

第八节：组复制系统变量---基础篇

1.概述：

用于组复制的大多数系统变量是可以动态修改其值的，并且在集群运行时可以更改它们的值。但是，在大多数情况下，变更仅在组成员上执行STOP GROUP_REPLICATION语句和START GROUP_REPLICATION语句重新启动组复制后才生效。而一小部分系统变量，修改它们的值会立即生效，不需要重启组复制，这些系统变量与流控相关，对以下系统变量的更改将在不重新启动组复制的情况下生效：

```
group_replication_advertise_recovery_endpoints  
  
group_replication_autorejoin_tries  
  
group_replication_consistency  
  
group_replication_exit_state_action  
  
group_replication_flow_control_applier_threshold  
  
group_replication_flow_control_certifier_threshold  
  
group_replication_flow_control_hold_percent  
  
group_replication_flow_control_max_commit_quota  
  
group_replication_flow_control_member_quota_percent  
  
group_replication_flow_control_min_quota  
  
group_replication_flow_control_min_recovery_quota  
  
group_replication_flow_control_mode  
  
group_replication_flow_control_period  
  
group_replication_flow_control_release_percent  
  
group_replication_force_members  
  
group_replication_member_expel_timeout  
  
group_replication_member_weight  
  
group_replication_transaction_size_limit  
  
group_replication_unreachable_majority_timeout
```

组复制的大多数系统变量在不同的组成员上可以具有不同的值。对于以下系统变量，建议在集群的所有节点上设置相同的值，以避免不必要的事务回滚，消息传递失败或消息恢复失败：

```
group_replication_auto_increment_increment  
  
group_replication_communication_max_message_size  
  
group_replication_compression_threshold  
  
group_replication_message_cache_size  
  
group_replication_transaction_size_limit
```

组复制节点上的某些系统变量（包括某些特定于组复制的系统变量和一些常规系统变量）是集群范围的配置设置。这些系统变量在集群所有节点上必须具有相同的值，并且在组复制运行时不能更改，需要先停止组复制，然后进行修改，修改了这些系统变量的值之后，还必须将整个集群重新引导才会生效，而不仅仅是重启某个节点上的组复制（这里不是指重新启动整个集群的mysqld进程，而是整个集群重新引导启动，即，使用系统变量group_replication_bootstrap_group= on重新引导集群）。这些系统变量如下：

```
group_replication_single_primary_mode  
  
group_replication_enforce_update_everywhere_checks  
  
group_replication_gtid_assignment_block_size  
  
default_table_encryption  
  
lower_case_table_names  
  
transaction_write_set_extraction
```

从MySQL 8.0.16开始，可以在集群仍在运行时使用group_replication_switch_to_single_primary_mode () 和group_replication_switch_to_multi_primary_mode () UDF来更改group_replication_single_primary_mode和group_replication_enforce_update_everywhere_checks的值。

ps:如果将组复制的许多系统变量作为命令行参数传递给server，则它们在server启动期间将无法完全验证。这些系统变量包括group_replication_group_name, group_replication_single_primary_mode, group_replication_force_members, SSL变量和流控系统变量。

在执行START GROUP_REPLICATION语句之前，不会对组复制的系统变量指定的IP地址、主机名等进行验证。因为在未启动组复制之前，组复制的组通信系统(GCS)不可用。

组复制的状态变量在MySQL 5.7版本中只有3个，在MySQL 8.0版本中只有4个，其中状态变量group_replication_primary_member在8.0版本中已经弃用，在将来的版本中将会移除（该状态变量在集群处于单主模式时，表示primary节点的UUID，集群处于多主模式时，该状态变量值是一个空字符串）

* Mysqlx_notified_by_group_replication：MySQL 8.0中新增的全局状态变量，表示发送到集群的通知数量

* Com_group_replication_start：全局，会话状态变量，表示执行start group_replication语句的次数

* Com_group_replication_stop：全局，会话状态变量，表示执行stop group_replication语句的次数

2.MGR系统变量：

特定用于组复制插件的系统变量如下：

```
group_replication_advertise_recovery_endpoints:
```

```
Command-Line Format --group-replication-advertise-recovery-endpoints=value
```

```
Introduced 8.0.21
```

```
System Variable group_replication_advertise_recovery_endpoints
```

```
Scope global
```

```
Dynamic Yes
```

```
SET_VAR Hint Applies No
```

```
Type String
```

```
Default Value DEFAULT
```

组复制运行时可以更改此系统变量的值。更改将立即对节点生效。但是已经收到系统变量的先前值的加入节点将继续使用该值。只有在值更改后加入的节点才能收到新值。

group_replication_advertise_recovery_endpoints指定加入节点如何建立与现有节点的连接以进行状态转移和分布式恢复。该连接用于远程克隆操作和从引导节点的二进制日志进行状态转移。

默认值DEFAULT是默认设置，表示加入节点使用集群现有节点的标准SQL客户端连接，该连接由MySQL Server的主机名和端口系统变量指定。如果report_port系统变量指定了备用端口号，则改用该端口号。performance.Replication_group_members表在MEMBER_HOST和MEMBER_PORT字段中显示此连接的地址和端口号。这在MySQL 8.0.20（含）以下版本中的集群节点中有效。

可以指定一个或多个分布式恢复端点，而不是DEFAULT，现有节点将这些端点通告给加入的节点供他们使用。通过提供分布式恢复端点，管理员可以独立于与组成员的常规MySQL客户端连接来控制分布式恢复流量。加入节点按列表中指定的顺序依次尝试每个端点。

将分布式恢复端点指定为IP地址和端口号的逗号分隔列表，例如：

```
group_replication_advertise_recovery_endpoints= "127.0.0.1:3306,127.0.0.1:4567,  
[::1]:3306,localhost:3306"
```

IPv4和IPv6地址以及主机名可以任意组合使用。IPv6地址必须在方括号中指定。主机名必须解析为本地IP地址。无法使用通配符地址格式，并且不能指定空列表。标准SQL客户端连接不会自动包含在分布式恢复端点列表中。如果要将其用作端点，则必须在列表中明确包含它。

有关如何选择IP地址和端口作为分布式恢复终点以及加入节点如何使用它们的详细信息，摘要如下：

IP地址不必为MySQL Server配置，但必须将其分配给server。

必须使用port，report_port或admin_port系统变量为MySQL Server配置端口。

如果使用admin_port，则复制用户需要适当的权限才能进行分布式恢复。

IP地址不需要添加到由group_replication_ip_allowlist或group_replication_ip_whitelist系统变量指定的组复制白名单中。

连接的SSL要求由group_replication_recovery_ssl_*变量指定。

```
group_replication_allow_local_lower_version_join
Command-Line Format --group-replication-allow-local-lower-version-join[={OFF|ON}]
System Variable group_replication_allow_local_lower_version_join
Scope      Global
Dynamic     Yes
SET_VAR     Hint Applies      No
Type        Boolean
Default Value  OFF
```

可以在组复制运行时更改此系统变量的值，但是该更改仅在停止并重新启动集群上的组复制后才生效。

group_replication_allow_local_lower_version_join允许当前server加入该集群，即使它运行的MySQL Server版本低于该集群内某一节点的最低MySQL版本。默认设置OFF，如果该server运行的MySQL版本比现有组成员的版本低，则不允许它们加入复制组。此标准策略确保集群中的所有节点之间都能够正常交换消息和应用事务。运行MySQL 8.0.17或更高版本的节点在检查其兼容性时会考虑该发行版的补丁版本。运行MySQL 8.0.16或更低版本或MySQL 5.7的节点仅考虑主要版本。

仅在以下情况下，将group_replication_allow_local_lower_version_join设置为ON：

必须在紧急情况下将MySQL server添加到集群中，以提高集群的容错能力，并且仅旧版本可用。

想回滚一个或多个节点的升级，而无需关闭整个集群并重新引导集群。

ps：将此选项设置为ON不会使新节点与集群兼容，并允许其加入集群而没有任何措施防止现有节点的不兼容行为。为了确保新节点的正确操作，采取以下两项预防措施：

在运行较低版本的MySQL server加入该集群之前，停止该节点上的所有写操作。

从运行较低版本的server加入集群的位置开始，停止对集群中其他节点的所有写入操作。

如果没有这些预防措施，运行较低版本的节点可能会遇到困难，并因错误而退出集群。

```
group_replication_auto_increment_increment
Command-Line Format --group-replication-auto-increment-increment=#
System Variable group_replication_auto_increment_increment
Scope      Global
Dynamic     Yes
SET_VAR     Hint Applies      No
Type        Integer
Default Value  7
Minimum Value  1
Maximum Value  65535
```

该系统变量在所有组成员上应具有相同的值。组复制运行时，不能更改此系统变量的值。必须停止组复制，更改系统变量的值，然后在每个组成员上重新启动组复制。

自动设置复制组中的每个节点的自增列的步长值，以确保在多主模式的集群中，每个组成员的自增列值有序且不重叠。

该系统变量的值在所有组成员上必须相同，一旦在某个Server中启动组复制，则该Server的普通系统变量auto_increment_increment将会被设置为组系统变量

group_replication_auto_increment_increment的值，将普通系统变量auto_increment_offset设置为Server id值。通过这些设置调整可以避免在多节点写的情况下不同组成员使用相同的自增值导致事务发生冲突回滚。

- 当节点停止组复制时，普通系统变量auto_increment_increment和auto_increment_offset的值将会恢复原状（启动组复制之前的值）

- 只有当系统变量`auto_increment_increment`和`auto_increment_offset`保持默认值时，组复制启动时才会自动做与组复制的适配调整和恢复，如果这两个系统变量的值被设置了非默认值，则组复制不会做自动调整（从MySQL 8.0开始，当集群处于单主模式下时，这两个系统变量也不会做自动调整）。所以，对于这两个系统变量的值，要么在组复制下不对其进行手工指定（让其使用默认值），要么就一定要设置正确，否则，在多主模式的集群中，很容易造成主键冲突。
- 系统变量`group_replication_auto_increment_increment`在组复制运行时无法修改，需要先停止组复制，修改该系统变量的值，然后再启动组复制。

