySQL versi 3.07.1, kompatibel dengan MySQL 8.0.36	Aurora MySQL versi 3.07.1 tersedia. Versi ini kompatibel dengan MySQL 8.0.36. Untuk detail selengkapnya, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0.	Juli 23, 2024

Aurora MySQL versi 2.11.6, kompatibel dengan MySQL 5.7.12	Aurora MySQL versi 2.11.6 tersedia. Versi ini kompatibel dengan MySQL 5.7.12. Untuk detail selengkapnya, lihat Aurora MySQL versi 2 yang kompatibel dengan MySQL 5.7.	Juli 19, 2024
Aurora MySQL versi 2.12.3, kompatibel dengan MySQL 5.7.44	Aurora MySQL versi 2.12.3 tersedia. Versi ini kompatibel dengan MySQL 5.7.44. Untuk detail selengkapnya, lihat Aurora MySQL versi 2 yang kompatibel dengan MySQL 5.7.	9 Juli 2024
Aurora MySQL versi 3.06.1, kompatibel dengan MySQL 8.0.34	Aurora MySQL versi 3.06.1 tersedia. Versi ini kompatibel dengan MySQL 8.0.34. Untuk detail selengkapnya, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0.	Juni 26, 2024
Aurora MySQL versi 3.04.3, kompatibel dengan MySQL 8.0.28	Aurora MySQL versi 3.04.3 tersedia. Versi ini kompatibel dengan MySQL 8.0.28. Untuk detail selengkapnya, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0.	Juni 26, 2024

Aurora MySQL versi 3.07.0, kompatibel dengan MySQL 8.0.36	Aurora MySQL versi 3.07.0 tersedia. Versi ini kompatibel dengan MySQL 8.0.36. Untuk detail selengkapnya, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0.	4 Juni 2024
Aurora MySQL versi 2.11.5, kompatibel dengan MySQL 5.7.12	Aurora MySQL versi 2.11.5 tersedia. Versi ini kompatibel dengan MySQL 5.7.12. Untuk detail selengkapnya, lihat Aurora MySQL versi 2 yang kompatibel dengan MySQL 5.7.	26 Maret 2024
Aurora MySQL versi 2.12.2, kompatibel dengan MySQL 5.7.44	Aurora MySQL versi 2.12.2 tersedia. Versi ini kompatibel dengan MySQL 5.7.44. Untuk detail selengkapnya, lihat Aurora MySQL versi 2 yang kompatibel dengan MySQL 5.7.	Maret 19, 2024
Aurora MySQL versi 3.04.2, kompatibel dengan MySQL 8.0.28	Aurora MySQL versi 3.04.2 tersedia. Versi ini kompatibel dengan MySQL 8.0.28. Untuk detail selengkapnya, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0.	Maret 15, 2024

Aurora MySQL versi 3.06.0, kompatibel dengan MySQL 8.0.34	Aurora MySQL versi 3.06.0 tersedia. Versi ini kompatibel dengan MySQL 8.0.34. Untuk detail selengkapnya, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0.	7 Maret 2024
Aurora MySQL versi 3.05.2, kompatibel dengan MySQL 8.0.32	Aurora MySQL versi 3.05.2 tersedia. Versi ini kompatibel dengan MySQL 8.0.32. Untuk detail selengkapnya, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0.	Januari 31, 2024
Aurora MySQL versi 2.12.1, kompatibel dengan MySQL 5.7.40	Aurora MySQL versi 2.12.1 tersedia. Versi ini kompatibel dengan MySQL 5.7.40. Untuk detail selengkapnya, lihat Aurora MySQL versi 2 yang kompatibel dengan MySQL 5.7.	28 Desember 2023
Aurora MySQL versi 3.03.3, kompatibel dengan MySQL 8.0.26	Aurora MySQL versi 3.03.3 tersedia. Versi ini kompatibel dengan MySQL 8.0.26. Untuk detail selengkapnya, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0.	8 Desember 2023

