

# ClickHouse (сбор событий аудита БД)

Информация, приведенная на данной странице, является разработкой команды pre-sales и/или community KUMA и **НЕ** является официальной рекомендацией вендора.

Для настройки базового аудита Clickhouse понадобится:

1. Для логирования обычных запросов (в том числе grant, create, drop) включить логирование в основном файле (по умолчанию включено):

```
/etc/clickhouse-server/config.xml
```

в секции logger как минимум необходимо задать формат information

```
#####  
<clickhouse>  
  <logger>  
    <level>information</level>  
#####
```

2. Для логирования подключений, таких как:

- Login пользователя
- Failure logon
- Logout пользователя

потребуется создать отдельный файл

```
/etc/clickhouse-server/config.d/session_log.xml
```

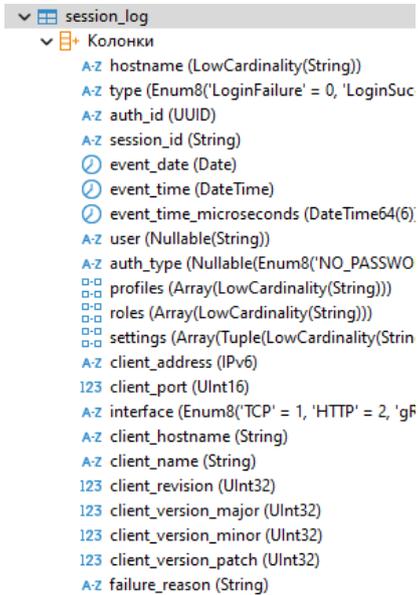
с содержимым:

```
<clickhouse>  
  <session_log>  
    <database>system</database>  
    <table>session_log</table>
```

```
<flush_interval_milliseconds>7500</flush_interval_milliseconds>
</session_log>
</clickhouse>
```

### 3. Перезапустить службу командой `systemctl restart clickhouse-server`

В базе `system` появится новая таблица со следующими колонками:



The screenshot shows the schema for the `session_log` table in ClickHouse. The columns are listed with their data types and constraints:

- `hostname` (LowCardinality(String))
- `type` (Enum8('LoginFailure' = 0, 'LoginSuc'))
- `auth_id` (UUID)
- `session_id` (String)
- `event_date` (Date)
- `event_time` (DateTime)
- `event_time_microseconds` (DateTime64(6))
- `user` (Nullable(String))
- `auth_type` (Nullable(Enum8('NO\_PASSWO')))
- `profiles` (Array(LowCardinality(String)))
- `roles` (Array(LowCardinality(String)))
- `settings` (Array(Tuple(LowCardinality(Strin)))
- `client_address` (IPv6)
- `client_port` (UInt16)
- `interface` (Enum8('TCP' = 1, 'HTTP' = 2, 'gF'))
- `client_hostname` (String)
- `client_name` (String)
- `client_revision` (UInt32)
- `client_version_major` (UInt32)
- `client_version_minor` (UInt32)
- `client_version_patch` (UInt32)
- `failure_reason` (String)

Создаём пользователя для подключения к БД для KUMA и выдаём ему необходимые права следующим командами (с использованием ранее настроенного аудита):

### Шаг 1. Создание пользователя для коллектора KUMA

```
CREATE USER kuma HOST IP '10.10.10.1/32' IDENTIFIED WITH sha256_password BY 'supersecretpassword';
```

#### Шаг 2. Создание VIEW для вывода нескольких запросов

```
CREATE VIEW combined_logs AS
```

```
SELECT event_time AS timestamp,
```

```
       query_kind AS deviceAction,
```

```
       user AS userName,
```

```
       toString(initial_address) AS sourceAddress,
```

```
       exception AS msg,
```

```
       query AS requestUrl,
```

```
□ query_duration_ms AS dcs1,
```

```
□ memory_usage AS dcs2
```

```
FROM system.query_log
```

```
WHERE query_kind IN ('Grant', 'Create', 'Drop') OR query_duration_ms > '600000' OR memory_usage > '1000000000'
```

```
ORDER BY timestamp DESC
```

UNION ALL

SELECT

event\_time AS timestamp,  
type AS deviceAction,  
user AS userName,  
toString(client\_address) AS sourceAddress,  
failure\_reason AS msg,  
NULL AS request\_url,

NULL AS dcs1,

NULL AS dcs2

FROM

system.session\_log

WHERE

type = 'LoginFailure'

#### Шаг 3. Назначение прав на VIEW

GRANT SELECT ON combined\_logs TO kuma

Пример как выглядит вывод VIEW:

