ClickHouse для инженеров и архитекторов БД

RBAC контроль доступа, квоты и ограничения

otus.ru

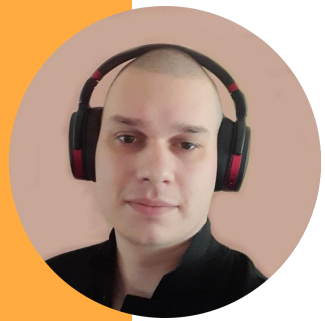
Проверить, идет ли запись

Меня хорошо видно && слышно?

Ставим “+”, если все хорошо
“-”, если есть проблемы

Тема вебинара

RBAC контроль доступа, квоты и ограничения



Константин Трофимов

Senior SRE / ClickHouse DBA в VK

Занимаюсь эксплуатацией ClickHouse с первых версий: 5 лет в VK, до этого в AdNow, до этого занимался Vertica. Сотни серверов, десятки кластеров, десятки петабайт данных.

Правила вебинара

Активно
участвуем

Off-topic обсуждаем
в учебной группе
#OTUS ClickHouse-2024-08

Задаем вопрос
в чат или голосом

Вопросы вижу в чате,
могу ответить не сразу

Условные обозначения

Индивидуально

Время, необходимое
на активность

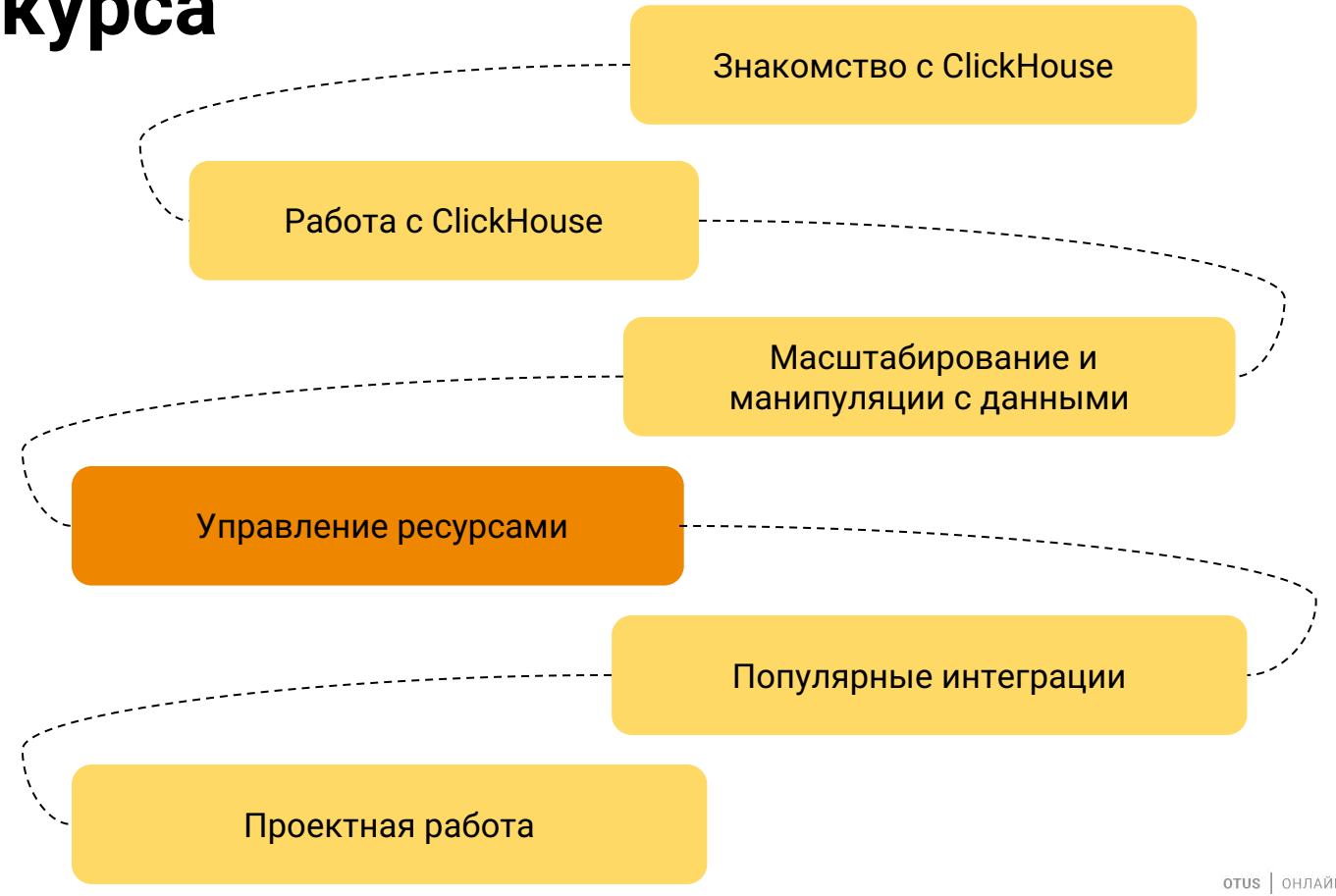
Пишем в чат

Говорим голосом

Документ

Ответьте себе или
задайте вопрос

Карта курса


Знакомство с ClickHouse
Работа с ClickHouse
Масштабирование и манипуляции с данными
Управление ресурсами
Популярные интеграции
Проектная работа

Темы модуля «Управление ресурсами»

1 (18). RBAC контроль доступа, квоты и ограничения

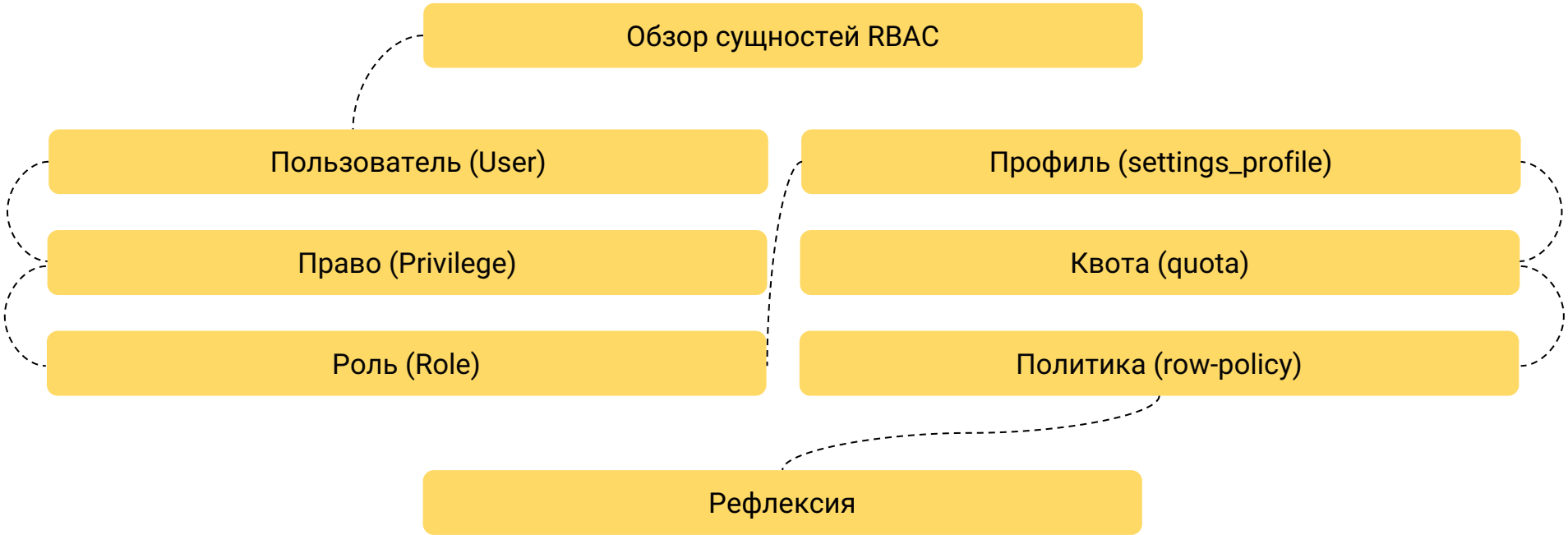
2 (19). Storage Policy и резервное копирование

