M->S(relay master)->S

Node1 (master)	Node2 (relay master)(replica)	Node3 (replica)
>vi my.cnf	>vi my.cnf	>vi my.cnf
Note: you should enable binary.log	**Note:** you should enable binary.log, relay.log, and log slave update	[mysqld]
[mysqld]	**log slave update:** It helps to write relay.log data to the binary log then node 3 IO thread easy pull data from node 2 binary log	basedir=/home/mysql_8_2 datadir=/home/node3/mysql_3 port=3310 server_id=5 socket=/tmp/node3.sock
basedir=/home/mysql_8_2 datadir=/home/node1/mysql_1 port=3306 server_id=7 socket=/tmp/node1.sock		#Error log log_error =/home/node3/error_3308.log

#Error log
log_error
=/home/node1/error_3308.log
	[mysqld]	#general log
		general_log=on
		general_log_file=/home/node3/error_
		general.log

#general log
general_log=on
general_log_file=/home/node1/error_
general.log
	basedir=/home/mysql_8_2
	datadir=/home/node2/mysql_2
	port=3309
	server_id=4
	socket=/tmp/node2.sock
		#binary log

#binary log
log-bin=/home/node1/binary/error_bi
nary.log
	#Error log
	log_error =/home/node2/error_3308.log
		#log-bin=/home/node3/binary/error_bi
		nary.log

#slow query log
slow_query_log = 1
slow-query_log_file=/home/node1/err
or_slow.log
long_query_time = 2
	#general log
	general_log=on
	general_log_file=/home/node2/error_ge
	neral.log
		#slow query log
		slow_query_log = 1
		slow-query_log_file=/home/node3/err
		or_slow.log
		long_query_time = 2

	#binary log log-bin=/home/node2/binary/error_binar y.log	#relay log relay_log = /home/node3/relay/relay.log

	#slow query log
	slow_query_log = 1
	slow-query_log_file=/home/node2/error
	_slow.log
	long_query_time = 2	

	#relay log
	relay_log = /home/node2/relay/relay.log	

	log_slave_updates = 1	

Create user in master for slave DB	Create user in master for slave DB	
>create user "antrow"@"localhost" identified by "mano"; > grant replication slave on *.* to antrow@'localhost'; (or) >grant replication slave on *.* to 'antrow'@'localhost' identified by "antrow";	>create user "mano"@"localhost" identified by "mano"; > grant replication slave on *.* to 'mano'@'localhost'; (or) >grant replication slave on *.* to 'mano'@'localhost' identified by "mano";	
Note: >If the server's new we no need to take backup. >If data is there in master we need to take backup and restore it to the slave **mysqldump** >mysqldump -u root -p -S /tmp/node1.sock --all-databases >node1.sql	**Note:** >If the server's new we no need to take backup. >If data is there in master we need to take backup and restore it to the slave **Restore** >mysql -u root -p –S /tmp/node2.sock <node1.sql	**Note:** >If the server's new we no need to take backup. >If data is there in master we need to take backup and restore it to the slave **Restore** >mysql -u root -p –S /tmp/node3.sock <node1.sql
Check binlog position >Show master status; master_log_file='error_binary.000003 master_log_pos=888		
	Enable replication on slave db for connect to the master >change master to master_host='127.0.0.1',master_user=' antrow',master_password='antrow',master_port=3306,master_log_file='error_binary.000003',master_log_pos=888;	

	>Start slave\G Slave_IO_Running: Yes Slave_SQL_Running: Yes >Stop slave\G Slave_IO_Running: No Slave_SQL_Running: No	
	Check binlog position >Show master status; master_log_file='binlog.000002' master_log_pos=1710	
		Enable replication on slave db for connect to the master >change master to master_host='127.0.0.1',master_user='mano',master_password='mano',master_port=3306,master_log_file='binlog.000002',master_log_pos=1710; Note: Master_host —> master ip Master_user —>we create user before in master that username Master_password—--> that user password Master_port —----> master server post Master_log_file —---->binary log file Master_log_pos —-----> binary log position

		>Start slave\G Slave_IO_Running: Yes Slave_SQL_Running: Yes >Stop slave\G Slave_IO_Running: No Slave_SQL_Running: No