Aurora MySQL versi 3.05.1, kompatibel dengan MySQL 8.0.32	Aurora MySQL versi 3.05.1 tersedia. Versi ini kompatibel dengan MySQL 8.0.32. Untuk detail selengkapnya, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0.	21 November 2023
Aurora MySQL versi 3.04.1, kompatibel dengan MySQL 8.0.28	Aurora MySQL versi 3.04.1 tersedia. Versi ini kompatibel dengan MySQL 8.0.28. Untuk detail selengkapnya, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0.	13 November 2023
Aurora MySQL versi 3.05.0.1, kompatibel dengan MySQL 8.0.32, Beta	Aurora MySQL versi 3.05.0.1 tersedia. Versi ini kompatibel dengan MySQL 8.0.32. Untuk detail selengkapnya, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0.	30 Oktober 2023
Aurora MySQL versi 3.05.0, kompatibel dengan MySQL 8.0.32	Aurora MySQL versi 3.05.0 tersedia. Versi ini kompatibel dengan MySQL 8.0.32. Untuk detail selengkapnya, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0.	25 Oktober 2023

Aurora MySQL versi 2.12.0.1, kompatibel dengan MySQL 5.7.40, Beta	Aurora MySQL versi 2.12.0.1 Beta tersedia. Versi ini kompatibel dengan MySQL 5.7.40. Untuk detail selengkapnya, lihat Aurora MySQL versi 2 yang kompatibel dengan MySQL 5.7.	25 Oktober 2023
Aurora MySQL versi 2.11.4, kompatibel dengan MySQL 5.7.12	Aurora MySQL versi 2.11.4 tersedia. Versi ini kompatibel dengan MySQL 5.7.12. Untuk detail selengkapnya, lihat Aurora MySQL versi 2 yang kompatibel dengan MySQL 5.7.	17 Oktober 2023
Aurora MySQL versi 3.03.2, kompatibel dengan MySQL 8.0.26	Aurora MySQL versi 3.03.2 tersedia. Versi ini kompatibel dengan MySQL 8.0.26. Untuk detail selengkapnya, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0.	29 Agustus 2023
Aurora MySQL versi 2.07.10, kompatibel dengan MySQL 5.7.12	Aurora MySQL versi 2.07.10 tersedia. Versi ini kompatibel dengan MySQL 5.7.12. Untuk detail selengkapnya, lihat Aurora MySQL versi 2 yang kompatibel dengan MySQL 5.7.	15 Agustus 2023

Aurora MySQL versi 3.04.0, kompatibel dengan MySQL 8.0.28	Aurora MySQL versi 3.04.0 tersedia. Versi ini kompatibel dengan MySQL 8.0.28. Untuk detail selengkapnya, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0.	31 Juli 2023
Aurora MySQL versi 2.12.0, kompatibel dengan MySQL 5.7.40	Aurora MySQL versi 2.12.0 tersedia. Versi ini kompatibel dengan MySQL 5.7.40. Untuk detail selengkapnya, lihat Aurora MySQL versi 2 yang kompatibel dengan MySQL 5.7.	25 Juli 2023
Aurora MySQL versi 2.11.3, kompatibel dengan MySQL 5.7.12	Aurora MySQL versi 2.11.3 tersedia. Versi ini kompatibel dengan MySQL 5.7.12. Untuk detail selengkapnya, lihat Aurora MySQL versi 2 yang kompatibel dengan MySQL 5.7.	9 Juni 2023
Aurora MySQL versi 3.03.1, kompatibel dengan MySQL 8.0.26	Aurora MySQL versi 1.22.3 tersedia. Versi ini kompatibel dengan MySQL 8.0.26. Untuk detail selengkapnya, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0.	11 Mei 2023

Aurora MySQL versi 2.07.9, kompatibel dengan MySQL 5.7.12.	Aurora MySQL versi 2.07.9 tersedia. Versi ini kompatibel dengan MySQL 5.7.12. Untuk detail selengkapnya, lihat Aurora MySQL versi 2 yang kompatibel dengan MySQL 5.7.	4 Mei 2023
Aurora MySQL versi 3.02.3, kompatibel dengan MySQL 8.0.23	Aurora MySQL versi 3.02.3 tersedia. Versi ini kompatibel dengan MySQL 8.0.23. Untuk detail selengkapnya, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0.	17 April 2023
Aurora MySQL versi 2.11.2, kompatibel dengan MySQL 5.7.12	Aurora MySQL versi 2.11.2 tersedia. Versi ini kompatibel dengan MySQL 5.7.12. Untuk detail selengkapnya, lihat Aurora MySQL versi 2 yang kompatibel dengan MySQL 5.7.	24 Maret 2023
Aurora MySQL versi 3.03.0, kompatibel dengan MySQL 8.0.26	Aurora MySQL versi 3.03.0 tersedia. Versi ini kompatibel dengan MySQL 8.0.26. Untuk detail selengkapnya, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0.	1 Maret 2023

