MYSQL Replication: Changing a Slave Database Server to be a Master

posted by   *crmanski* on   *Mon, 06/15/2009 - 3:38pm*

Senario:
Our webservers use a database backend which is also replicated to two other servers. In the even that there is a failure then one of the others can take over. oldserver1.berlinschools.org was not doing well and was also the master in a mysql replication setup with two "slave" servers. One older (olderserver2.berlinschools.org) and one basically new (newserver.berlinschools.org). The need was to make it so oldserver1's master status was passed onto newserver and oldserver2 would look to ne for slave updates.

Preparation:
To make this easier on myself first in preparation I made sure that every php/mysql web app that I have running is using a dns name for its mysql server setting (drupal, moodle, gallery, xoops, etc). I choose master.berlinschools.org. To test this name I made sure that the command: dig master.berlinschools.org would respond correctly. I also checked the hosts file for entries. One did have this setting, because it was using an external service provider's DNS instead of our internal.

Making it happen:
I opened 3 separate terminals to each mysql server and logged in as root. Then I logged onto mysql as root with this command...
mysql -u root -pMyMYSQL-PasswordHERE (Note: no space between -p and the actual password)

DNS:
I moved back to my dns configuration and punched in the new ip address for the mysqlmaster dns entry and forced DNS to update. I updated the HOST file on the one that needed it.

MYSQL:
On the master first ran...
FLUSH LOGS;

On the new master I ran...
STOP SLAVE;
RESET MASTER;
CHANGE MASTER TO MASTER_HOST='newserversIPaddress';

On the slave I ran...
STOP SLAVE;
RESET MASTER;
CHANGE MASTER TO MASTER_HOST='newserversIPaddress; (current IP of newserver)
START SLAVE;

Back to the old master I ran...
STOP SLAVE;
RESET MASTER;
CHANGE MASTER TO MASTER_HOST='newserversIPaddress';
START SLAVE;

See the MYSQL official replication FAQ for more information:

http://dev.mysql.com/doc/refman/5.0/en/replication-faq.html

TECHNOLOGY:

MYSQL

Replication

Disaster Prevention