3 (20). Метрики и мониторинг. Логирование

4 (21). Профилирование запросов

5 (17). Сессия Q&A

Маршрут вебинара



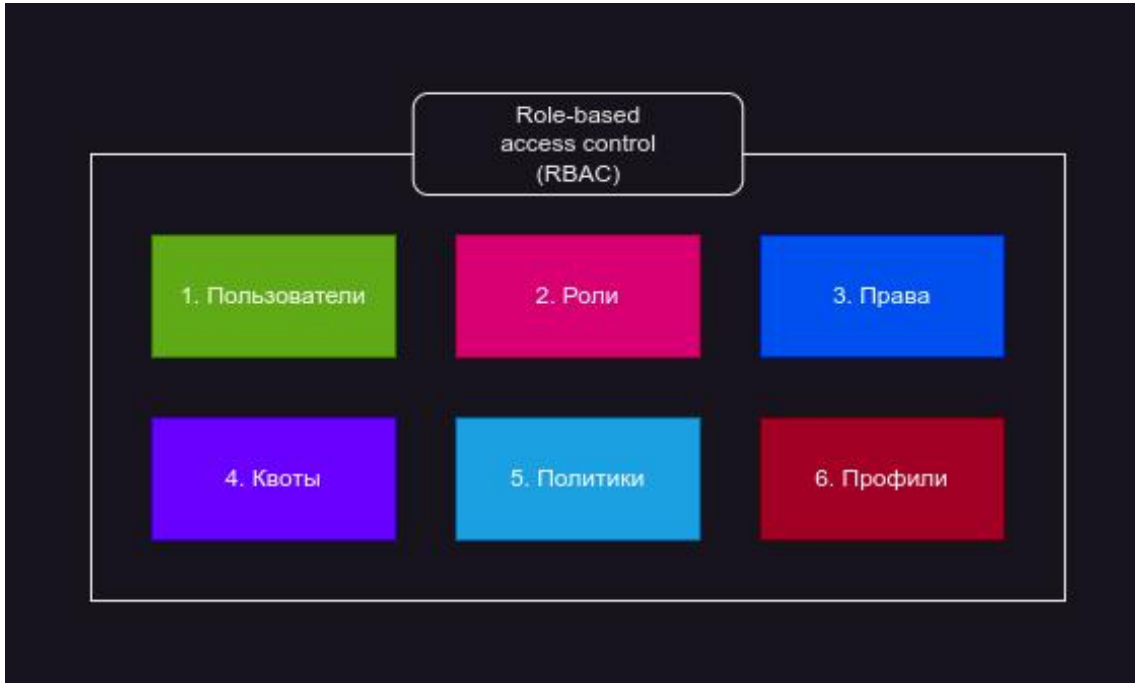
Обзор сущностей RBAC

Что такое RBAC в ClickHouse

Модель доступа, управляемая из SQL, включающая

6 взаимосвязанных типов сущностей:

- пользователь (user)
- роль (role)
- право (privilege)
- профиль (settings_profile)
- квота (quota)
- политика (row-policy)