Aurora MySQL versi 2.11.1, kompatibel dengan MySQL 5.7.12.	Aurora MySQL versi 2.11.1 tersedia. Versi ini kompatibel dengan MySQL 5.7.12. Untuk detail selengkapnya, lihat Aurora MySQL versi 2 yang kompatibel dengan MySQL 5.7.	14 Februari 2023
Aurora MySQL versi 3.02.2, kompatibel dengan MySQL 8.0.23	Aurora MySQL versi 3.02.2 tersedia. Versi ini kompatibel dengan MySQL 8.0.23. Untuk detail selengkapnya, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0.	18 November 2022
Aurora MySQL versi 2.10.3, kompatibel dengan MySQL 5.7	Aurora MySQL versi 2.10.3 tersedia. Versi ini kompatibel dengan MySQL 5.7. Untuk detail selengkapnya, lihat Aurora MySQL versi 2 yang kompatibel dengan MySQL 5.7.	1 November 2022
Aurora MySQL versi 2.11.1, kompatibel dengan MySQL 5.7.12	Aurora MySQL versi 2.11.0 tersedia. Versi ini kompatibel dengan MySQL 5.7.12. Untuk detail selengkapnya, lihat Aurora MySQL versi 2 yang kompatibel dengan MySQL 5.7.	25 Oktober 2022

Aurora MySQL versi 3.02.1, kompatibel dengan MySQL 8.0.23	Aurora MySQL versi 3.02.1 tersedia. Versi ini kompatibel dengan MySQL 8.0.23. Untuk detail selengkapnya, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0.	7 September 2022
Peningkatan in-place untuk Aurora Serverless v1 yang kompatibel dengan MySQL 5.6	Anda dapat melakukan peningkatan in-place untuk klaster Aurora Serverless v1 yang kompatibel dengan MySQL 5.6 untuk mengubah klaster yang ada menjadi klaster Aurora Serverles s v1 yang kompatibel dengan MySQL 5.7. Untuk informasi selengkapnya, lihat Pembaruan mesin Aurora MySQL Nirserver 5.7 2020-06-18 (versi 2.07.1).	16 Juni 2022
Peningkatan in-place untuk Aurora Serverless v1 yang kompatibel dengan MySQL 5.6	Anda dapat melakukan peningkatan in-place untuk klaster Aurora Serverless v1 yang kompatibel dengan MySQL 5.6 untuk mengubah klaster yang ada menjadi klaster Aurora Serverles s v1 yang kompatibel dengan MySQL 5.7. Untuk informasi selengkapnya, lihat Pembaruan mesin Aurora MySQL Nirserver 5.7 2020-06-18 (versi 2.07.1).	16 Juni 2022

Pembaruan mesin basis data Aurora MySQL 2022-06-16 (versi 2.07.8) tersedia.	Aurora MySQL versi 2.07.8 tersedia.	16 Juni 2022
Aurora MySQL versi 3.02.0, kompatibel dengan MySQL 8.0.23	Aurora MySQL versi 3.02.0 tersedia. Versi ini kompatibel dengan MySQL 8.0.23. Untuk detail selengkapnya, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0.	20 April 2022
Aurora MySQL versi 3.01.1, kompatibel dengan MySQL 8.0.23	Aurora MySQL versi 3.01.1 tersedia. Versi ini kompatibel dengan MySQL 8.0.23. Untuk detail selengkapnya, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0.	15 April 2022
Rilis awal	Rilis awal Catatan Rilis Aurora MySQL.	22 Maret 2022
Aurora MySQL versi 2.10.2	Aurora MySQL versi 2.10.2 tersedia.	26 Januari 2022
Aurora MySQL versi 2.08.4	Aurora MySQL versi 2.08.4 tersedia.	6 Januari 2022
Aurora MySQL versi 2.07.7	Aurora MySQL versi 2.07.7 tersedia.	24 November 2021