如果你的集群中有更多或更少的组成员，则可以在组复制启动之前调整好该系统变量的值，以对应你的集群中的节点数量。

```
group_replication_autorejoin_tries
Command-Line Format --group-replication-autorejoin-tries=#
Introduced 8.0.16
System Variable group_replication_autorejoin_tries
Scope Global
Dynamic Yes
SET_VAR Hint Applies No
Type Integer
Default Value (≥ 8.0.21) 3
Default Value (≤ 8.0.20) 0
Minimum Value 0
Maximum Value 2016
```

可以在组复制运行时更改此系统变量的值，该更改将立即生效。发生问题（表示需要该行为）时，将读取系统变量的当前值。

`group_replication_autorejoin_tries`指定在达到`group_replication_unreachable_majority_timeout`设置之前，如果节点被驱逐或无法联系该集群的大多数节点，则尝试进行自动重新加入该集群的次数。当节点被驱逐或多数节点无法达到超时，它将尝试重新加入（使用当前插件选项值），然后继续进行进一步的自动重新加入尝试，直至达到指定的尝试次数。自动重新加入尝试失败后，节点将等待5分钟，然后再进行下一次尝试。如果指定的尝试次数已用完，而节点没有重新加入或停止，则节点将继续执行`group_replication_exit_state_action`系统变量指定的操作。

在MySQL 8.0.20之前，默认设置为0，意味着该节点不会尝试自动重新加入。从MySQL 8.0.21开始，默认设置为3，这意味着该节点自动进行3次尝试重新加入该集群的尝试，每次间隔5分钟。最多可以指定2016次尝试。

在自动重新连接尝试期间，节点保持超级只读模式，并且不接受写入，但是仍然可以在节点上进行读取，随着时间的推移，读取到旧数据的可能性越来越高。如果在任何时间段都无法忍受旧数据的读取，将`group_replication_autorejoin_tries`设置为0。

```
group_replication_bootstrap_group
Command-Line Format --group-replication-bootstrap-group[={OFF|ON}]
System Variable group_replication_bootstrap_group
Scope Global
Dynamic Yes
SET_VAR Hint Applies No
Type Boolean
Default Value OFF
```

`group_replication_bootstrap_group`将此节点配置为引导该集群。仅在第一次启动该集群或重新启动整个集群时，才必须在一个节点上设置此系统变量。自引导集群后，将此选项设置为OFF。应同时在动态文件和配置文件中将其设置为OFF。在集群运行时启动两个节点或使用此选项重新启动一个节点可能会导致人为脑裂的情况，其中两个具有相同名称的独立集群将被引导。

```
group_replication_clone_threshold
Command-Line Format --group-replication-clone-threshold=#
Introduced 8.0.17
System Variable group_replication_clone_threshold
Scope Global
Dynamic Yes
SET_VAR Hint Applies No
Type Integer
Default Value 9223372036854775807
Minimum Value 1
Maximum Value 9223372036854775807
```

可以在组复制运行时更改此系统变量的值，但是该更改仅在停止并重新启动组成员上的组复制后才生效。

group_replication_clone_threshold指定现有节点和加入节点之间的事务间隔（以事务数量为单位），从而触发在分布式恢复过程中使用远程克隆操作将状态转移到加入节点。如果加入节点与集群的引导节点之间的事务间隔超过阈值，则组复制将通过远程克隆操作开始分布式恢复。如果事务间隔低于阈值，或者在技术上无法进行远程克隆操作，则组复制将直接从引导节点的二进制日志进行状态转移。

ps:不要在活跃的集群中为group_replication_clone_threshold使用较低的设置。如果在进行远程克隆操作的同时在集群中发生了超过阈值的事务，则加入节点在重新启动后会再次触发远程克隆操作，并且可以无限期地继续进行。为避免这种情况，请确保将阈值设置为一个数字，该阈值应大于在远程克隆操作所花费的时间段内集群中预期发生的事务数。

要使用此功能，必须预先设置引导节点和加入节点以支持克隆。当远程克隆操作执行完成之后，组复制将会自动管理新加入集群的节点，例如：重启加入节点的Server进程，并将设置系统变量group_replication_start_on_boot=ON，不需要人工干预。但如果不使用远程克隆操作，则必须手动重启加入节点的Server进程。另外，由于远程克隆操作将会替换加入节点上现有的数据字典，所以组复制会检查加入节点中是否存在集群中没有的事务，如果存在，则分布式恢复操作将报错终止，因为这些额外的事务将会被远程克隆操作删除（存在丢失数据的风险，因此需要先自行处理这些额外的事务）。

默认设置为 9223372036854775807，这是GTID中允许的事务最大序列号（该数值也是计算机中最大的64位有符号的整型值，即， $2^{63} - 1$ ），这意味着从引导节点到加入节点之间的分布式恢复总是会优先尝试使用基于二进制日志的状态传输，而不是基于克隆操作的状态传输。但是如果无法使用引导节点的二进制日志进行状态传输，则无论阈值如何设置，组复制总是会尝试执行基于远程克隆操作的状态传输。因为加入节点所需的事务在集群中任何现有节点的二进制日志中都不可用（无法找到）。如果不想在组复制中使用克隆功能，则不要在任何Server上安装克隆插件。

```
group_replication_communication_debug_options

Command-Line Format --group-replication-communication-debug-options=value
System Variable group_replication_communication_debug_options
Scope Global
Dynamic Yes
SET_VAR Hint Applies No
Type String
Default Value GCS_DEBUG_NONE
Valid Values
GCS_DEBUG_NONE

GCS_DEBUG_BASIC

GCS_DEBUG_TRACE

XCOM_DEBUG_BASIC
```



```
XCOM_DEBUG_TRACE
```

```
GCS_DEBUG_ALL
```

可以在运行组复制时更改系统变量的值，但是该更改仅在停止并重新启动组成员上的组复制后才生效。

group_replication_communication_debug_options配置Debug级别的调试信息，提供给不同的组复制组件，例如组通信系统（GCS）和组通信引擎（XCom，Paxos变体）。调试信息存储在数据目录的GCS_DEBUG_TRACE文件中。

可以指定为组合如下字符串的一组可用选项。提供以下选项：

GCS_DEBUG_NONE:禁用GCS和XCom的所有调试级别。

GCS_DEBUG_BASIC:在GCS中启用基本调试信息。

GCS_DEBUG_TRACE:启用GCS中的跟踪信息。

XCOM_DEBUG_BASIC:启用XCom中的基本调试信息。

XCOM_DEBUG_TRACE:启用XCom中的跟踪信息。

GCS_DEBUG_ALL:启用GCS和XCom的所有调试级别。

对于系统变量group_replication_communication_debug_options的设置，可以同时设置多个调试级别值，多个值之间使用逗号分隔。当设置调试级别GCS_DEBUG_NONE时，只有不带其他调试级别选项值时GCS_DEBUG_NONE才会生效，如果有其他更高级别的值，则，其他调试级别的设置值生效（可以多个调试级别同时生效）。但是，如果将调试级别设置为GCS_DEBUG_ALL，则该设置值将覆盖所有其他值（只剩下这一个值）。

```
group_replication_communication_max_message_size
```

```
Command-Line Format --group-replication-communication-max-message-size=#
```

```
Introduced 8.0.16
```

```
System Variable group_replication_communication_max_message_size
```

```
Scope global
```

```
Dynamic Yes
```

```
SET_VAR Hint Applies No
```

```
Type Integer
```

```
Default value 10485760
```

```
Minimum value 0
```

```
Maximum value 1073741824
```

该系统变量在所有组成员上应具有相同的值。组复制运行时，不能更改此系统变量的值。必须停止组复制，更改系统变量的值，然后在每个组成员上重新启动组复制。在此过程中，组成员之间的系统变量值可以不同，但可能会回滚组成员上的某些事务。

group_replication_communication_max_message_size指定用于组复制通信的最大消息大小。大于此大小的消息将自动拆分为多个片段，这些片段将分别发送并由接收端重新组合。

默认情况下，最大消息大小设置为10485760字节（10 MiB），这意味着默认情况下，MySQL 8.0.16发行版中使用碎片。允许的最大值与slave_max_allowed_packet系统变量的最大值相同，为1073741824字节（1 GB）。group_replication_communication_max_message_size的设置必须小于slave_max_allowed_packet设置，因为应用程序线程无法处理大于slave_max_allowed_packet的消息片段。要关闭分段，为group_replication_communication_max_message_size指定零值。

ps:系统变量group_replication_communication_max_message_size的值需要在集群中所有组成员中设置为相同的值。

为了使复制组的节点使用分段，该集群的通信协议版本必须为MySQL 8.0.16或更高版本。使用 `group_replication_get_communication_protocol()` UDF查看集群的通信协议版本。如果使用的是较低版本，则组成员不对消息进行分段。如果所有组成员都支持，则可以使用 `group_replication_set_communication_protocol()` UDF将组的通信协议设置为更高版本。

```
group_replication_components_stop_timeout

Command-Line Format --group-replication-components-stop-timeout=#
System Variable group_replication_components_stop_timeout
Scope      Global
Dynamic    Yes
SET_VAR Hint Applies      No
Type       Integer
Default Value      31536000
Minimum Value      2
Maximum Value      31536000
```

在运行组复制时可以更改此系统变量的值，但是该更改仅在停止并重新启动组成员上的组复制后才生效。

`group_replication_components_stop_timeout`指定关闭时组复制等待每个组件的超时时间（以秒为单位）。

```
group_replication_compression_threshold

Command-Line Format --group-replication-compression-threshold=#
System Variable group_replication_compression_threshold
Scope      Global
Dynamic    Yes
SET_VAR Hint Applies      No
Type       Integer
Default Value      1000000
Minimum Value      0
Maximum Value      4294967295
```

该系统变量在所有组成员上应具有相同的值。组复制运行时可以更改此系统变量的值。在停止并重新启动成员上的组复制后，更改将对每个组成员生效。在此过程中，允许组成员之间的系统变量值不同，但是消息传递在所有成员上的效率并不相同。

`group_replication_compression_threshold`指定阈值（以字节为单位），在该阈值之上，压缩将应用于组成员之间发送的消息。如果将此系统变量设置为零，则禁用压缩。