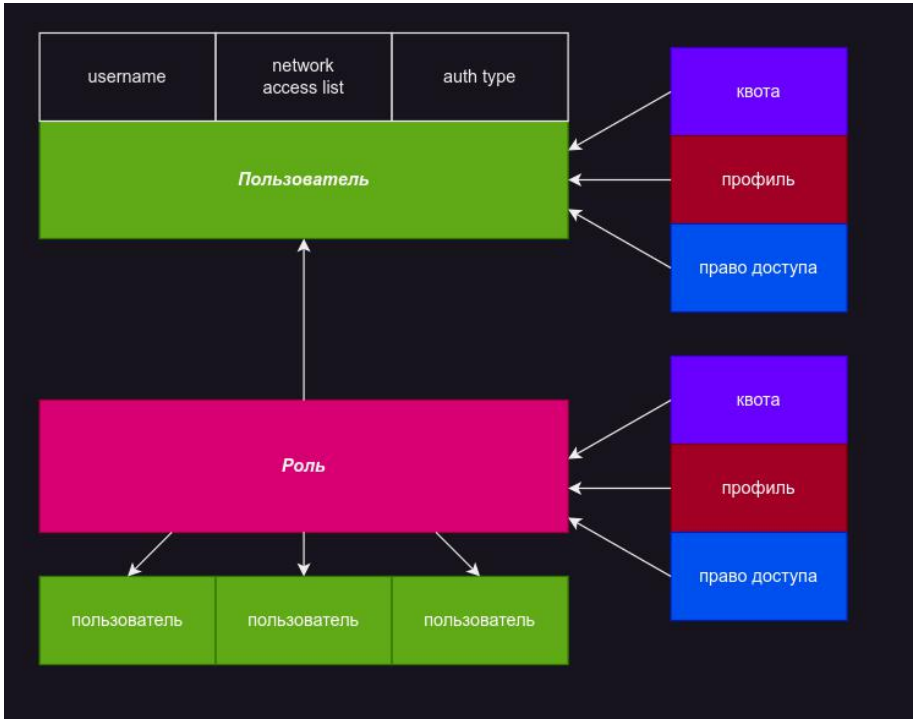
Пользователи и роли

Пользователь (user) - способ доступа к ClickHouse, используя имя, и некоторый из способов авторизации. Обладает свойствами:

- имя пользователя (username)
- список сетей, адресов, с которых разрешен доступ, в том числе можно указывать DNS-именами (network access list)
- способ авторизации (auth type), например, и наиболее часто используется вариант авторизации паролем

К пользователю можно применять права доступа, профили настроек, квоты, которые являются самостоятельными сущностями.

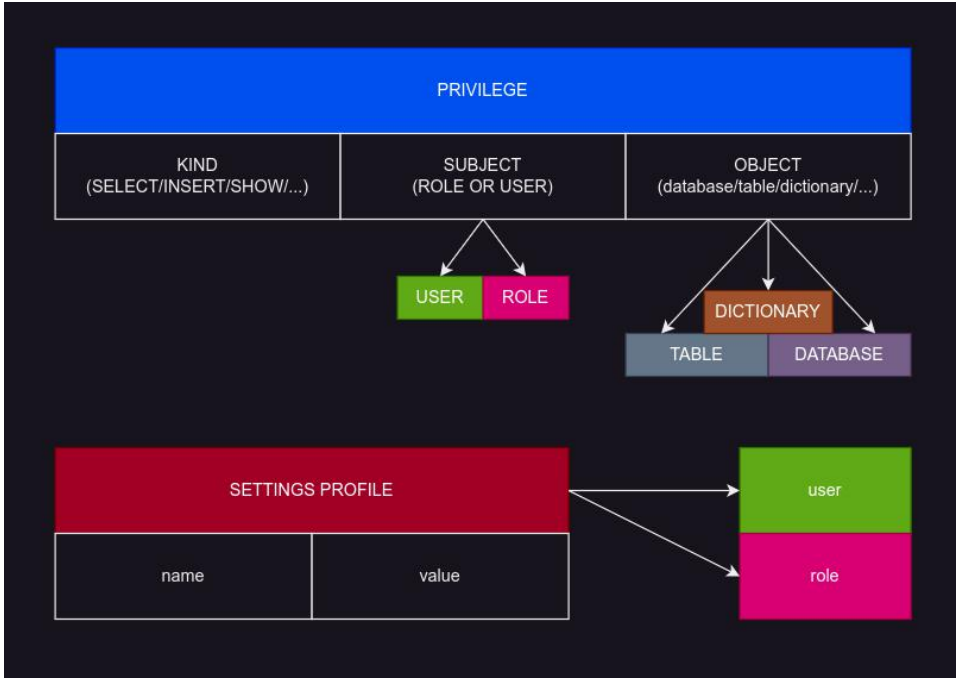
Роль (role) не обладает собственными свойствами, только именем, однако к роли можно применять то же самое что и к пользователю, и потом роль применять на ряд пользователей. Роль используется для логического объединения доступов в группы.



