CRIAR USUARIO E OFERECER GRANT

create user 'zabbix\_read'@'%' identified by 'zabbix\_read';

grant select on *.* to 'zabbix\_read'@'%';

Alterar USER com limite de tempo (max\_statement\_time)

<https://mariadb.com/kb/en/alter-user/>

ALTER USER 'backupuser'@'%' WITH MAX\_STATEMENT\_TIME 43200;

show variables like 'max\_statement\_time';

ALTER USER 'zabbix\_proxy

matar querys por usuario:

SELECT GROUP\_CONCAT(CONCAT('KILL ',id,';') SEPARATOR ' ') FROM information\_schema.processlist WHERE user == 'otrs2';

matar querys por tempo:

SELECT GROUP\_CONCAT(CONCAT('KILL ',id,';') SEPARATOR ' ') FROM information\_schema.processlist where info is not null and time > 60;

SELECT count FROM hosts where

Coletar itens não suportados:

select count(irt.itemid) as count, i.hostid, [h.name](http://h.name), irt.error from item\_rtdata irt inner join items i on irt.itemid = i.itemid inner join hosts h on h.hostid = i.hostid where i.status =0 and irt.state = 1 group by i.hostid, irt.error order by count;

==========

Coletar itens com tempo baixo de atualização:

select count(i.itemid) as count, h.host, i.delay from items i inner join  hosts h on h.hostid = i.hostid  where i.status = 0 and h.status = 0 and i.delay = "30s" group by h.host, i.delay order by count;

===========

QUERY QUE LISTA PROFILES COM DASH UTILIZANDO INTERVALO DE ATUALIZAÇÃO

SELECT \* FROM profiles WHERE value\_int <60;

Query para DELETAR o refresh-time dos usuários com tempo de atualização de 10s:

DELETE from profiles WHERE value\_int <60;

pesquisar e mudar refresh time e numero de linhas

select alias,refresh from users where refresh <=1m;

select alias,refresh from users where refresh >=60 and refresh<=300;

update users set refresh=120 where refresh <60;

select alias,rows\_per\_page from users where rows\_per\_page > 100;

update users set rows\_per\_page=150 where rows\_per\_page > 100;

==========

Atualizar profiles por telas exemplo - é possível ajustar o tipo de tela na coluna idx

update profiles set value\_str = "now-1h" where idx = "web.auditlog.filter.from";

update profiles set value\_str = "now-1h" where idx = "web.auditlog.filter.from";

==========

LISTAR SESSÕES POR USER

SELECT count(\*), s.userid, u.alias FROM sessions s INNER JOIN users u ON s.userid = u.userid GROUP BY userid;

delete from sessions where userid = 163 limit 10000;

==========

MOSTRA TAMANHO DAS DATABASES

SELECT table\_schema AS "Database", SUM(data\_length + index\_length) / 1024 / 1024 AS "Size (MB)" FROM information\_schema.TABLES GROUP BY table\_schema;

==========

limpa os buracos de espaço em disco alocados pela tabela

MOSTRA OPTIMIZE POR DATABASE

SELECT table\_name AS 'Table', round(((data\_length + index\_length) / 1024 / 1024 / 1024), 2) 'Size (GB)', round(((data\_free) / 1024 / 1024 / 1024), 2) 'Data Free Size (GB)' FROM information\_schema.TABLES WHERE table\_schema = "zabbix" order by 'Size (GB)' desc;

RODAR OPTIMIZE:

OPTIMIZE TABLE events, event\_recovery;

==========

MOSTRA ITENS COM MENOS DE 1 MIN

select count(i.itemid) as count, h.host, i.delay from items i inner join hosts h on h.hostid = i.hostid where (i.flags = 0 or i.flags = 4) and proxy\_hostid = 11968 and i.status = 0 and h.status = 0 and (i.delay = '1m' or i.delay = '30s' or i.delay = '15s' or i.delay = '60s') group by delay, h.host order by count;

==================

SELECT DE ITENS DE 1 MINUTO POR PROXY

select count(i.itemid) as count, h.host, i.delay from items i inner join hosts h on h.hostid = i.hostid where (i.flags = 0 or i.flags = 4) and proxy\_hostid = 61187 and i.status = 0 and h.status = 0 and (i.delay = '1m' or i.delay = '30s' or i.delay = '15s' or i.delay = '60s') group by delay, h.host order by count;

===================

Outras querys

SELECT count(\*), s.userid, u.alias FROM sessions s INNER JOIN users u ON s.userid = u.userid GROUP BY userid;

#Select de usuários e sessões na tabela sessions.