Aurora MySQL versi 3.01.0, kompatibel dengan MySQL 8.0.23	Aurora MySQL versi 3.01.0 tersedia. Versi ini kompatibel dengan MySQL 8.0.23. Untuk detail selengkapnya, lihat Aurora MySQL versi 3 yang kompatibel dengan MySQL 8.0.	18 November 2021
Aurora MySQL versi 2.09.3	Aurora MySQL versi 2.09.3 tersedia.	12 November 2021
Aurora MySQL versi 2.10.1	Aurora MySQL versi 2.10.1 tersedia.	21 Oktober 2021
Aurora MySQL versi 1.23.4	Aurora MySQL versi 1.23.4 tersedia.	30 September 2021
Aurora MySQL versi 2.07.6	Aurora MySQL versi 2.07.6 tersedia.	2 September 2021
Aurora MySQL versi 2.07.5	Aurora MySQL versi 2.07.5 tersedia.	6 Juli 2021
Aurora MySQL versi 1.23.3	Aurora MySQL versi 1.23.3 tersedia.	28 Juni 2021
Aurora MySQL versi 1.22.5	Aurora MySQL versi 1.22.5 tersedia.	3 Juni 2021
Aurora MySQL versi 2.10.0	Aurora MySQL versi 2.10.0 tersedia. Beberapa sorotan termasuk ketersediaan instance pembaca yang lebih tinggi selama restart penulis, peningkatan patching zero-downtime (ZDP), peningkatan restart zero-downtime (ZDR), dan optimasi cache binlog. I/O	25 Mei 2021

Aurora MySQL versi 1.23.2	Aurora MySQL versi 1.23.2 tersedia.	18 Maret 2021
Aurora MySQL versi 2.07.4	Aurora MySQL versi 2.07.4 tersedia.	4 Maret 2021
Aurora MySQL versi 1.22.4	Aurora MySQL versi 1.22.4 tersedia.	4 Maret 2021
Aurora MySQL versi 2.09.2	Aurora MySQL versi 2.09.2 tersedia.	26 Februari 2021
Aurora MySQL versi 2.09.1	Aurora MySQL versi 2.09.1 tersedia.	11 Desember 2020
Aurora MySQL versi 1.23.1	Aurora MySQL versi 1.23.1 tersedia.	24 November 2020
Aurora MySQL versi 2.08.3	Aurora MySQL versi 2.08.3 tersedia.	12 November 2020
Aurora MySQL versi 2.07.3	Aurora MySQL versi 2.07.3 tersedia.	10 November 2020
Aurora MySQL versi 1.22.3	Aurora MySQL versi 1.22.3 tersedia.	9 November 2020
Aurora MySQL versi 2.09.0	Aurora MySQL versi 2.09.0 tersedia.	17 September 2020
Aurora MySQL versi 1.23.0	Aurora MySQL versi 1.23.0 tersedia.	2 September 2020
Aurora MySQL versi 2.08.2	Aurora MySQL versi 2.08.2 tersedia.	28 Agustus 2020
Aurora MySQL versi 2.04.9	Aurora MySQL versi 2.04.9 tersedia.	14 Agustus 2020

Aurora MySQL versi 2.08.1	Aurora MySQL versi 2.08.1 tersedia.	18 Juni 2020
Aurora MySQL versi 1.22.2 untuk klaster kueri paralel	Aurora MySQL versi 1.22.2 tersedia saat Anda membuat klaster kueri paralel.	18 Juni 2020
Aurora MySQL versi 1.20.1 untuk klaster kueri paralel	Aurora MySQL versi 1.20.1 tersedia saat Anda membuat klaster kueri paralel.	11 Juni 2020
Aurora MySQL versi 2.08.0	Aurora MySQL versi 2.08.0 tersedia.	2 Juni 2020
Aurora MySQL versi 1.19.6 untuk klaster kueri paralel	Aurora MySQL versi 1.19.6 tersedia saat Anda membuat klaster kueri paralel.	2 Juni 2020
Aurora MySQL versi 2.07.2	Aurora MySQL versi 2.07.2 tersedia.	17 April 2020
Aurora MySQL versi 1.22.2	Aurora MySQL versi 1.22.2 tersedia.	5 Maret 2020
Aurora MySQL versi 1.20.1	Aurora MySQL versi 1.20.1 tersedia.	5 Maret 2020
Aurora MySQL versi 1.19.6	Aurora MySQL versi 1.19.6 tersedia.	5 Maret 2020
Aurora MySQL versi 1.17.9	Aurora MySQL versi 1.17.9 tersedia.	5 Maret 2020
Aurora MySQL versi 2.07.1	Aurora MySQL versi 2.07.1 tersedia.	23 Desember 2019
Aurora MySQL versi 1.22.1	Aurora MySQL versi 1.22.1 tersedia.	23 Desember 2019