Права доступа и профили настроек

Право доступа (privilege) - разрешающее правило, применяемое к ***пользователю*** или ***роли***, дающее возможнось делать конкретный вид операций (INSERT/SELECT/ALTER/и т.д.), над конкретным объектом, таким как база данных, таблица, словарь.

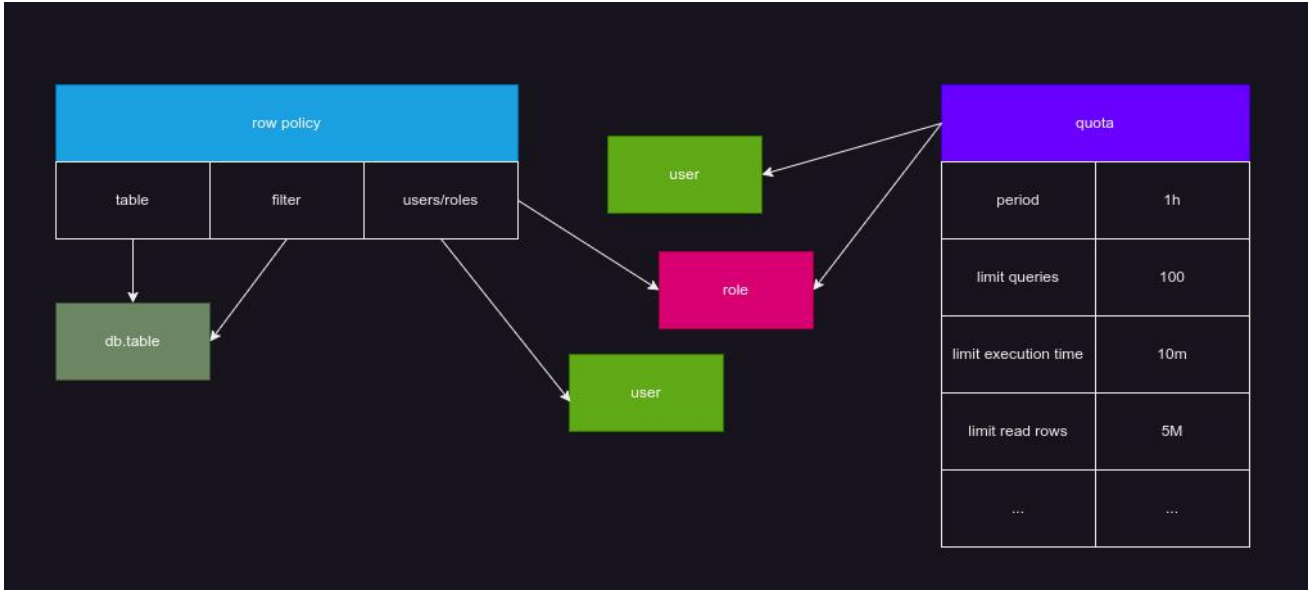
Профиль настроек (SETTINGS PROFILE) - коллекция настроек, применямых к ***роли*** или ***пользователю***, набор пар ключ-значение, позволяет например включить/выключить логирование запросов, или наложить ограничения на сложность запросов.



Квоты и политики

Квота (quota) - ограничение в потреблении ресурсов за интервал времени. Задается интервал, например 1 час. Задается ограничение, например на кол-во запросов. Пользователь сделавший запросов в количестве равном ограничению, на очередной запрос получит ошибку «квота исчерпана», и так до конца часа. С начала нового часа счетчик сделанных запросов сбрасывается и пользователь снова может расходовать свою квоту. Применяется к ролям и пользователям.