timestamp	deviceAction	userName	sourceAddress	msg	requestUrl	dcs1	dcs2
2025-02-24 16:58:51	LoginFailure	test	127.0.0.1	test: Authentication failed: password is incorrect, or there is no user with such name.			
2025-02-26 09:22:39	Grant	default	127.0.0.1		GRANT SELECT ON combined_logs TO remote	0	0
2025-02-26 09:22:39	Grant	default	127.0.0.1		GRANT SELECT ON combined_logs TO remote	0	0
2025-02-26 09:22:39	Create	default	127.0.0.1		CREATE VIEW combined_logs AS SELECT event_time AS timestamp, user AS userName, query_kind AS de	0	0
2025-02-26 09:22:39	Create	default	127.0.0.1		CREATE VIEW combined_logs AS SELECT event_time AS timestamp, user AS userName, query_kind AS de	9	0
2025-02-26 09:21:12	Create	default	127.0.0.1	Code 386. DB-Exception: There is no supertype for types String, IPv6 because some of them are StringFixedStringEnum and some of them are not. (NO_COMMON_TYPE) (version 25.1.2.3 (official build))	CREATE VIEW combined_logs AS SELECT event_time AS timestamp, user AS userName, query_kind AS de	2	0
2025-02-26 09:20:01	Create	default	127.0.0.1	Code 386. DB-Exception: There is no supertype for types String, IPv6 because some of them are StringFixedStringEnum and some of them are not. (NO_COMMON_TYPE) (version 25.1.2.3 (official build))	CREATE VIEW combined_logs AS SELECT event_time AS timestamp, user AS userName, query_kind AS de	2	0
2025-02-26 09:19:09	Create	default	127.0.0.1	Code 386. DB-Exception: There is no supertype for types IPv6, String because some of them are StringFixedStringEnum and some of them are not. (NO_COMMON_TYPE) (version 25.1.2.3 (official build))	CREATE VIEW combined_logs AS SELECT event_time AS timestamp, user AS userName, query_kind AS de	2	0
2025-02-26 09:18:59	Create	default	127.0.0.1	Code 386. DB-Exception: There is no supertype for types IPv6, String because some of them are StringFixedStringEnum and some of them are not. (NO_COMMON_TYPE) (version 25.1.2.3 (official build))	CREATE VIEW combined_logs AS SELECT event_time AS timestamp, user AS userName, query_kind AS devi	9	0
2025-02-25 16:30:35	Drop	default	127.0.0.1		DROP USER admin1	0	0
2025-02-25 16:30:35	Drop	default	127.0.0.1		DROP USER admin1	0	0
2025-02-25 16:30:34	Create	default	127.0.0.1		CREATE USER admin1 IDENTIFIED	0	0
2025-02-25 16:30:24	Create	default	127.0.0.1		CREATE USER admin1 IDENTIFIED	0	0
2025-02-25 16:30:18	Create	default	127.0.0.1		CREATE USER admin IDENTIFIED	175	0
2025-02-25 16:21:58	Create	default	127.0.0.1	Code 493. DB-Exception: user 'timeweb_admin1' cannot insert because user 'admin1' already exists in local_directory. (ACCESS_DENIED_ALREADY_EXISTS) (version 25.1.2.3 (official build))	CREATE USER admin IDENTIFIED	3	0

Настройка инстанса завершена, можно приступать к подключению логов в KUMA.

Итого данной View мы выводим следующие события:

- Login пользователя
- Failure logon
- Logout пользователя
- Длительный запрос в базу (более 10 минут, как пример)(условие query\_duration\_ms > '600000' в запросе )
- Большое потребление памяти при запросе (более 1 Гб, как пример)(условие memory\_usage > '1000000000')
- Создание пользователя
- Назначение прав пользователю
- Удаление пользователя

В KUMA необходимо создать коллектор с транспортом sql (плейсхолдер для Clickhouse - ?) и параметрами как на скриншоте:

# Редактирование коллектора

Подключение источников

Транспорт

Парсинг событий

Фильтрация событий

Агрегация событий

Обогащение событий

Маршрутизация

Проверка параметров

## Транспорт

Подключите источник, от которого хотите получать события. Подробнее см. [в онлайн-справке](#).

Основные параметры

Дополнительные параметры

Коннектор

Создать

Тип\* ⓘ

sql

Запрос по умолчанию\*

```
SELECT * FROM combined_logs
WHERE timestamp > ?
```

Переключаться к БД каждый раз при отправке запроса



Интервал запросов, сек.

60

## Соединение

Тип базы данных\*

Clickhouse

URL\*

clickhouse:// 10.10.10.10:9000

Авторизация\*

Обычная

Секрет\*

clickhouse\_kuma

Режим TLS ⓘ

Выключено

Столбец идентификатора\*

timestamp

Начальное значение идентификатора\*

2025-02-01 19:46:40

Запрос ⓘ

Добавьте запрос, который следует использовать вместо запроса по умолчанию

Интервал запросов, сек.

0

+ Добавить подключение

Выбираем нормализатор Clickhouse (доступен в Community pack), добавляем необходимые точки назначения и инсталлируем службу коллектора.

Либо, если нам **не нужны события входа в БД** можем использовать запрос только в таблицу по умолчанию:

```
###Шаг 1. Создание пользователя для коллектора KUMA
```

```
CREATE USER kuma HOST IP '10.10.10.1/32' IDENTIFIED WITH sha256_password BY 'supersecretpassword';
```

```
####Шаг 2. Назначение прав на выполнение SELECT к system.query_log
```

```
GRANT SELECT ON system.query_log TO kuma
```

```
####Шаг 3. Используем запрос для KUMA коллектора
```

```
SELECT event_time AS timestamp,
```

```
    query_kind AS deviceAction,
```

```
    user AS userName,
```

```
    toString(initial_address) AS sourceAddress,
```

```
    exception AS msg,
```

```
    query AS requestUrl,
```

```
    query_duration_ms AS dcs1,
```

```
    memory_usage AS dcs2
```

```
FROM system.query_log
```

```
WHERE query_kind IN ('Grant', 'Create', 'Drop') OR query_duration_ms > '600000' OR memory_usage > '1000000000' AND timestamp > ?
```

```
ORDER BY timestamp DESC
```

---

Revision #12

Created 31 January 2025 17:01:37 by Anton

Updated 24 March 2025 10:34:58 by Boris RZR