组复制使用LZ4压缩算法来压缩在集群中发送的消息。LZ4压缩算法支持的最大输入大小为2113929216字节。此限制低于`group_replication_compression_threshold`系统变量的最大可能值，该值与XCom接受的最大消息大小匹配。使用LZ4压缩算法时，不要为`group_replication_compression_threshold`设置大于2113929216字节的值，因为在启用消息压缩时无法提交超过此大小的事务。

```
group_replication_consistency

Command-Line Format --group-replication-consistency=value
Introduced      8.0.14
System Variable group_replication_consistency
Scope          Global, Session
Dynamic        Yes
SET_VAR Hint Applies      No
Type           Enumeration
```



```
Default Value    EVENTUAL
Valid Values
EVENTUAL

BEFORE_ON_PRIMARY_FAILOVER

BEFORE

AFTER

BEFORE_AND_AFTER
```

组复制运行时可以更改此系统变量的值。group_replication_consistency是mysql系统变量，而不是特定于Group Replication插件的变量，因此，不需要重新启动Group Replication即可使更改生效。更改系统变量的会话值将立即生效，更改全局值将对更改后开始的新会话生效。需要GROUP_REPLICATION_ADMIN特权才能更改此系统变量的全局设置。

group_replication_consistency控制集群提供的事务一致性保证。可以全局或按事务配置一致性。对于只读（RO）和读写（RW）事务，都必须考虑该变量的影响。以下列表显示了此变量的取值，以提高事务一致性保证的顺序：

EVENTUAL

RO和RW事务在执行之前都不会等待前面的事务应用完成（即，事务直接执行，不等待积压事务应用完成）。这是系统变量group_replication_consistency的默认值（也是引入该系统变量之前组复制的默认行为）。RW事务不等待其他节点应用事务。意味着设置该值节点中的事务可以先于其他节点外部化。还意味着，在主节点发生故障转移时，新的主节点不需要等待积压的事务（来自旧的主节点的事务），允许立即接受新的RO和RW事务，这可能造成新的RO事务读取到陈旧的数据（因为之前旧主节点中的数据还未完全同步到新的主节点）、新的RW事务可能由于冲突导致回滚（冲突认证检测会发现新的RW事务可能与来自旧主节点的积压RW事务发生冲突）。

BEFORE_ON_PRIMARY_FAILOVER

新的RO或RW事务在新的主节点应用完成来自旧的主节点的积压事务之前，会被保持（不应用，类似于处在等待状态，积压事务被应用完成之后，才会处理新的RO和RW事务）。这能够确保当主节点故障转移发生时，客户端总是能查询到发生故障的主节点上的最新值，从而保证了一致性。但这意味着客户端需要自行处理新的主节点在应用积压事务过程中的延迟（这里指的是客户端访问新主要节点的响应延迟）。通常这种延迟很小，但是实际延迟时间的长短取决于积压事务的大小。

BEFORE

RW事务在应用(applied)之前会等待所有前面的事务（积压事务）完成。RO事务在执行(executed)之前会等待所有前面的事务（积压事务）完成。这样使得事务仅通过牺牲响应延迟就可以确保读取到最新的值。实际上，只是确保了RO事务上的同步，对于RW事务来说，只是等待了它之前积压的事务完成，并不会等待它自身发起的RW事务在所有的其他组成员上完成应用（不过，由于RO事务要求同步，RO事务能够将一部分甚至大部分数据进行同步，所以能够一定程度上减少RW事务上的同步开销，也就是说，该一致性级别适合于写多读少的场景）。对于BEFORE这个一致性级别，它涵盖了BEFORE_ON_PRIMARY_FAILOVER提供的一致性保证。

AFTER

RW事务会等待它的更改应用到集群中所有其他节点。此一致性级别对RO事务没有影响（因为RO事务不会产生数据变更）。它只确保在本地成员上提交RW事务时，该RW事务的数据变更会在集群中其他所有节点中应用，以便所有后续的事务在任何节点上都能够获取到最新的数据（通过确保只在RW事务上使用同步，RW事务会将所有写入的新数据都实时同步到集群中其他的所有节点中，这就减少了RO事务上的同步开销。也就是说，该一致性级别比较适合读多写少的场景）。对于AFTER这个一致性级别，它涵盖了BEFORE_ON_PRIMARY_FAILOVER提供的一致性保证。

BEFORE_AND_AFTER

一致性级别要求最高，RW和RO事务执行时都要求数据同步，RW事务在执行时需要等待之前的积压事务应用完成，且需要等待自己的数据变更在其他所有组成员上都应用。RO事务在执行时需要等待之前的积压事务应用完成。该一致性级别适合对数据的读写一致性都要求高的场景。

group_replication_enforce_update_everywhere_checks

```
Command-Line Format --group-replication-enforce-update-everywhere-checks[={OFF|ON}]
System Variable group_replication_enforce_update_everywhere_checks
Scope      global
Dynamic    Yes
SET_VAR Hint Applies      No
Type       Boolean
Default value      OFF
```

此系统变量是集群范围的配置设置。必须在所有组成员上具有相同的值，并且在组复制运行时不能更改，需要完全重新引导集群（由具有group_replication_bootstrap_group = ON的节点进行引导）才能使值更改生效。

从MySQL 8.0.16开始，可以在集群仍在运行时使用

group_replication_switch_to_single_primary_mode () 和

group_replication_switch_to_multi_primary_mode () UDF更改此系统变量的值。

group_replication_enforce_update_everywhere_checks启用或禁用对多主模式运行的集群更新的严格一致性检查。默认设置是禁用检查。在单主模式下，必须在所有组成员上禁用此变量。在多主模式下，启用此选项后，将按以下方式检查语句以确保它们与多主模式兼容：

如果事务是在SERIALIZABLE隔离级别下执行的，则在与集群同步时，该事务将提交失败。

如果事务是针对具有级联约束的外键的表执行的，则该事务在与集群同步时将无法提交。

group_replication_exit_state_action

```
Command-Line Format --group-replication-exit-state-action=value
Introduced      8.0.12
System Variable group_replication_exit_state_action
Scope          global
Dynamic        Yes
SET_VAR Hint Applies      No
Type           Enumeration
Default value (>= 8.0.16)      READ_ONLY
Default value (>= 8.0.12, <= 8.0.15)  ABORT_SERVER
Valid values (>= 8.0.18)
ABORT_SERVER

OFFLINE_MODE

READ_ONLY

Valid values (>= 8.0.12, <= 8.0.17)
ABORT_SERVER

READ_ONLY
```

可以在组复制运行时更改此系统变量的值，该更改将立即生效。发生问题（表示需要该行为）时，将读取系统变量的当前值。

group_replication_exit_state_action配置当此实例意外离开集群时（例如，在遇到应用程序错误后，或者在失去多数节点的情况下，或者由于怀疑超时而导致该集群的另一个节点驱逐它时），组复制的行为方式。

与失多数节点失联的情况下，离开集群的超时时间由

group_replication_unreachable_majority_timeout系统变量设置，被怀疑离开集群的超时时间由group_replication_member_expel_timeout系统变量设置。被驱逐的组成员直到重新连接到该集群才知道被驱逐了，因此仅当该节点重新连接时，或者该节点对自己提出怀疑并驱逐自己时，才执行指定的操作。

当组成员由于怀疑超时或失去多数节点而被驱逐时，如果该节点的group_replication_autorejoin_tries系统变量设置为指定了自动重新加入尝试的次数，则在超级只读时，它首先进行指定次数的尝试，然后执行group_replication_exit_state_action指定的操作。如果发生应用程序错误，则不会进行自动重新尝试，因为这些尝试不可恢复。

当group_replication_exit_state_action设置为READ_ONLY时，如果节点意外退出该集群或用尽了自动重试尝试次数，则该实例将MySQL切换为超级只读模式（通过将系统变量super_read_only设置为ON）。READ_ONLY退出动作是在引入该系统变量之前MySQL 8.0发行版的行为，并再次成为MySQL 8.0.16的默认行为。

当group_replication_exit_state_action设置为OFFLINE_MODE时，如果节点意外退出该集群或用尽其自动重新加入尝试次数，则该实例将MySQL切换为离线模式（通过将系统变量offline_mode设置为ON）。在这种模式下，已连接的客户端用户将在其下一个请求时断开连接，并且不再接受连接，但具有CONNECTION_ADMIN特权（或不建议使用的SUPER特权）的客户端用户除外。组复制还将系统变量super_read_only设置为ON，因此即使客户端已使用CONNECTION_ADMIN或SUPER特权进行连接，客户端也无法进行任何写入操作。MySQL 8.0.18提供了OFFLINE_MODE退出操作。

当group_replication_exit_state_action设置为ABORT_SERVER时，如果节点意外退出了该集群或用尽了自动重新加入尝试次数，则该实例将关闭MySQL。在将系统变量添加到MySQL 8.0.15（含）时，此设置是MySQL 8.0.12的默认设置。

ps:如果在节点成功加入集群之前发生故障，则不会执行指定的退出操作。如果在本地配置检查期间失败，或者加入节点的配置和集群的配置不匹配，就是这种情况。在这种情况下，系统变量super_read_only保留其原始值，继续接受连接，并且实例不会关闭MySQL。为确保当组复制未启动时实例无法接受更新，因此我们建议在启动时在实例的配置文件中将super_read_only = ON设置为成功后，primary节点上的组复制将变为OFF。当实例配置为在引导时启动组复制

（group_replication_start_on_boot = ON）时，此保护措施尤其重要，但是在使用START GROUP_REPLICATION命令手动启动组复制时，此保护措施也很有用。

```
group_replication_flow_control_applier_threshold
```

```
Command-Line Format --group-replication-flow-control-applier-threshold=#
```

```
System Variable group_replication_flow_control_applier_threshold
```

```
Scope      Global
```

```
Dynamic Yes
```

```
SET_VAR Hint Applies      No
```

```
Type      Integer
```

```
Default Value      25000
```

```
Minimum Value      0
```

```
Maximum Value      2147483647
```

可以在组复制运行时更改此系统变量的值，该更改将立即生效。

group_replication_flow_control_applier_threshold指定在应用程序队列中触发流控制的等待事务数。

```
group_replication_flow_control_certifier_threshold

Command-Line Format --group-replication-flow-control-certifier-threshold=#
System Variable group_replication_flow_control_certifier_threshold
Scope      Global
Dynamic    Yes
SET_VAR    Hint Applies    No
Type       Integer
Default Value 25000
Minimum Value 0
Maximum Value 2147483647
```