Политика (row-policy) - префильтр, срабатывающий при выборке данных, пропускающий данные не попадающие под фильтр, ещё до аггрегации. Позволяет скрыть данные от пользователей по условию, например ‘колонка_отдел’=’менеджер’ покажет только те строки, в которых соблюдается такое условие. Применяется к ролям и пользователям.



Вопросы?

Ставим “+”,
если вопросы есть

Ставим “–”,
если вопросов нет

Пользователь (User)

Управление пользователями

Пользователь (user) - способ доступа к ClickHouse, используя имя, и некоторый из способов авторизации.

создание пользователя

CREATE USER [IF NOT EXISTS | OR REPLACE] name1 [, name2 [,...]] ***параметры пользователя***

изменение пользователя

ALTER USER [IF NOT EXISTS] name1 [, name2 [,...]] ***параметры пользователя***

удаление пользователя

DROP USER [IF EXISTS] name [,...]

список пользователей

SHOW USERS // покажет и пользователей созданных через xml-конфигурацию

или

SELECT name FROM system.users WHERE storage = 'local_directory' // покажет только RBAC-пользователей

Параметры пользователя для CREATE USER и ALTER USER

Список доступных параметров:

1) авторизация

IDENTIFIED WITH plaintext_password // (или другой способ авторизации)

BY 'my secret password' // (или WITH ‘другой идентификатор’)

2) список разрешенных сетей

HOST IP 'address'

3) необязательные параметры

[VALID UNTIL datetime] // будет недействителен после заданного времени

[IN access_storage_type] // пока не используется, можно не указывать

[DEFAULT ROLE role [,...]] // роль, если не назначено никаких ролей

[DEFAULT DATABASE database | NONE] // бд для подключения по умолчанию

[GRANTEES {user | role | ANY | NONE} [,...] [EXCEPT {user | role} [,...]]] // кому может передавать свои права

[SETTINGS variable [= value] [MIN [=] min_value] [MAX [=] max_value] // настройки встроенного в пользователь профиля

[SETTINGS PROFILE 'profile_name'] // применение к пользователю готового профиля настроек

Авторизация

1) без авторизации, пустит с любым паролем
   NOT IDENTIFIED
   IDENTIFIED WITH no_password
2) парольная авторизация, при создании передается пароль, в ClickHouse хранится sha256 хеш
   IDENTIFIED BY 'qwerty'
   IDENTIFIED WITH plaintext_password BY 'qwerty'
   IDENTIFIED WITH sha256_password BY 'qwerty'
2.1) или другой хеш
   IDENTIFIED WITH double_sha1_password BY 'qwerty'
   IDENTIFIED WITH bcrypt_password BY 'qwerty'
3) парольная авторизация, при создании передается хеш, хранится он же
   IDENTIFIED WITH sha256_hash BY 'hash' or IDENTIFIED WITH sha256_hash BY 'hash' SALT 'salt'
   IDENTIFIED WITH double_sha1_hash BY 'hash'
   IDENTIFIED WITH bcrypt_hash BY 'hash'
4) авторизация по ssh-ключу (взаимодействие с ssh-agent в Linux)
   IDENTIFIED WITH ssh_key BY KEY 'public_key' TYPE 'ssh-rsa', KEY 'another_public_key' TYPE 'ssh-ed25519'
5) авторизация по SSL CN сертификата, с которым подключается клиент (требует openssl настроек сервера, включая CA-верификацию)
   IDENTIFIED WITH ssl_certificate CN 'mysite.com:user'
6) авторизация при помощи популярных служб авторизации
   IDENTIFIED WITH ldap SERVER 'server_name'
   IDENTIFIED WITH kerberos or IDENTIFIED WITH kerberos REALM 'realm'
7) авторизация внешним сервисом авторизации (только http basic, требует настройки <http_authentication_servers> в конфигурации ClickHouse)
   IDENTIFIED WITH http SERVER 'http_server'
   IDENTIFIED WITH http SERVER 'http_server' SCHEME 'basic'

