TiDB Allowlist Plugin User Guide

August 4, 2022

Introduction

By using the TiDB Allowlist plugin, you can manage and control an allowlist directly without developing and managing the allowlist function in your business layer. You don't need to add users one by one or manage user permissions using the MySQL user feature. This greatly improves the management efficiency.

This document describes how to compile, package, and use the allowlist plugin.

Download the plugin

You can download the plugin on TiDB Enterprise Edition Downloads.

Deploy the plugin

After downloading the plugin, you can use either TiDB Operator or TiUP to deploy the allowlist plugin.

Use TiDB Operator to deploy the plugin

Configure TidbCluster CR.

```
tidb:
  additionalContainers:
  - command:
    - sh
    - -c
    - touch /var/log/tidb/tidb-whitelist.log; tail -n0 -F /var/log/tidb/tidb-whitelist.log;
    image: busybox:1.26.2
```

```
  imagePullPolicy: IfNotPresent
  name: whitelistlog
  resources:
    limits:
      cpu: 100m
      memory: 50Mi
    requests:
      cpu: 20m
      memory: 5Mi
  terminationMessagePath: /dev/termination-log
  terminationMessagePolicy: File
  volumeMounts:
  - mountPath: /var/log/tidb
    name: slowlog
baseImage: <your_private_docker_hub_username>/tidb-ent:v5.4.0
plugins:
- whitelist-1
```

Use TiUP to deploy the plugin

Add the plugin to an existing cluster

1. Create a file on all target servers.

```
tiup cluster exec cluster_name --command "mkdir {{.DeployDir}}/plugin" -R tidb
```

2. Upload the plugin to all target servers.

```
tiup cluster push cluster_name ~/Download/whitelist-1.so {{.DeployDir}}/plugin/whitelist-1.so
-R tidb
```

3. Change the TiDB configurations.

```
tiup cluster edit-config cluster_name
```

In the config field of each TiDB node, configure `plugin.dir` as the absolute path. `plugin.load` specifies the name of the plugin.

```
tidb_servers:
-  host: 172.16.7.213
   ssh_port: 22
   port: 4000
   status_port: 10080
   deploy_dir: /tidb-deploy/tidb-4000
   log_dir: /tidb-deploy/tidb-4000/log
   arch: amd64
   os: linux
   config:
      plugin.dir: /tidb-deploy/tidb-4000/plugin
      plugin.load: whitelist-1
```

4. Restart all TiDB nodes.

```
tiup cluster reload cluster_name -R tidb
```

Tips:

- In step 3, if all nodes have the same deployment path, you can directly configure it globally in server_configs.tidb to avoid manually modifying each node. If the paths are different, you can modify the path of the plugin in steps 1 and 2 to be the same path.

- If you do not need to deploy the plugin on all TiDB nodes, you can specify the node (the ID shown by the tiup cluster display) with the -N parameter.

Deploy the plugin during installation

Similar to steps in “Add the plugin to an existing cluster”. You need to configure `plugin.load` and `plugin.dir` before deployment, and run `tiup cluster exec` and `tiup cluster push` before starting the cluster.

Scale out

Similar to steps in “Add the plugin to an existing cluster”. You need to specify the node that is scaled out (the ID shown by the tiup cluster display) with the -N parameter.

Note that you need to upload the plugin file (whitelist-1.so in the above example) to the target server while the scaling is in progress, and make sure you finish uploading before the scaling ends. Otherwise an error will occur.

If an error occurs during scaling, you can perform the following to reinstall the plug-in.

1. Scale out the cluster again.
2. Upload the plug-in file after the scaling is completed.
3. Run the `tiup cluster start cluster_name command` to restart the cluster.

Upgrade the cluster

1. Delete the old plugin.

```
tiup cluster exec cluster_name --command "rm {{.DeployDir}}/plugin/whitelist-1.so" -R tidb
```

2. Upload the new plugin to all target servers. Use the same name to save you from changing the configurations.

```
tiup cluster push cluster_name ~/Download/whitelist-1_new.so
{{.DeployDir}}/plugin/whitelist-1.so -R tidb
```

Note: The domain name cannot contain "-" (such as `tidb-server1`). Otherwise the command will fail to run.

