

Examen Práctico UD1, UD2, UD3 y UD4

Administración de Sistemas Gestores de Bases de
Datos

10/12/2020

2º ASIR

Rafael Jiménez Cobos

Contenido

Ejercicio 1 3

Ejercicio 2 4

Ejercicio 3 5

Ejercicio 4 5

Ejercicio 5 5

Ejercicio 6 7

Ejercicio 7 7

Ejercicio 8 8

Ejercicio 9 8

Ejercicio 1. Por defecto la información que queda registrada no incluye todos los registros. Modifica la configuración de MariaDB para que quede registrada toda la actividad aunque el rendimiento de la misma disminuya. Crea la base de datos examen_01 y comprueba cómo queda registrado el evento.

Editamos el fichero /etc/mysql/mariadb.conf.d/50-server.cnf y habilitamos lo siguiente:

Fichero log general. Des-comentamos las siguientes líneas:

```
#
# * Logging and Replication
#
# Both location gets rotated by the cronjob.
# Be aware that this log type is a performance killer.
# Recommend only changing this at runtime for short testing periods if needed!
general_log_file      = /var/log/mysql/mysql.log
general_log           = 1
```

Fichero consultas lentas. Des-comentamos las siguientes líneas:

```
# When running under systemd, error logging goes via stdout/stderr to journald
# and when running legacy init error logging goes to syslog due to
# /etc/mysql/conf.d/mariadb.conf.d/50-mysqld_safe.cnf
# Enable this if you want to have error logging into a separate file
#log_error = /var/log/mysql/error.log
# Enable the slow query log to see queries with especially long duration
slow_query_log_file    = /var/log/mysql/mariadb-slow.log
long_query_time        = 10
log_slow_verbosity     = query_plan,explain
log-queries-not-using-indexes
```

Fichero log binario. Des-comentamos las siguientes líneas:

```
# The following can be used as easy to replay backup logs or for replication.
# note: if you are setting up a replication slave, see README.Debian about
#       other settings you may need to change.
server-id              = 1
log_bin                = /var/log/mysql/mysql-bin.log
expire_logs_days       = 10
```

A continuación, creamos la base de datos:

```
MariaDB [(none)]> CREATE DATABASE examen_01;
Query OK, 1 row affected (0.000 sec)

MariaDB [(none)]>
```

Por último, vamos a comprobar el fichero log general (/var/log/mysql/mysql.log) para ver si se ha registrado este evento:

```
201210 1:07:54      31 Connect  root@localhost on  using Socket
          31 Query    select @@version_comment limit 1
201210 1:08:09      31 Query    CREATE DATABASE examen_01
201210 1:08:25      31 Quit
```

Ejercicio 2. Descarga el fichero del examen de moodle para recuperar la copia de seguridad en la base de datos creada anteriormente.

Recuperamos la base de datos:

```
rafajimenez@examen:~$ sudo mysql examen_01 < casino.sql
rafajimenez@examen:~$ _
```

Comprobación:

[illegible]

Ejercicio 3. Configura MariaDB para que permita conexiones desde máquinas distintas a localhost.

Editamos el fichero `/etc/mysql/mariadb.conf.d/50-server.cnf` y comentamos la siguiente línea:

```
# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
#bind-address            = 127.0.0.1
```

Ejercicio 4. Crea un usuario llamado `examen_01` con acceso completo a la base de datos anterior desde la dirección IP del cliente.

```
MariaDB [(none)]> CREATE USER 'examen_01'@'192.168.112.10';
Query OK, 0 rows affected (0.003 sec)

MariaDB [(none)]> GRANT ALL PRIVILEGES ON examen_01.* to 'examen_01'@'192.168.112.10' IDENTIFIED BY
'usuario' WITH GRANT OPTION;
Query OK, 0 rows affected (0.003 sec)

MariaDB [(none)]>
```

Ejercicio 5. Configura la conexión del cliente gráfico para acceder a la base de datos del servidor desde el cliente.

Modificamos el fichero `/etc/phpmyadmin/config.inc.php`, cambiando el valor de la variable `$dbserver` por la IP de nuestro servidor:



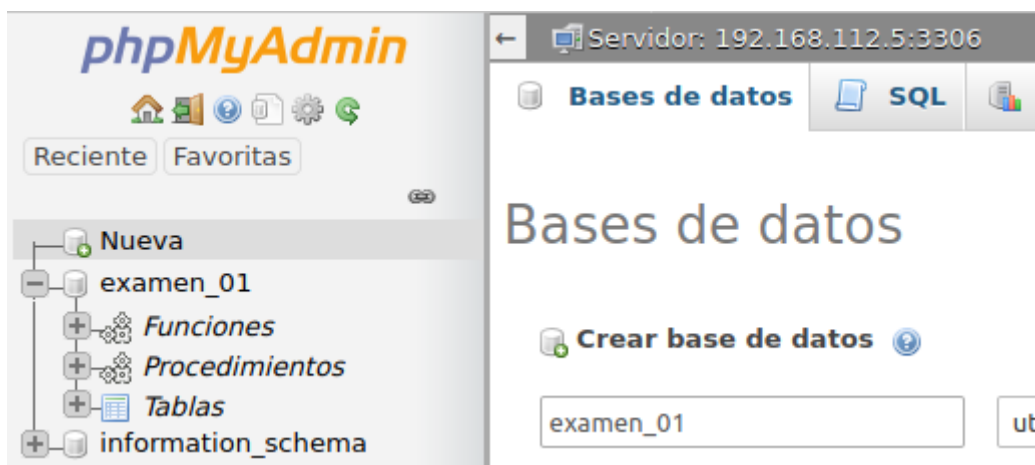
```
rafajimenez@cliente-examen: ~
GNU nano 4.8 /etc/phpmyadmin/config.inc.php
/* Configure according to dbconfig-common if enabled */
if (!empty($dbname)) {
    /* Authentication type */
    $cfg['Servers'][$i]['auth_type'] = 'cookie';
    /* Server parameters */
    if (empty($dbserver)) $dbserver = '192.168.112.5';
    $cfg['Servers'][$i]['host'] = $dbserver;

    if (!empty($dbport) || $dbserver != '192.168.112.5') {
        $cfg['Servers'][$i]['connect_type'] = 'tcp';
        $cfg['Servers'][$i]['port'] = $dbport;
    }
}
```

A continuación, modificamos el fichero `/etc/phpmyadmin/config-db.php`, introduciendo la siguiente información:

```
GNU nano 4.8 /etc/phpmyadmin/config-db.php
<?php
##
## database access settings in php format
## automatically generated from /etc/dbconfig-common/php
## by /usr/sbin/dbconfig-generate-include
##
## by default this file is managed via ucf, so you should
## worry about manual changes being silently discarded.
## you'll probably also want to edit the configuration f
## above too.
##
$dbuser='examen_01';
$dbpass='usuario';
$dbasepath='';
$dbname='examen_01';
$dbserver='192.168.112.5';
$dbport='3306';
$dbtype='mysql';
```

Comprobación:



Ejercicio 6. Modifica la configuración del script en bash para que pueda ejecutarse atacando a la base de datos del examen.

```

rafajimenez@examen: ~
GNU nano 3.2          casino.sh

#!/bin/bash

function anadir_tirada {
    echo "Pulse Intro para mover la ruleta"
    read TECLA
    echo "CALL nueva_tirada()" | mysql examen_01
    echo " "
    read TECLA
    echo "Pulse Intro para continuar"
}

while [ true ]
do
    echo "Resumen de las tiradas"
    echo "CALL resultados()" | mysql -t examen_01
    echo "Fin de sesión"
done
  
```

Ejercicio 7. Crea una función en MariaDB que calcule el porcentaje de victorias a partir del número de victorias y derrotas llamada porcentaje().

Detalles				
Nombre de rutina	porcentaje			
Tipo	FUNCTION			
Parámetros	Nombre	Tipo	Longitud/Valores	Opciones
	↑ victorias	INT	11	
	↑ derrotas	INT	11	
Agregar parámetro				
Retornar el tipo	FLOAT			
Retornar longitud/valores				
Retornar opciones				
Definición	<pre> 1 BEGIN 2 if (victorias+derrotas)>0 THEN 3 return 100*victorias/(victorias+derrotas); 4 end if; 5 END </pre>			

Ejercicio 8. Modifica el disparador tiradas_AI_Trigger para que haga uso de esta función en lugar de calcularlo.

Detalles

Nombre del disparador	tiradas_AI_Trigger
Tabla	tiradas
Tiempo	AFTER
Evento	INSERT
Definición	<pre>1 BEGIN 2 DECLARE ganador INT(1); 3 4 SET ganador=calcula_ganador(NEW.resultado); 5 6 UPDATE resumen SET victorias=victorias+1 where resultado=ganador; 7 UPDATE resumen SET derrotas=derrotas+1 where resultado!=ganador; 8 UPDATE resumen SET porcentaje=porcentaje(victorias,derrotas) where (victorias+derrotas)>0; 9 10 END</pre>
Definidor	root@localhost

Ejercicio 9. Haz una copia de seguridad de la base de datos con mysqldump de forma que explícitamente incluya triggers, funciones y procedimientos.

```
rafajimenez@examen:~$ sudo mysqldump --routines examen_01 > examen_01.sql
[sudo] password for rafajimenez:
```