список разрешенных сетей

1) только локалхост (ipv4 и ipv6)

   HOST LOCAL

3) по IP адресу (можно использовать подсети)

   HOST IP ‘address’

   HOST IP ‘network/mask’

4) по PTR-записи

   HOST NAME ‘name’

   HOST REGEXP ‘regexp’

   HOST LIKE ‘pattern’

5) отовсюду и ни откуда

   HOST [ANY | NONE]

можно использовать несколько через запятую, например

   HOST LOCAL, IP ‘10.0.12.0/24’, LIKE ‘%.developers-xxx.deparment.my.org’

Как получить список текущих RBAC-пользователей

```
SELECT
    name,
    auth_type,
    auth_params
FROM system.users
WHERE storage = 'local_directory'

Query id: cb1f111a-1c6e-4d6d-863e-941afed87c9a

┌─name──────────────────┬─auth_type──────────┬─auth_params──────────────────────────────────────┐
│ anonymous             │ no_password        │ {}                                               │
│ partner-project.zone  │ ssl_certificate    │ {"common_names":["partner-project.zone"]}        │
│ backend               │ plaintext_password │ {}                                               │
│ developer             │ ssh_key            │ {}                                               │
└───────────────────────┴────────────────────┴──────────────────────────────────────────────────┘

4 rows in set. Elapsed: 0.005 sec.
```

Примеры

```
CREATE USER developer
   IDENTIFIED WITH ssh_key
   BY KEY
`AAAAB3NzaC1yc2EAAAADAQABAAABAQDtaNeHbX0pIoNNeCjKCfXAhFf/wLzsdR+IZ8ujsGIlu4/gP1UH8P
Dr61x7/5KrCkycuR/kIqSePCt56lmHlLxvurhWR4VITuZC427GtuF4c8b+pEDulam7r/dmuYvcsjIVFywg0KMilNw
N575OCJZiZU/5TTJRqe7uZipPV87my2D7R5PUVfS1OrvmphS/YF2aepd5V8aheGuHgEeOcrUB0B2lsAUVBa
hUahN7my+QFpv6zJDr3hsdl8d/zS1P3W/jNAieQxbMW4oWuyXJZCbt7ded5oIEzVHsJtOTgP6ok31uOyy20dJ
buE9ReqCYdifgOQvkmG3mcyl7Ixiod7GF`
TYPE `ssh-rsa`

CREATE USER backend IDENTIFIED WITH plaintext_password BY 'qwerty'

CREATE USER `partner-project.zone` IDENTIFIED WITH ssl_certificate CN 'partner-project.zone'

CREATE USER anonymous NOT IDENTIFIED
```

Вопросы?

Ставим “+”,
если вопросы есть

Ставим “–”,
если вопросов нет

Роль (Role)

Управление ролями

Роль (role) не обладает собственными свойствами, только именем, однако к роли можно применять то же самое что и к пользователю, и потом роль применять на ряд пользователей. Роль используется для логического объединения доступов в группы.

Пользователь получает права роли только после выполнения SET ROLE ‘роль’, для роли по умолчанию выполняется SET ROLE DEFAULT автоматически

Синтаксис SET ROLE:

SET ROLE DEFAULT - взять роль по умолчанию

SET ROLE NONE - отказаться от всех ролей

SET ROLE ALL - взять все роли

SET ROLE role [,...] - список ролей

... EXCEPT role [,...] - кроме ролей

Управление ролями

Создать роль: ***CREATE ROLE name***

Удалить роль: ***DROP ROLE role***

Назначить роль пользователю: ***GRANT role TO user***

Отозвать роль у пользователя: ***REVOKE role FROM user***

установить роль по умолчанию пользователю

SET DEFAULT ROLE {NONE | role [,...] | ALL | ALL EXCEPT role [,...]} TO {user|CURRENT_USER} [,...]

