
Tranalyzer2

clickhouseSink



ClickHouse



Tranalyzer Development Team

Contents

1	clickhouseSink	1
1.1	Description	1
1.2	Dependencies	1
1.3	Configuration Flags	2
1.4	Example	3

1 clickhouseSink

1.1 Description

The clickhouseSink plugin outputs flows to a ClickHouse database.

1.2 Dependencies

1.2.1 External Libraries

This plugin depends on the **clickhouse-cpp** and **clickhouse** libraries.

On macOS, the **clickhouse-cpp** library can be installed with Brew¹ (see below). Unfortunately, there is no package for the **clickhouse-cpp** library on Linux, but you can install it from source with the following commands:

```
$ git clone https://github.com/clickhouse/clickhouse-cpp
$ cd clickhouse-cpp
$ mkdir build
$ cd build
$ cmake ..
$ make
$ sudo make install
$ sudo ldconfig
```

Make sure `/usr/local/lib/` is in your library path:

```
$ ldconfig -p | tail -n +2 | grep -o '/.*/' | sort -u
```

If `/usr/local/lib/` is **NOT** in your library path, you can add it with

```
$ echo "/usr/local/lib/" | sudo tee -a /etc/ld.so.conf.d/mylibs.conf
$ sudo ldconfig
```

ClickHouse packages are available, but repositories must be added in most cases:

1.2.2 Ubuntu

Make sure to run **ALL** the commands below. Otherwise an old version of the clickhouse server (incompatible with the plugin) will be installed.

```
$ sudo apt install libabsl-dev
$ sudo apt-get install apt-transport-https ca-certificates dirmngr
$ sudo apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv E0C56BD4
$ echo "deb https://repo.clickhouse.com/deb/stable/ main/" | sudo tee \
    /etc/apt/sources.list.d/clickhouse.list
$ sudo apt-get update
$ sudo apt-get install -y clickhouse-server clickhouse-client
$ sudo service clickhouse-server start
```

¹Brew is a packet manager for macOS that can be found here: <https://brew.sh>

1.2.3 CentOS or RedHat

```
$ sudo yum install abseil-cpp
$ sudo yum install yum-utils
$ sudo rpm --import https://repo.clickhouse.com/CLICKHOUSE-KEY.GPG
$ sudo yum-config-manager --add-repo https://repo.clickhouse.com/rpm/clickhouse.repo
$ sudo yum install clickhouse-server clickhouse-client
$ sudo /etc/init.d/clickhouse-server start
```

1.2.4 Arch Linux

Replace yay with your favorite AUR helper.

```
$ sudo pacman -S abseil-cpp
$ yay -S clickhouse-server-bin clickhouse-client-bin clickhouse-common-static-bin
$ sudo systemctl start clickhouse-server
```

1.2.5 openSUSE

```
$ sudo zypper install abseil-cpp
$ sudo zypper install clickhouse
```

1.2.6 macOS

```
$ brew install clickhouse-cpp
$ wget 'https://builds.clickhouse.com/master/macos/clickhouse'
$ chmod a+x ./clickhouse
$ ./clickhouse
```

1.2.7 Core Configuration

This plugin requires the following core configuration:

- `$T2HOME/tranalyzer2/src/tranalyzer.h`:
 - `BLOCK_BUF=0`

1.3 Configuration Flags

The following flags can be used to control the output of the plugin:

Name	Default	Description
CLICKHOUSE_OVERWRITE_DB	2	0: abort if DB already exists 1: overwrite DB if it already exists 2: reuse DB if it already exists
CLICKHOUSE_OVERWRITE_TABLE	2	0: abort if table already exists 1: overwrite table if it already exists 2: append to table if it already exists
CLICKHOUSE_TRANSACTION_NFLOWS	10000	0: one transaction > 0: one transaction every n flows

Name	Default	Description
CLICKHOUSE_HOST	"127.0.0.1"	Address of the database
CLICKHOUSE_DBPORT	9000	Port the DB is listening to
CLICKHOUSE_USER	"default"	Username to connect to DB
CLICKHOUSE_PASS	" "	Password to connect to DB
CLICKHOUSE_DBNAME	"tranalyzer"	Name of the database
CLICKHOUSE_TABLE_NAME	"flow"	Name of the table

1.3.1 Environment Variable Configuration Flags

The following configuration flags can also be configured with environment variables (ENVCTRL>0):

- CLICKHOUSE_HOST
- CLICKHOUSE_DBPORT
- CLICKHOUSE_USER
- CLICKHOUSE_PASSWORD
- CLICKHOUSE_DBNAME
- CLICKHOUSE_TABLE_NAME

1.4 Example

```
# Run Tranalyzer
$ t2 -r file.pcap

# Connect to the ClickHouse database
$ clickhouse-client -d tranalyzer

# Number of flows
:) SELECT count(*) FROM flow;

# 10 first srcIP/dstIP pairs
:) SELECT "srcIP", "dstIP" FROM flow LIMIT 10;

# All flows from 1.2.3.4 to 1.2.3.5
:) SELECT * FROM flow WHERE "srcIP.1" = 4 AND "srcIP.2" = toIPv6('1.2.3.4') AND
    "dstIP.1" = 4 AND "dstIP.2" = toIPv6('1.2.3.5');
```

For examples of more complex queries, have a look in \$T2HOME/scripts/t2fm/clickhouse/.

1.4.1 Clean up an existing database

```
# Connect to the ClickHouse database
$ clickhouse-client
```

```
# Drop the database
:) DROP DATABASE tranalyzer;
```