可以在组复制运行时更改此系统变量的值，该更改将立即生效。

group_replication_flow_control_certifier_threshold指定认证队列中触发流控制的等待事务数。

```
group_replication_flow_control_hold_percent

Command-Line Format --group-replication-flow-control-hold-percent=#
System Variable group_replication_flow_control_hold_percent
Scope      Global
Dynamic    Yes
SET_VAR    Hint Applies    No
Type       Integer
Default Value 10
Minimum Value 0
Maximum Value 100
```

可以在组复制运行时更改此系统变量的值，该更改将立即生效。

group_replication_flow_control_hold_percent定义不使用的集群配额百分比（不用于对外提供服务，而专用于组成员追赶积压事务的组配额），以便处于流量控制下的组成员能够赶上积压的事务。如果值为0，则表示不为组成员预留用于追赶积压事务的配额。

PS：如果正常运行的集群中，有组成员发生crash并重新加入集群，或者有新的Server加入集群，则其成功加入集群之后跟集群中的现有节点之间可能存在较大的数据延迟，此时，如果集群中的负载比较高，就可能导致新加入的Server无法追赶上集群中的最新数据。所以，实际情况下，默认为组成员预留了10%的集群配额（即，节点对外提供服务的能力保留了10%），用于组成员存在数据延迟时追赶集群中的最新数据。

```
group_replication_flow_control_max_commit_quota

Command-Line Format --group-replication-flow-control-max-commit-quota=#
System Variable group_replication_flow_control_max_commit_quota
Scope      Global
Dynamic    Yes
SET_VAR    Hint Applies    No
Type       Integer
Default Value 0
Minimum Value 0
Maximum Value 2147483647
```

可以在组复制运行时更改此系统变量的值，该更改将立即生效。

group_replication_flow_control_max_commit_quota定义集群的最大流量控制配额，或者在启用流量控制的任何时间段内的最大可用配额。值为0表示没有设置最大配额。该系统变量的值不能小于group_replication_flow_control_min_quota和group_replication_flow_control_min_recovery_quota。

PS：当该系统变量设置为非0值时。如果设置的非0值小于流量控制机制计算所得的总配额，则该系统变量生效（该系统变量设置的值作为总配额，此时总配额不受系统变量group_replication_flow_control_hold_percent约束），否则，该系统变量设置的值失效。也就是说，以较小的值为准。

```
group_replication_flow_control_member_quota_percent

Command-Line Format --group-replication-flow-control-member-quota-percent=#
System Variable group_replication_flow_control_member_quota_percent
Scope      Global
Dynamic    Yes
SET_VAR Hint Applies      No
Type       Integer
Default Value      0
Minimum Value      0
Maximum Value     100
```

可以在组复制运行时更改此系统变量的值，该更改将立即生效。

group_replication_flow_control_member_quota_percent定义节点在计算配额时应假定可用于自身的配额百分比。值为0表示配额应在上一个期间作为写入节点之间平均分配。

为成员计算配额时，该系统变量指定一个假定自己可用的配额百分比。如果值为0，则表示总配额应该按照上一个流量控制检查周期中存在写入操作的组成员之间平均分配（即，未参与写入操作的组成员不参与总配额的分配）；如果值大于0，则每个组成员就会按照该系统变量设置的百分比进行计算配额值。所以，该系统变量更适合在多主模式下进行设置（注意：一旦决定要设置该值为非0值，则建议在所有组成员中都分配一个非0值，且所有成员的总百分比加起来不超过100%；另外，代码层面并没有限制所有成员的百分比加起来不能超过100%，而是由用户在该系统变量的有效值范围内自己控制），在单主模式建议保持默认值，否则可能会有性能损失（在单主模式下设置该值为非0值也无效）。

```
group_replication_flow_control_min_quota

Command-Line Format --group-replication-flow-control-min-quota=#
System Variable group_replication_flow_control_min_quota
Scope      Global
Dynamic    Yes
SET_VAR Hint Applies      No
Type       Integer
Default Value      0
Minimum Value      0
Maximum Value    2147483647
```

可以在组复制运行时更改此系统变量的值，该更改将立即生效。

控制可分配给组成员的最低流量控制配额（任何情况下对应组成员的最小流量控制配额），它与上一个流量控制检查周期中所计算出的最小配额无关。值为0表示没有最小配额限制。该系统变量设置的值不能大于系统变量group_replication_flow_control_max_quota设置的值。


```
group_replication_flow_control_min_recovery_quota
```

```
Command-Line Format --group-replication-flow-control-min-recovery-quota=#
System Variable group_replication_flow_control_min_recovery_quota
Scope      global
Dynamic    Yes
SET_VAR Hint Applies      No
Type       Integer
Default Value      0
Minimum Value      0
Maximum Value      2147483647
```

可以在组复制运行时更改此系统变量的值，该更改将立即生效。

当集群中有节点正在恢复状态中时，该系统变量用于指定在流量控制中可以分配给一个节点的最低配额（当集群中有节点处于故障恢复中时，通过该系统变量为成员指定一个最小配额，可以避免集群中的正常节点出现性能大幅抖动），它与上一个流量控制检查周期中所计算出的最小配额无关。值为0表示没有最小配额限制。该系统变量的设置值不能大于系统变量group_replication_flow_control_max_quota设置的值。

PS：当系统变量group_replication_flow_control_min_quota设置为非0值时，系统变量group_replication_flow_control_min_recovery_quota设置的值无效。

```
group_replication_flow_control_mode
```

```
Command-Line Format --group-replication-flow-control-mode=value
System Variable group_replication_flow_control_mode
Scope      global
Dynamic    Yes
SET_VAR Hint Applies      No
Type       Enumeration
Default Value      QUOTA
Valid Values
DISABLED

QUOTA
```

可以在组复制运行时更改此系统变量的值，该更改将立即生效。

group_replication_flow_control_mode指定用于流控制的模式。

PS：配额就是当流量控制被触发时，集群中的所有可读写节点允许并行写入的最大事务数量。配额应该使用集群中最低服务能力的组成员所能承载的最大并行写事务能力为基准进行设置，该配额也是整个集群中的总配额，单主模式时，该配额会全部作用于单个主节点，多主模式时，该配额会在集群中上一次流量控制检查周期中检测到的存在写操作的所有主节点之间平均分配（即，多主模式下，上一次流量检查周期内不存在写操作的主节点，不会参与分配配额）。配额会通过组成员之间每秒传递的组通讯消息中携带的各节点状态信息进行计算（通过前后2个状态信息的差值可以计算出每个节点的事务处理能力），计算完成之后，组成员就会按照计算好的每个节点自己的配额来对自己进行流量控制。另外，由于组复制是以plugin的方式动态加载到MySQL Server上的，所以，流量控制起作用的地方不可能在MySQL Server，只可能在plugin内部，但应该尽可能靠近MySQL Server。所以，组复制在一个事务进入plugin但还未进行全局原子广播时就会判断是否需要对该事务采取进行流量控制。


```
group_replication_flow_control_period
```

```
Command-Line Format --group-replication-flow-control-period=#
System Variable group_replication_flow_control_period
Scope      Global
Dynamic    Yes
SET_VAR Hint Applies      No
Type       Integer
Default Value      1
Minimum Value      1
Maximum Value      60
```

可以在组复制运行时更改此系统变量的值，该更改将立即生效。

group_replication_flow_control_period定义流控制迭代之间要等待的秒数，在该迭代期间中发送流控制消息并运行流控制管理任务。

定义在流量控制的每个检查周期之间等待的秒数，即，流量控制统计的时间间隔，通过发送流控制消息来实现并进行流量控制管理任务（每个组成员将按照该系统变量设置的间隔时间定期发送自己的状态信息给其他组成员，然后，每个组成员会收集所有组成员发送的流量控制信息，进行统计与计算下一个流量控制周期中每个组成员的配额）。

PS：

- 流量控制周期越短，控制的精度就越高，性能也更加平滑。但是，如果你的应用场景中有大事务，则应该将流量控制周期调整为与大事务提交的速度相匹配（提交耗时），否则流量控制可能就无法正常发挥应有的作用。
- 当流量控制被触发时，每个组成员会根据每个流量控制检查周期中计算出的自己的配额值执行写入负载控制，但只对自身的写入负载进行控制（不会影响组中其他节点的负载），这与Galera的流控机制有所不同。

```
group_replication_flow_control_release_percent
```

```
Command-Line Format --group-replication-flow-control-release-percent=#
System Variable group_replication_flow_control_release_percent
Scope      Global
Dynamic    Yes
SET_VAR Hint Applies      No
Type       Integer
Default Value      50
Minimum Value      0
Maximum Value      1000
```

可以在组复制运行时更改此系统变量的值，该更改将立即生效。

group_replication_flow_control_release_percent定义当集群不再需要进行流量控制时，如何释放集群的配额，当流量控制退出时，也会计算每一个流量控制周期的配额，默认的策略是每个周期增加50%的配额，直到配额增加到与外部应用的实际最大负载相同（这个时候就相当于解除了流量控制了），或者增加到外部应用负载触发了新一轮的流控阈值为止。如果值为0，则意味着流量控制退出时，组复制允许外部的全部实际应用负载全部进入集群中（设置为1000时，表示允许以10倍于当前配额的速度释放），但这极有可能导致迅速触发新一轮的流量控制进而导致集群的性能急剧下降。

group_replication_force_members

```
Command-Line Format --group-replication-force-members=value
System Variable group_replication_force_members
Scope      Global
Dynamic    Yes
SET_VAR Hint Applies      No
Type       String
```

此系统变量用于强制指定新的组成员身份。可以在组复制运行时更改此系统变量的值，该更改将立即生效。只需要要保留在集群中的组成员之一上设置该系统变量的值。

以逗号分割的地址列表，例如"host1:port1,host2:port2"。此系统变量用于强制创建新的组成员资格，其中被排除的节点（未写入进该系统变量中的节点）在该变量设置成功之后就不会收到新视图消息且会被阻塞（阻塞写入操作）。然后你需要手动移除/关闭被排除在该系统变量的参数列表之外的节点。另外。如果列表中的存在任何无效主机名，则可能导致后续执行START GROUP_REPLICATION语句失败，因为这些无法通讯的Server无法响应组成员间的通讯请求。