SELECT s.status, s.userid, u.alias FROM sessions s INNER JOIN users u ON s.userid = u.userid GROUP BY userid;

select itemid, hostid, name, delay, master\_itemid from items where flags = 1 and delay = "1m";

select itemid, hostid, name, delay from items where flags = 1 and delay = "1m";

select count(*), delay from items where flags = 1 group by delay order by count(*) desc;

DELETE FROM sessions WHERE userid =42;

select count(irt.itemid) as count, i.hostid, [h.name](http://h.name), irt.error from item\_rtdata irt inner join items i on irt.itemid = i.itemid inner join hosts h on h.hostid = i.hostid where i.status =0 and irt.state = 1 group by i.hostid, irt.error order by count;

select count(irt.itemid) as count, h.host from item\_rtdata irt inner join items i on irt.itemid = i.itemid inner join hosts h on h.hostid = i.hostid inner join hosts\_groups hg on i.hostid = hg.hostid inner join hstgrp g on hg.groupid = g.groupid where i.status =0 and irt.state = 1 and g.groupid = 93 group by i.hostid order by count;

select count(irt.itemid) as count from item\_rtdata irt inner join items i on irt.itemid = i.itemid inner join hosts h on h.hostid = i.hostid where i.status =0 and irt.state = 1;

select count(irt.itemid) as count from item\_rtdata irt inner join items i on irt.itemid = i.itemid inner join hosts h on h.hostid = i.hostid where i.status =0 and irt.state = 1 and i.type = 11;

select count(irt.itemid) as count, h.host from item\_rtdata irt inner join items i on irt.itemid = i.itemid inner join hosts h on h.hostid = i.hostid inner join hosts\_groups hg on i.hostid = hg.hostid inner join hstgrp g on hg.groupid = g.groupid where i.status =0 and irt.state = 1 and i.type = 11 group by i.hostid order by count;

select count(i.itemid) as count, h.host, i.delay from items i inner join hosts h on h.hostid = i.hostid where i.status = 0 and h.status = 0 and i.delay = "1s" group by h.host, i.delay order by count;

How do you Disable Default 30s Auto Refresh for zabbix Graph viewing

Comando para ver slow queries rodando nas tabelas zabbix

SHOW PROCESSLIST;

Slow queries com Tempo maior que 1 minuto

select \* from INFORMATION\_SCHEMA.PROCESSLIST where Time > 3600;

Select nos eventos da partição do zabbix procurar por last executed

SELECT \* FROM INFORMATION\_SCHEMA.events\G;

SQL queries for Zabbix Posted byVyacheslav19.12.2017 Leave a commenton SQL queries for Zabbix

I will write some useful examples of sql queries for the Zabbix database:

Search for a host by name:

1 2 SELECT \* FROM hosts WHERE host like '%name%'; SELECT \* FROM hosts WHERE name like '%name%'; Find the data items of the specified host:

1 SELECT \* FROM items WHERE hostid = '10105'; Find the history of the values for the specified data item:

1 SELECT \* FROM history WHERE itemid = '24526'; Delete the whole history of the data element until 01.11.2014 (the time is specified in Unix format, converters can be found through the search engine):

1 DELETE FROM history WHERE itemid = '24526' AND clock < '1414800000'; Delete all data history until 01.11.2014:

1 DELETE FROM history WHERE clock < '1414800000'; Massively changed the interval and dynamics of changes to the data elements of templates and hosts, I have the following queries (the first identifies the template ID or host, the second changes the intervals):

1 2 3 SELECT \* FROM hosts WHERE host="Template ICMP Ping"; UPDATE items SET delay=3600 WHERE hostid=10105 AND delay=600; UPDATE items SET trends=180 WHERE hostid=10047 AND trends=365;