3. Upgrade the cluster.

```
tiup cluster upgrade cluster_name version
```

View plugin status

You can view the current plugin loading and enabling status by running the `show plugins` command in MySQL:

- Name: plugin name
- Status: plugin status, including the loading status (uninitialized, ready) and the enabling status (enable, disable)
- Type: plugin type
- Library: location of the plugin binary file
- Version: plugin version

Enable or disable the plugin

After the allowlist plugin is enabled, TiDB filters users based on user IP addresses. Only IP addresses in the allowlist are allowed to log in to TiDB. You can enable or disable the plugin on any TiDB node, because the setting applies to all TiDB nodes in the TiDB cluster.

Run the following command to enable the allowlist plugin:

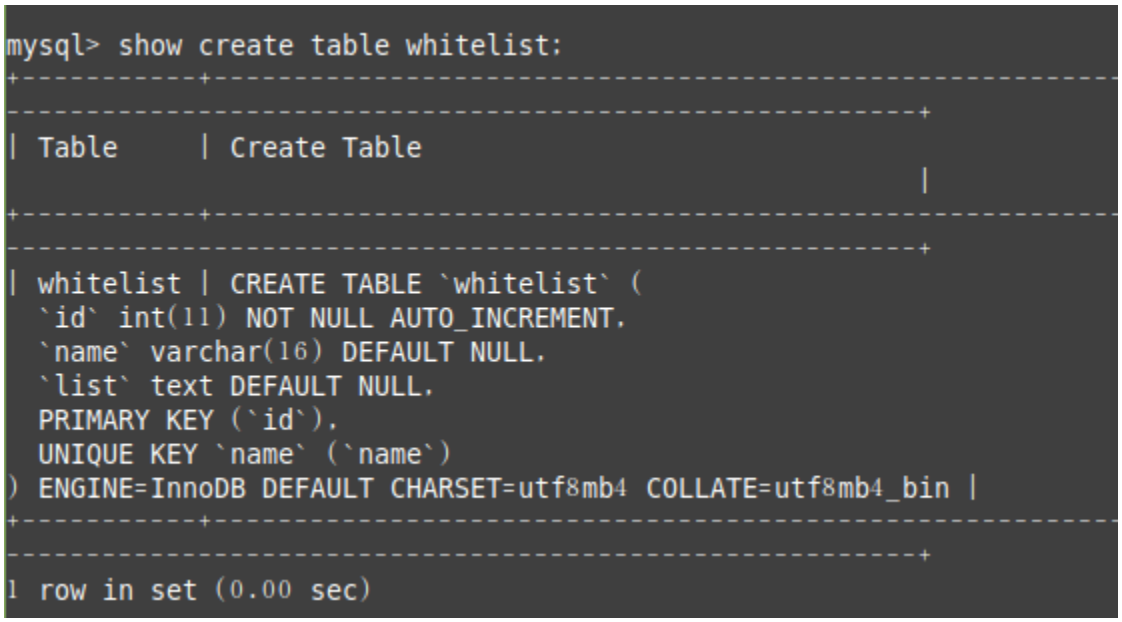
```
mysql> admin plugins enable whitelist;
```

To allow access of all IP addresses, you can disable the allowlist plugin. Run the following command to disable the allowlist plugin:

```
mysql> admin plugins disable whitelist;
```

Add data to the allowlist

After the allowlist plugin is enabled, TiDB will read the allowlist from the `mysql.whitelist` system table.



The allowlist table lists IP addresses in strings with each separated by a single comma. The IP segments are expressed in CIDR format, such as 192.0.2.1/24. For details, see func ParseCIDR.

The allowlist plugin is generally used to allow or deny Web access. To verify whether the plugin is functional, take the following steps:

1. Add an allowlist record to allow IP addresses in the 192.0.2.0 and 127.0.0.0 network segments to access the server:

```
mysql> insert into mysql.whitelist (name, list) values ('default', '192.0.2.1/24,127.0.0.1/24');
Query OK, 1 row affected (0.04 sec)
```

2. After the configuration takes effect, an error is reported when accessing TiDB by using a non-allowlisted machine:

```
➜  bin git:(master) mysql -h zenlife.tk -P 4000 -u root
ERROR 2013 (HY000): Lost connection to MySQL server at 'reading initial communication packet', system error: 0
```