在系统变量group_replication_force_members指定的每个Server上需要使用系统变量group_replication_local_address设置一个有效的IP地址（或主机名）与端口，如果是IPV6地址，则需要在方括号中指定，例如：

```
"198.51.100.44:33061,
[2001:db8:85a3:8d3:1319:8a2e:370:7348]:33061,example.org:33061"
```

用于组复制（XCom）的组通信引擎将检查提供的IP地址格式是否有效，并检查是否没有包括当前无法访问的任何组成员。否则，新配置将不会通过验证，因此必须谨慎配置，仅包括集群中可访问的online节点。列表中任何不正确的值或无效的主机名都可能导致该集群被无效的配置阻止。

在强制执行新的成员资格配置之前，必须确保要排除的节点已关闭，这一点很重要。如果不是，请在继续操作之前将其关闭。仍在线的组成员可以自动形成新的配置，如果已经进行过，则强制进行进一步的新配置可能会为该集群造成人为的裂脑情况。

使用group_replication_force_members系统变量成功强制新的组成员身份并取消阻止该集群后，请确保清除系统变量。group_replication_force_members必须为空才能执行START GROUP_REPLICATION语句。

group_replication_group_name

```
Command-Line Format --group-replication-group-name=value
System Variable group_replication_group_name
Scope      Global
Dynamic    Yes
SET_VAR Hint Applies      No
Type       String
```

可以在运行组复制时更改此系统变量的值，但是该更改仅在停止并重新启动组成员上的组复制后才生效。

group_replication_group_name指定此实例所属的集群的名称，该名称必须是有效的UUID。在二进制日志中为组复制事务设置GTID时在内部使用此UUID。

group_replication_group_seeds

```
Command-Line Format --group-replication-group-seeds=value
System Variable group_replication_group_seeds
Scope      Global
Dynamic    Yes
SET_VAR Hint Applies      No
Type       String
```

可以在运行组复制时更改此系统变量的值，但是该更改仅在停止并重新启动组成员上的组复制后才生效。

group_replication_group_seeds是组成员列表，加入节点可以连接到该组成员以获取所有当前组成员的详细信息。加入节点使用这些详细信息来选择并连接到组成员，以获得与集群同步所需的数据。该列表由每个包含的种子节点的单个内部网络地址或主机名组成，该内部网络地址或主机名是在种子节点的group_replication_local_address系统变量中配置的（不是由MySQL服务器的主机名和端口系统变量指定的种子节点的SQL客户端连接）。种子节点的地址指定为以逗号分隔的列表，例如host1: port1, host2: port2。必须在方括号中指定IPv6地址。例如：

```
group_replication_group_seeds= "198.51.100.44:33061,
[2001:db8:85a3:8d3:1319:8a2e:370:7348]:33061, example.org:33061"
```

在执行START GROUP_REPLICATION语句并且组通信系统（GCS）可用之前，不会验证为此变量指定的值。

通常，此列表由集群的所有节点组成，但是可以选择组成员的子集作为种子节点。该列表必须至少包含一个有效的节点地址。启动组复制时，将验证每个地址。如果列表不包含任何有效的成员地址，则执行START GROUP_REPLICATION失败。

节点加入复制组时，它将尝试连接到其group_replication_group_seeds系统变量中列出的第一个种子节点。如果拒绝连接，则加入节点尝试按顺序连接到列表中的每个其他种子节点。如果加入节点连接到种子节点，但没有因此而被添加到复制组中（例如，因为种子节点在其白名单中没有加入节点的地址并关闭了连接），则加入节点将继续执行以下操作：按顺序尝试列表中其余的种子节点。

加入节点必须使用种子节点在group_replication_group_seeds选项中发布的相同协议（IPv4或IPv6）与种子节点进行通信。出于组复制的IP地址权限的目的，种子节点上的白名单必须包括种子节点提供的协议的加入节点的IP地址，或解析为该协议地址的主机名。如果该地址或主机名与种子节点的播发协议不匹配，则除了该加入节点的group_replication_local_address外，还必须设置并允许该地址或主机名。如果加入节点没有适用协议的允许地址，则拒绝其连接尝试。

group_replication_gtid_assignment_block_size

```
Command-Line Format --group-replication-gtid-assignment-block-size=#
System Variable group_replication_gtid_assignment_block_size
Scope      Global
Dynamic    Yes
SET_VAR Hint Applies      No
Type       Integer
Default Value      1000000
Minimum Value      1
Maximum Value (64-bit platforms)  9223372036854775807
Maximum Value (32-bit platforms)  4294967295
```

此系统变量是集群范围的配置设置。它必须在所有组成员上具有相同的值，并且在组复制运行时不能更改，并且需要完全重新引导集群（由具有group_replication_bootstrap_group = ON的节点进行引导）才能使值更改生效。

group_replication_gtid_assignment_block_size指定为每个组成员保留的连续GTID的数量。为每个节点保留的连续GTID数。每个节点从中进行消耗，并在其需要的时候获取更多的GTID数（类似于分布式事务中的全局序列，该系统变量设置的值表示每个成员每一次从全局序列中获取多大范围的连续数字范围来作为自身写事务的GTID号）。

```
group_replication_ip_allowlist
```

```
Command-Line Format --group-replication-ip-allowlist=value
```

```
Introduced 8.0.22
```

```
System Variable group_replication_ip_allowlist
```

```
Scope Global
```

```
Dynamic Yes
```

```
SET_VAR Hint Applies No
```

```
Type String
```

```
Default Value AUTOMATIC
```

MySQL 8.0.22提供了group_replication_ip_allowlist来代替group_replication_ip_whitelist。组复制正在运行时，无法更改此系统变量的值。

group_replication_ip_allowlist指定允许哪些主机连接到集群。必须在复制组中的其他实例上在group_replication_local_address中为每个组成员指定对应的地址。在执行START GROUP_REPLICATION语句并且组通信系统（GCS）可用之前，不会验证此变量指定的值。

默认情况下，此系统变量设置为AUTOMATIC，允许来自主机上活跃的专用子网的连接。用于组复制（XCom）的组通信引擎会自动扫描主机上的活跃端口，并使用专用子网中的地址来标识这些端口。这些地址以及IPv4和（来自MySQL 8.0.14）IPv6的本地主机IP地址用于创建组复制白名单。

专用地址白名单不能用于来自专用网络外部server的连接。对于位于的实例之间的组复制连接，必须提供公用IP地址并将其指定为显式白名单。如果为白名单指定任何条目，则不会自动添加专用地址，因此，如果使用其中任何一个，则必须显式指定它们。本地主机IP地址是自动添加的。

作为group_replication_ip_allowlist选项的值，可以指定以下各项的任意组合：

IPv4地址（例如198.51.100.44）

具有CIDR表示法的IPv4地址（例如192.0.2.21/24）

来自MySQL 8.0.14的IPv6地址（例如2001:db8:85a3:8d3:1319:8a2e:370:7348）

来自MySQL 8.0.14的，带有CIDR表示法的IPv6地址（例如2001:db8:85a3:8d3::/64）

主机名（例如example.org）

带有CIDR表示法的主机名（例如，www.example.com / 24）

在MySQL 8.0.14之前，主机名只能解析为IPv4地址。从MySQL 8.0.14开始，主机名可以解析为IPv4地址和/或IPv6地址。如果主机名同时解析为IPv4和IPv6地址，则IPv4地址始终用于组复制连接。可以将CIDR表示法与主机名或IP地址结合使用，以允许使用具有特定网络前缀的IP地址块，但是请确保指定子网中的所有IP地址都在你控制范围。

逗号必须分隔白名单中的每个条目。例如：

```
"192.0.2.21/24,198.51.100.44,203.0.113.0/24,2001:db8:85a3:8d3:1319:8a2e:370:7348,example.org,www.example.com/24"
```

如果加入节点具有IPv4 `group_replication_local_address`时，但是`group_replication_group_seeds`选项中使用IPv6地址列出了该集群的种子节点，则必须为所提供的协议设置允许加入节点的地址在种子节点（或解析为该协议地址的主机名）的白名单中存在。

例如，可以根据安全要求在不同的组成员上配置不同的白名单，以使不同的子网分开。但是，这可能会在重新配置集群时引起问题。如果没有特定的安全要求，请在集群的所有节点上使用相同的白名单。

对于主机名，仅当另一台服务器发出连接请求时，才进行名称解析。无法解析的主机名不会用于白名单验证，并且警告消息会写入错误日志。对已解析的主机名执行反向DNS（FCrDNS）验证。

ps:主机名本质上不如白名单中的IP地址安全。FCrDNS验证提供了良好的保护，但可能会受到某些类型的攻击的损害。仅在绝对必要时在白名单中指定主机名，并确保所有用于名称解析的组件（例如DNS服务器）都在控制范围内。也可以使用hosts文件在本地实现名称解析，以避免使用外部组件。

```
group_replication_ip_whitelist

Command-Line Format --group-replication-ip-whitelist=value
Deprecated 8.0.22
System Variable group_replication_ip_whitelist
Scope Global
Dynamic Yes
SET_VAR Hint Applies No
Type String
Default Value AUTOMATIC
```

从MySQL 8.0.22开始，不推荐使用`group_replication_ip_whitelist`，并且可以使用`group_replication_ip_allowlist`替换它。对于这两个系统变量，默认值为AUTOMATIC。如果一个系统变量已设置为用户定义的值，而另一个则未设置，则使用更改后的值。如果两个系统变量都已设置为用户定义的值，则使用`group_replication_ip_allowlist`的值。

新系统变量的工作方式与旧系统变量的工作方式相同，只是术语有所更改。为`group_replication_ip_allowlist`给出的行为描述适用于新旧系统变量。

```
group_replication_local_address

Command-Line Format --group-replication-local-address=value
System Variable group_replication_local_address
Scope Global
Dynamic Yes
SET_VAR Hint Applies No
Type String
```