Aurora MySQL versi 2.07.0	Aurora MySQL versi 2.07.0 tersedia.	25 November 2019
Aurora MySQL versi 1.22.0	Aurora MySQL versi 1.22.0 tersedia.	25 November 2019
Aurora MySQL versi 1.21.0	Aurora MySQL versi 1.21.0 tersedia.	25 November 2019
Aurora MySQL versi 2.06.0	Aurora MySQL versi 2.06.0 tersedia.	22 November 2019
Aurora MySQL versi 2.04.8	Aurora MySQL versi 2.04.8 tersedia.	20 November 2019
Aurora MySQL versi 2.04.7	Aurora MySQL versi 2.04.7 tersedia.	14 November 2019
Aurora MySQL versi 2.05.0	Aurora MySQL versi 2.05.0 tersedia.	11 November 2019
Aurora MySQL versi 1.20.0	Aurora MySQL versi 1.20.0 tersedia.	11 November 2019
Aurora MySQL versi 2.04.6	Aurora MySQL versi 2.04.6 tersedia.	19 September 2019
Aurora MySQL versi 1.19.5	Aurora MySQL versi 1.19.5 tersedia.	19 September 2019
Aurora MySQL versi 2.04.5	Aurora MySQL versi 2.04.5 tersedia.	8 Juli 2019
Aurora MySQL versi 1.19.2	Aurora MySQL versi 1.19.2 tersedia.	5 Juni 2019
Aurora MySQL versi 2.04.4	Aurora MySQL versi 2.04.4 tersedia.	29 Mei 2019

Aurora MySQL versi 2.04.3	Aurora MySQL versi 2.04.3 tersedia.	9 Mei 2019
Aurora MySQL versi 1.19.1	Aurora MySQL versi 1.19.1 tersedia.	9 Mei 2019
Aurora MySQL versi 2.04.2	Aurora MySQL versi 2.04.2 tersedia.	2 Mei 2019
Aurora MySQL versi 2.04.1	Aurora MySQL versi 2.04.1 tersedia.	25 Maret 2019
Aurora MySQL versi 2.04	Aurora MySQL versi 2.04 tersedia.	25 Maret 2019
Aurora MySQL versi 2.03.4	Aurora MySQL versi 2.03.4 tersedia.	7 Februari 2019
Aurora MySQL versi 1.19.0	Aurora MySQL versi 1.19.0 tersedia.	7 Februari 2019
Aurora MySQL versi 2.03.3	Aurora MySQL versi 2.03.3 tersedia.	18 Januari 2019
Aurora MySQL versi 1.17.8	Aurora MySQL versi 1.17.8 tersedia.	17 Januari 2019
Aurora MySQL versi 2.03.2	Aurora MySQL versi 2.03.2 tersedia.	9 Januari 2019
Aurora MySQL versi 2.03.1	Aurora MySQL versi 2.03.1 tersedia.	24 Oktober 2018
Aurora MySQL versi 2.03	Aurora MySQL versi 2.03 tersedia.	11 Oktober 2018
Aurora MySQL versi 2.02.5	Aurora MySQL versi 2.02.5 tersedia.	8 Oktober 2018

Aurora MySQL versi 1.17.7	Aurora MySQL versi 1.17.7 tersedia.	8 Oktober 2018
Aurora MySQL versi 2.02.4	Aurora MySQL versi 2.02.4 tersedia.	21 September 2018
Aurora MySQL versi 1.18.0	Aurora MySQL versi 1.18.0 tersedia.	20 September 2018
Aurora MySQL versi 1.17.6	Aurora MySQL versi 1.17.6 tersedia.	6 September 2018

Terjemahan disediakan oleh mesin penerjemah. Jika konten terjemahan yang diberikan bertentangan dengan versi bahasa Inggris aslinya, utamakan versi bahasa Inggris.