可以在运行组复制时更改此系统变量的值，但是该更改仅在停止并重新启动组成员上的组复制后才生效。

`group_replication_local_address`设置该节点为其他节点提供的连接所提供的网络地址，指定为host:port格式的字符串。集群的所有节点都必须可以访问此地址，因为组通信引擎将其用于组复制（XCom，Paxos变体），用于远程XCom实例之间的TCP通信。与本地实例的通信是通过使用共享内存的输入通道进行的。

ps:请勿使用此地址与应用通信。这不是MySQL实例SQL协议的主机和端口。

组复制将在`group_replication_local_address`中指定的地址或主机名用作复制组中组成员的唯一标识符。只要主机名或IP地址不同，就可以对复制组的所有节点使用相同的端口，并且只要端口都不同，就可以对所有节点使用相同的主机名或IP地址。推荐的`group_replication_local_address`端口为33061。请注意，在执行START GROUP_REPLICATION语句并且组通信系统（GCS）可用之前，不会验证此变量

指定的值。

由group_replication_local_address配置的网络地址必须是所有组成员均可解析的。例如，如果每个实例位于具有固定网络地址的不同主机上，则可以使用主机的IP地址，例如10.0.0.1。如果使用主机名，则必须使用标准名称，并确保可以通过DNS解析正确配置的/etc/hosts文件或其他名称解析过程来解析该主机名。从MySQL 8.0.14起，可以使用IPv6地址（或解析为它们的主机名）以及IPv4地址。必须在方括号中指定IPv6地址，以区分端口号，例如：

```
group_replication_local_address= "[2001:db8:85a3:8d3:1319:8a2e:370:7348]:33061"
```

如果指定为实例的组复制本地地址的主机名同时解析为IPv4和IPv6地址，则IPv4地址将始终用于组复制连接。

为实现组复制的IP地址白名单，每个组成员使用系统变量group_replication_local_address提供给其他节点用于组通讯的地址，需要在其他组成员的系统变量group_replication_ip_whitelist指定的白名单地址中允许其连接，否则双方就无法正常通讯。如果任何种子节点在其系统变量group_replication_group_seeds中指定了一个IPV6地址，但是加入节点在其系统变量group_replication_local_address中指定了一个IPV4地址（或相反。种子节点使用IPV4地址，加入节点使用IPV6地址），则必须在种子节点的白名单地址中添加加入节点所使用的协议地址（为减少维护成本，这种情况建议在所有节点的白名单地址中都设置一样的协议地址），否则，新的Server将会被拒绝连接。

```
group_replication_member_expel_timeout

Command-Line Format --group-replication-member-expel-timeout=#
Introduced 8.0.13
System Variable group_replication_member_expel_timeout
Scope Global
Dynamic Yes
SET_VAR Hint Applies No
Type Integer
Default Value (≥ 8.0.21) 5
Default Value (≤ 8.0.20) 0
Minimum Value 0
Maximum Value (≥ 8.0.14) 3600
Maximum Value (≤ 8.0.13) 31536000
```

可以在组复制运行时更改此系统变量的值，该更改将立即生效。每当组复制检查超时，都会读取系统变量的当前值。并非所有组成员都必须具有相同的设置，但建议这样做是为了避免意外驱逐。

group_replication_member_expel_timeout以秒为单位指定组复制组成员在产生怀疑之后，从该集群驱逐涉嫌失败的节点之前要等待的时间。在产生怀疑之前的最初5秒检测时间不计入该时间。直到并包括MySQL 8.0.20，group_replication_member_expel_timeout的值默认为0，这意味着没有等待时间，并且在5秒钟的检测时间结束后，可疑成员应立即被驱逐出集群。从MySQL 8.0.21开始，该值默认为5，这意味着在5秒钟的检测时间后5秒钟，可疑成员将被处以驱逐。

更改组成员的group_replication_member_expel_timeout的值将立即对该组成员的现有和将来的怀疑生效。因此，可以将此作为强制怀疑超时并驱逐可疑节点的方法，以允许更改集群配置。

增加group_replication_member_expel_timeout的值可以帮助避免在速度较慢或稳定性较差的网络上避免不必要的驱逐，或者在预期的瞬时网络中断或主机速度减慢的情况下避免不必要的驱逐。如果可疑节点在怀疑超时之前再次变为活跃状态，它将应用由其余组成员缓冲的所有消息并进入“online”状态，而无需手工干预。可以指定最大3600秒（1小时）的超时值。重要的是要确保XCom的消息缓存足够大，以包含指定时间段以及初始5秒检测时间段内的预期消息量，否则节点将无法重新连接。可以使用group_replication_message_cache_size系统变量来调整缓存大小限制。

如果超过了超时时间，可疑节点将在怀疑超时后立即被驱逐出集群。如果该节点能够恢复通信并收到将其驱逐的视图，并且该节点已将group_replication_autorejoin_tries系统变量设置为指定自动重新加入尝试的次数，则该节点将继续进行指定次数的重新加入该集群的尝试，在超级只读模式下。如果节点没有指定任何自动重新加入尝试，或者节点已用尽指定的尝试次数，则该节点将遵循系统变量group_replication_exit_state_action指定的操作。

```
group_replication_member_weight

Command-Line Format --group-replication-member-weight=#
System Variable group_replication_member_weight
Scope      Global
Dynamic    Yes
SET_VAR    Hint Applies      No
Type       Integer
Default Value 50
Minimum Value 0
Maximum Value 100
```

可以在组复制运行时更改此系统变量的值，该更改将立即生效。发生故障转移时，将读取系统变量的当前值。

group_replication_member_weight指定可以分配给节点的百分比权重，以在发生故障转移时（例如，当现有primary节点离开单主模式的集群时）影响节点被选举为primary节点的机会。为节点分配数字权重，以确保选择特定的节点，例如在计划的主要维护期间，或者确保在故障转移时确定特定硬件所在节点的优先级。

对于如下配置的节点组成的集群：

节点1: group_replication_member_weight = 30, server_uuid = aaaa

节点2: group_replication_member_weight = 40, server_uuid = bbbb

节点3: group_replication_member_weight = 40, server_uuid = cccc

节点4: group_replication_member_weight = 40, server_uuid = dddd

在选举新的primary节点期间，以上节点将被分为节点2，节点3，节点4和节点1。这导致在发生故障转移时，member-2被选为新的primary节点。

```
group_replication_message_cache_size

Command-Line Format --group-replication-message-cache-size=#
Introduced      8.0.16
System Variable group_replication_message_cache_size
Scope          Global
Dynamic        Yes
SET_VAR        Hint Applies      No
Type           Integer
Default Value  1073741824 (1 GB)
Minimum Value  (64-bit platforms, ≥ 8.0.21) 134217728 (128 MB)
Minimum Value  (64-bit platforms, ≤ 8.0.20) 1073741824 (1 GB)
Minimum Value  (32-bit platforms, ≥ 8.0.21) 134217728 (128 MB)
Minimum Value  (32-bit platforms, ≤ 8.0.20) 1073741824 (1 GB)
Maximum Value  (64-bit platforms) 18446744073709551615 (16 EiB)
Maximum Value  (32-bit platforms) 315360004294967295 (4 GB)
```

该系统变量在所有组成员上应具有相同的值。组复制运行时可以更改此系统变量的值。在停止并在该节点上重新启动组复制之后，该更改将对每个组成员生效。在此过程中，允许组成员之间的系统变量值不同，但是如果断开连接，节点可能无法重新连接。

`group_replication_message_cache_size`设置用于组复制（XCom）的组通信引擎中的消息缓存可用的最大内存量。XCom消息高速缓存保存作为共识协议一部分在组成员之间交换的消息（及其元数据）。在其他功能中，消息缓存用于恢复在一段时间后无法与其他组成员通信的节点重新连接到该集群的丢失消息。

`group_replication_member_expel_timeout`系统变量确定节点的初始等待时间（除了最初的5秒检测时间之外），该等待时间（最多一个小时）允许节点返回集群而不是被驱逐。XCom消息缓存的大小应参考此时间段内的预期业务量来设置，以便它包含节点成功返回所需的所有丢失消息。在MySQL 8.0.20之前，默认值为5秒检测时间，而从MySQL 8.0.21开始，默认值为5秒检测时间之后的5秒等待时间，总时间为10秒。

考虑到MySQL Server其他缓存和对象的大小，请确保系统上有足够的内存用于所选的缓存大小限制。默认设置为1073741824字节（1 GB）。最低设置也是1 GB，直到MySQL 8.0.20。从MySQL 8.0.21开始，最小设置为134217728字节（128 MB），这使得可以在具有有限可用内存量且良好的网络连接的主机上进行部署，以最大程度地减少组成员的连接暂时性丢失的频率和持续时间。请注意，使用`group_replication_message_cache_size`设置的限制仅适用于缓存中存储的数据，并且缓存结构需要额外的50 MB内存。

缓存大小限制可以在运行时动态增加或减少。如果减小缓存大小限制，则XCom会删除已确定并交付的最旧的消息条目，直到当前大小低于该限制为止。当从消息缓存中删除当前无法访问的成员可能需要恢复的消息时，组复制的组通信系统（GCS）会通过警告消息发出警报。

```
group_replication_poll_spin_loops
```

```
Command-Line Format --group-replication-poll-spin-loops=#
```

```
System Variable group_replication_poll_spin_loops
```

```
Scope      global
```

```
Dynamic Yes
```

```
SET_VAR Hint Applies      No
```

```
Type       Integer
```

```
Default Value 0
```

```
Minimum Value 0
```

```
Maximum Value (64-bit platforms) 18446744073709551615
```

```
Maximum Value (32-bit platforms) 4294967295
```

可以在运行组复制时更改此系统变量的值，但是该更改仅在停止并重新启动组成员上的组复制后才生效。

`group_replication_poll_spin_loops`指定在线程等待更多传入网络消息之前，组通信线程等待通信引擎互斥锁被释放的次数。

group_replication_recovery_complete_at

Command-Line Format `--group-replication-recovery-complete-at=value`

System Variable group_replication_recovery_complete_at

Scope **Global**

Dynamic Yes

SET_VAR Hint Applies **No**

Type Enumeration

Default Value TRANSACTIONS_APPLIED

Valid Values

TRANSACTIONS_CERTIFIED

TRANSACTIONS_APPLIED

可以在运行组复制时更改此系统变量的值，但是该更改仅在停止并重新启动组成员上的组复制后才生效。

group_replication_recovery_complete_at指定在从现有节点进行状态转移后处理缓存的事务时在分布式恢复过程中应用的策略。可以选择是在节点已加入并验证其加入该集群之前错过的所有事务之后 (TRANSACTIONS_CERTIFIED)，还是仅在节点已接收，认证并应用它们之后 (TRANSACTIONS_APPLIED)，将节点标记为在线。

group_replication_recovery_compression_algorithm

Command-Line Format `--group-replication-recovery-compression-algorithm=value`

Introduced **8.0.18**

System Variable group_replication_recovery_compression_algorithm

Scope **Global**

Dynamic Yes

SET_VAR Hint Applies **No**

Type **Set**

Default Value uncompressed

Valid Values

zlib

zstd

uncompressed

可以在运行组复制时更改此系统变量的值，但是该更改仅在停止并重新启动组成员上的组复制后才生效。

该系统变量设置组复制中基于二进制日志的状态传输时，是否允许其分布式恢复连接使用压缩算法，其有效算法与系统变量protocol_compression_algorithms的有效算法相同（但他们的默认值不相同，系统变量protocol_compression_algorithms的默认值为"zlib,zstd,uncompressed"）。

- zlib：表示允许连接使用zlib压缩算法。
- zstd：表示允许连接使用zstd压缩算法（zstd 1.3）。
- uncompressed：表示不使用压缩算法。

如果Server已配置为支持克隆插件，并且在分布式恢复期间使用远程克隆操作（使用基于远程克隆的状态传输），则此系统变量不适用这种状态传输方法，如果需要对这种状态传输方法启用压缩，则需要使用克隆插件的系统变量clone_enable_compression进行设置。

```
group_replication_recovery_get_public_key
```

```
Command-Line Format --group-replication-recovery-get-public-key[={OFF|ON}]
System Variable group_replication_recovery_get_public_key
Scope      Global
Dynamic    Yes
SET_VAR Hint Applies      No
Type       Boolean
Default Value      OFF
```

可以在运行组复制时更改此系统变量的值，但是该更改仅在停止并重新启动组成员上的组复制后才生效。

group_replication_recovery_get_public_key指定是否从主要节点请求基于RSA密钥对的密码交换所需的公钥，如果系统变量group_replication_recovery_public_key_path设置的公钥文件有效，则它将优先于系统变量group_replication_recovery_get_public_key生效（即，这种情况下系统变量group_replication_recovery_get_public_key设置的值不生效）。如果在分布式恢复中没有为组复制恢复通道group_replication_recovery启用SSL，但使用了caching_sha2_password插件来验证组复制的复制账号（这也是MySQL 8.0中的默认设置）。那么，这种情况下，系统变量group_replication_recovery_get_public_key设置的值将生效。

```
group_replication_recovery_public_key_path
```

```
Command-Line Format --group-replication-recovery-public-key-path=file_name
System Variable group_replication_recovery_public_key_path
Scope      Global
Dynamic    Yes
SET_VAR Hint Applies      No
Type       File name
Default Value      NULL
```

可以在运行组复制时更改此系统变量的值，但是该更改仅在停止并重新启动组成员上的组复制后才生效。

group_replication_recovery_public_key_path指定文件的路径名，该文件包含源进行基于RSA密钥对的密码交换secondary节点所需的副本。该文件必须为PEM格式。如果将group_replication_recovery_public_key_path设置为有效的公共密钥文件，则它优先于group_replication_recovery_get_public_key。如果未使用SSL通过group_replication_recovery通道进行分布式恢复（因此，group_replication_recovery_use_ssl设置为OFF），并且组复制的复制用户帐户使用caching_sha2_password插件（MySQL 8.0中的默认设置）或sha256_password进行身份验证，则此变量适用插入。（对于sha256_password，设置group_replication_recovery_public_key_path仅在使用OpenSSL构建MySQL时适用。）

ps:对于组复制用户使用sha256_password插件进行身份认证时，要使得系统变量group_replication_recovery_public_key_path生效，MySQL需要基于OpenSSL构建。

```
group_replication_recovery_reconnect_interval
```

```
Command-Line Format --group-replication-recovery-reconnect-interval=#  
System Variable group_replication_recovery_reconnect_interval  
Scope      Global  
Dynamic Yes  
SET_VAR Hint Applies      No  
Type      Integer  
Default Value      60  
Minimum Value      0  
Maximum Value      31536000
```

可以在运行组复制时更改此系统变量的值，但是该更改仅在停止并重新启动组成员上的组复制后才生效。

group_replication_recovery_reconnect_interval指定当在集群中找不到合适的引导节点用于分布式恢复时，会尝试重新连接（寻找）可用的引导节点，此系统变量设置重连间隔时间（单位为秒）。

```
group_replication_recovery_retry_count
```

```
Command-Line Format --group-replication-recovery-retry-count=#  
System Variable group_replication_recovery_retry_count  
Scope      Global  
Dynamic Yes  
SET_VAR Hint Applies      No  
Type      Integer  
Default Value      10  
Minimum Value      0  
Maximum Value      31536000
```

可以在运行组复制时更改此系统变量的值，但是该更改仅在停止并重新启动组成员上的组复制后才生效。

group_replication_recovery_retry_count指定在放弃之前，加入的节点尝试连接到可用引导节点以进行分布式恢复的次数。

```
group_replication_recovery_ssl_ca
```

```
Command-Line Format --group-replication-recovery-ssl-ca=value  
System Variable group_replication_recovery_ssl_ca  
Scope      Global  
Dynamic Yes  
SET_VAR Hint Applies      No  
Type      String
```

可以在运行组复制时更改此系统变量的值，但是该更改仅在停止并重新启动组成员上的组复制后才生效。

group_replication_recovery_ssl_ca指定文件的路径，该文件包含用于分布式恢复连接的受信任SSL证书颁发机构列表。

如果已将此实例设置为支持克隆，并且已将group_replication_recovery_use_ssl设置为ON，则组复制会自动将克隆SSL选项clone_ssl_ca的设置配置为与实例的设置相匹配。

group_replication_recovery_ssl_capath

```
Command-Line Format --group-replication-recovery-ssl-capath=value
System Variable group_replication_recovery_ssl_capath
Scope      Global
Dynamic Yes
SET_VAR Hint Applies      No
Type       String
```

可以在运行组复制时更改此系统变量的值，但是该更改仅在停止并重新启动组成员上的组复制后才生效。

group_replication_recovery_ssl_capath指定目录的路径，该目录包含用于分布式恢复连接的受信任SSL证书颁发机构证书。

group_replication_recovery_ssl_cert

```
Command-Line Format --group-replication-recovery-ssl-cert=value
System Variable group_replication_recovery_ssl_cert
Scope      Global
Dynamic Yes
SET_VAR Hint Applies      No
Type       String
```

可以在运行组复制时更改此系统变量的值，但是该更改仅在停止并重新启动组成员上的组复制后才生效。

group_replication_recovery_ssl_cert指定用于建立分布式恢复安全连接的SSL证书文件的名称。

如果已将此实例设置为支持克隆，并且已将group_replication_recovery_use_ssl设置为ON，则组复制会自动将克隆SSL选项clone_ssl_cert的设置配置为与实例的设置相匹配。

group_replication_recovery_ssl_cipher

```
Command-Line Format --group-replication-recovery-ssl-cipher=value
System Variable group_replication_recovery_ssl_cipher
Scope      Global
Dynamic Yes
SET_VAR Hint Applies      No
Type       String
```

可以在运行组复制时更改此系统变量的值，但是该更改仅在停止并重新启动组成员上的组复制后才生效。

group_replication_recovery_ssl_cipher指定SSL加密的允许算法列表，可用的加密算法需要看编译MySQL Server时使用的ssl lib库版本，不同的ssl lib库支持的加密算法有所不同。

group_replication_recovery_ssl_cr1

```
Command-Line Format --group-replication-recovery-ssl-cr1=value
System Variable group_replication_recovery_ssl_cr1
Scope      Global
Dynamic Yes
SET_VAR Hint Applies      No
Type       File name
```


可以在运行组复制时更改此系统变量的值，但是该更改仅在停止并重新启动组成员上的组复制后才生效。

group_replication_recovery_ssl_crl指定目录的路径，该目录包含证书吊销列表的文件。

```
group_replication_recovery_ssl_crlpath
```

```
Command-Line Format --group-replication-recovery-ssl-crlpath=value
System Variable group_replication_recovery_ssl_crlpath
Scope Global
Dynamic Yes
SET_VAR Hint Applies No
Type Directory name
```

可以在运行组复制时更改此系统变量的值，但是该更改仅在停止并重新启动组成员上的组复制后才生效。

group_replication_recovery_ssl_crlpath指定目录文件的路径，该目录包含证书吊销列表的文件。

```
group_replication_recovery_ssl_key
```

```
Command-Line Format --group-replication-recovery-ssl-key=value
System Variable group_replication_recovery_ssl_key
Scope Global
Dynamic Yes
SET_VAR Hint Applies No
Type String
```

可以在运行组复制时更改此系统变量的值，但是该更改仅在停止并重新启动组成员上的组复制后才生效。

group_replication_recovery_ssl_key指定用于建立安全连接的SSL密钥文件的名称。

如果已将此实例设置为支持克隆，并且已将group_replication_recovery_use_ssl设置为ON，则组复制会自动将克隆插件的系统变量clone_ssl_key的值设置为匹配系统变量group_replication_recovery_ssl_key的值。

```
group_replication_recovery_ssl_verify_server_cert
```

```
Command-Line Format --group-replication-recovery-ssl-verify-server-cert[=
{OFF|ON}]
System Variable group_replication_recovery_ssl_verify_server_cert
Scope Global
Dynamic Yes
SET_VAR Hint Applies No
Type Boolean
Default value OFF
```

可以在运行组复制时更改此系统变量的值，但是该更改仅在停止并重新启动组成员上的组复制后才生效。

group_replication_recovery_ssl_verify_server_cert指定在分布式恢复过程中用于检查引导节点的证书中Server的Common Name（共同名称）值。

group_replication_recovery_tls_ciphersuites

Command-Line Format `--group-replication-recovery-tls-ciphersuites=value`

Introduced **8.0.19**

System Variable `group_replication_recovery_tls_ciphersuites`

Scope **Global**

Dynamic **Yes**

SET_VAR Hint Applies **No**

Type **String**

Default Value **NULL**

可以在运行组复制时更改此系统变量的值，但是该更改仅在停止并重新启动组成员上的组复制后才生效。

当TLSv1.3用于分布式恢复连接的连接加密，并且此实例是分布式恢复连接中的客户端（即加入节点）时，`group_replication_recovery_tls_ciphersuites`指定一个或多个允许密码组的冒号分隔列表。如果在使用TLSv1.3时将此系统变量设置为NULL（如果未设置系统变量，则为默认值），则允许默认启用的密码套件。如果将此系统变量设置为空字符串，则不允许使用密码套件，因此不使用TLSv1.3。从MySQL 8.0.19开始，此系统变量可用。

group_replication_recovery_tls_version

Command-Line Format `--group-replication-recovery-tls-version=value`

Introduced **8.0.19**

System Variable `group_replication_recovery_tls_version`

Scope **Global**

Dynamic **Yes**

SET_VAR Hint Applies **No**

Type **String**

Default Value **TLSv1,TLSv1.1,TLSv1.2,TLSv1.3**

可以在运行组复制时更改此系统变量的值，但是该更改仅在停止并重新启动组成员上的组复制后才生效。

当此实例是分布式恢复连接中的客户端（即加入节点）时，`group_replication_recovery_tls_version`指定一个或多个允许用于连接加密的TLS协议的逗号分隔列表。确保指定的版本是连续的（例如，“TLSv1, TLSv1.1, TLSv1.2”）。如果未设置此系统变量，则使用默认的“TLSv1, TLSv1.1, TLSv1.2, TLSv1.3”。每个分布式恢复连接中涉及的组成员（作为客户端（加入节点）和服务端（引导节点））协商它们都设置为支持的最高协议版本。该系统变量可从MySQL 8.0.19获得。

group_replication_recovery_use_ssl

Command-Line Format `--group-replication-recovery-use-ssl[={OFF|ON}]`

System Variable `group_replication_recovery_use_ssl`

Scope **Global**

Dynamic **Yes**

SET_VAR Hint Applies **No**

Type **Boolean**

Default Value **OFF**

可以在运行组复制时更改此系统变量的值，但是该更改仅在停止并重新启动组成员上的组复制后才生效。

`group_replication_recovery_use_ssl`指定组成员之间的“组复制”分布式恢复连接是否应使用SSL。

如果Server被设置为支持克隆插件，其将此系统变量设置为ON，则组复制将在基于远程克隆操作的状态传输以及基于二进制日志的状态传输中使用SSL；如果将此系统变量设置为OFF，则组复制不会在基于远程克隆操作的状态传输中使用SSL。

```
group_replication_recovery_zstd_compression_level
```

```
Command-Line Format --group-replication-recovery-zstd-compression-level=#
```

```
Introduced 8.0.18
```

```
System Variable group_replication_recovery_zstd_compression_level
```

```
Scope Global
```

```
Dynamic Yes
```

```
SET_VAR Hint Applies No
```

```
Type Integer
```

```
Default Value 3
```

```
Minimum Value 1
```

```
Maximum Value 22
```

可以在运行组复制时更改此系统变量的值，但是该更改仅在停止并重新启动组成员上的组复制后才生效。

group_replication_recovery_zstd_compression_level指定组复制在分布式恢复连接中使用zstd压缩算法时的压缩级别。允许的级别从1到22，值越大表示压缩级别越高。默认的zstd压缩级别为3。对于不使用zstd压缩的分布式恢复连接，此变量不生效。

```
group_replication_single_primary_mode
```

```
Command-Line Format --group-replication-single-primary-mode[={OFF|ON}]
```

```
System Variable group_replication_single_primary_mode
```

```
Scope Global
```

```
Dynamic Yes
```

```
SET_VAR Hint Applies No
```

```
Type Boolean
```

```
Default Value ON
```

此系统变量是集群范围的配置设置。它必须在所有组成员上具有相同的值，并且在组复制运行时不能更改，并且需要完全重新引导集群（由具有group_replication_bootstrap_group = ON的实例进行引导）才能使值更改生效。从MySQL 8.0.16开始，可以在集群仍在运行时使用group_replication_switch_to_single_primary_mode () 和group_replication_switch_to_multi_primary_mode () UDF更改此系统变量的值。

group_replication_single_primary_mode指示该集群自动选择一个实例作为处理读/写工作负载的server。该server为读写节点，所有其他server均为SECONDARIES。

```
group_replication_ssl_mode
```

```
Command-Line Format --group-replication-ssl-mode=value
```

```
System Variable group_replication_ssl_mode
```

```
Scope Global
```

```
Dynamic Yes
```

```
SET_VAR Hint Applies No
```

```
Type Enumeration
```

```
Default Value DISABLED
```

```
Valid Values
```

```
DISABLED
```

REQUIRED

VERIFY_CA

VERIFY_IDENTITY

可以在运行组复制时更改此系统变量的值，但是该更改仅在停止并重新启动组成员上的组复制后才生效。

group_replication_ssl_mode设置组复制成员之间的组通信连接的安全状态。可能的取值如下：

DISABLED：建立未加密的连接（默认）。

REQUIRED：如果server支持安全连接，则建立安全连接。

VERIFY_CA：与REQUIRED相似，但另外还要针对配置的证书颁发机构（CA）证书验证服务器TLS证书。

VERIFY_IDENTITY：与VERIFY_CA类似，但还要验证该server证书是否与尝试连接的主机匹配。

group_replication_start_on_boot

Command-Line Format --group-replication-start-on-boot[={OFF|ON}]

System Variable group_replication_start_on_boot

Scope Global

Dynamic Yes

SET_VAR Hint Applies No

Type Boolean

Default Value ON

可以在运行组复制时更改此系统变量的值，但是该更改仅在停止并重新启动组成员上的组复制后才生效。

group_replication_start_on_boot指定实例在启动过程中是否应自动启动组复制（ON）或不启动（OFF）。如果将此选项设置为ON，则在将远程克隆操作用于分布式恢复后，组复制会自动重新启动。

若要在实例启动期间自动启动组复制，必须使用CHANGE MASTER TO语句将用于分布式恢复的用户凭据存储在实例上的复制元数据存储库中。如果希望在仅将用户凭据存储在内存中的START GROUP_REPLICATION语句上指定用户凭据，请确保将group_replication_start_on_boot设置为OFF。

group_replication_tls_source

Command-Line Format --group-replication-tls-source=value

Introduced 8.0.21

System Variable group_replication_tls_source

Scope Global

Dynamic Yes

SET_VAR Hint Applies No

Type Enumeration

Default Value mysql_main

Valid Values

mysql_main

mysql_admin

可以在运行组复制时更改此系统变量的值，但是该更改仅在停止并重新启动组成员上的组复制后才生效。

group_replication_tls_source指定用于组复制的TLS资料的来源。

```
group_replication_transaction_size_limit
```

```
Command-Line Format --group-replication-transaction-size-limit=#
System Variable group_replication_transaction_size_limit
Scope      Global
Dynamic    Yes
SET_VAR    Hint Applies      No
Type       Integer
Default Value 150000000
Minimum Value 0
Maximum Value 2147483647
```

该系统变量在所有组成员上应具有相同的值。组复制运行时可以更改此系统变量的值。更改将立即对组成员生效，并从该成员上开始的下一个事务开始应用。在此过程中，允许组成员之间的系统变量值不同，但是某些事务可能会提交失败。

group_replication_transaction_size_limit配置复制组接受的最大事务大小（以字节为单位）。大于此大小的事务将被接收节点回滚，并且不会广播到该集群。大事务可能会在内存分配方面导致复制组出现问题，这可能会导致系统变慢，或者在网络带宽消耗方面可能会导致怀疑成员由于正在忙于处理大型事务而发生故障。

当此系统变量设置为0时，该集群接受的事务大小没有限制。从MySQL 8.0开始，此系统变量的默认设置为150000000字节（约143 MB）。处理事务所花费的时间与其大小成正比，请根据需要该集群允许的最大消息大小来调整此系统变量的值。在所有组成员上，group_replication_transaction_size_limit的值都应该相同。

```
group_replication_unreachable_majority_timeout
```

```
Command-Line Format --group-replication-unreachable-majority-timeout=#
System Variable group_replication_unreachable_majority_timeout
Scope      Global
Dynamic    Yes
SET_VAR    Hint Applies      No
Type       Integer
Default Value 0
Minimum Value 0
Maximum Value 31536000
```

可以在组复制运行时更改此系统变量的值，该更改将立即生效。发生问题（表示需要该行为）时，将读取系统变量的当前值。

group_replication_unreachable_majority_timeout指定遭受网络分区并且无法连接到多数节点的成员在离开集群之前等待的秒数。假设在一复制组中有5个节点(S1、S2、S3、S4、S5)，如果(S1、S2)和(S3、S4、S5)之间断开连接，则表示存在了网络分区。第一个部分(S1,S2)是少数（因为节点数少于集群中的半数）。第二个部分(S3、S4、S5)是多数，因此，第二部分可以正常执行仲裁，形成新的组成员资格并正常对外提供服务，而少数部分将等待系统变量

group_replication_unreachable_majority_timeout指定的时间进行网络重新连接。

默认情况下，group_replication_unreachable_majority_timeout设置为0，这意味着由于网络分区而处于少数群体的成员将永远等待离开该组。如果设置了超时，则在经过指定的时间后，少数状态的节点处理的所有阻塞事务将被回滚，且处于少数状态的节点将转变为ERROR状态。。如果节点的group_replication_autorejoin_tries系统变量设置为指定自动重新加入尝试的次数，则在超级只读模式下，该节点将进行指定次数的重新加入集群的尝试。如果节点没有指定任何自动重新加入尝试，或者节点已用尽指定的尝试次数，则该节点将遵循系统变量group_replication_exit_state_action指定的操作。

ps:如果有一个对称集群，例如只有两个节点（S0，S2），如果发生网络分区，则两个部分都被判定为少数节点部分，整个集群都无法对外提供服务，这个时候，如果系统变量group_replication_unreachable_majority_timeout设置为非0值，那么所有节点在等待超时时间之后都会转变为ERROR状态。

```
group_replication_primary_member
```

当集群以单主模式运行时，显示primary节点的UUID。如果集群以多主模式运行，则显示一个空字符串。

ps:group_replication_primary_member状态变量已被弃用，并计划在以后的版本中删除。