Список текущих ролей

```
SELECT *
FROM system.roles

Query id: aaa3c6d6-2661-42cc-bf79-c20b6be55841

┌─name─────────┬─id───────────────────────────────────┬─storage─────────┐
│ permissive   │ b6444787-b67c-bef3-beb9-83b8ac55a9a3 │ local_directory │
└──────────────┴──────────────────────────────────────┴─────────────────┘

1 row in set. Elapsed: 0.002 sec.
```

Список кому выдана какая роль

```
SELECT *
FROM system.role_grants

Query id: e76fd877-bdec-41dc-9f61-342b2e6d0e4d

┌─user_name─┬─role_name─┬─granted_role_name─┬─granted_role_id──────────────────────┬─granted_role_is_default─┬─with_admin_option─┐
│ developer │ NULL      │ permissive        │ b6444787-b67c-bef3-beb9-83b8ac55a9a3 │                       1 │                 0 │
└───────────┴───────────┴───────────────────┴──────────────────────────────────────┴─────────────────────────┴───────────────────┘

1 row in set. Elapsed: 0.003 sec.
```

Вопросы?

Ставим “+”,
если вопросы есть

Ставим “–”,
если вопросов нет

Право (Privilege)

Управление

Выдача прав:

GRANT [ON CLUSTER cluster_name]
  privilege[(column_name [,...])] [,...]
ON {db.table|db.*|*.*|table|*}
TO {user | role | CURRENT_USER} [,...]
[WITH GRANT OPTION] [WITH REPLACE OPTION]

Отзыв прав:

REVOKE [ON CLUSTER cluster_name]
  privilege[(column_name [,...])] [,...]
ON {db.table|db.*|*.*|table|*}
FROM {user | CURRENT_USER} [,...] | ALL | ALL
EXCEPT {user | CURRENT_USER}

Виды прав

Полный список для текущей версии можно получить из типа колонки ***access_type*** таблицы ***system.grants***, запросом

```
SELECT privilege
FROM system.columns
ARRAY JOIN extractAll(type, '\'[^\']+\') AS privilege
WHERE (database = 'system') AND (table = 'grants') AND (name = 'access_type')
ORDER BY privilege DESC
```

наиболее популярные типы:

SELECT/INSERT/SHOW/dictGet для работы приложений и пользователей

ALTER/DROP/CREATE для миграций

Объекты на которые получаются права

База, выдается как ***ON db.****

Таблица, выдается как ***ON db.table***

Словарь, выдается как ***ON db.dict***

словари созданные без указания базы точечно награнтать нельзя

ON *.* - выдать на всё, например на все словари

Специальные операции, например «SYSTEM DROP DNS CACHE», обычно на них нет необходимости в разграничении прав и они выполняются из под пользователя с полными правами командой эксплуатации.

Специальные функции, например «FILE» позволяет использовать табличную функцию file(), «S3» позволяет использовать одноименную табличную функцию.

Примеры

```
GRANT SELECT, SHOW ON public_data.* TO anonymous;

GRANT INSERT, SELECT, SHOW ON prod_data.* TO backend;

GRANT SELECT, SHOW ON stage_data.* TO developer;

GRANT dictGet ON dicts.* TO backend,developer;
```

Список текущих прав

```
SELECT *
FROM system.grants
WHERE user_name IN (
    SELECT name
    FROM system.users
    WHERE storage = 'local_directory'
)

Query id: e9f2401e-5a97-4db0-acc6-871b31f75cd8

┌─user_name─┬─role_name─┬─access_type──────┬─database─┬─table─┬─column─┬─is_partial_revoke─┬─grant_option─┐
│ anonymous │ NULL      │ SYSTEM DROP CACHE │ NULL     │ NULL  │ NULL   │                 0 │            0 │
│ anonymous │ NULL      │ SHOW             │ public_data │ NULL  │ NULL   │                 0 │            0 │
│ anonymous │ NULL      │ SELECT           │ public_data │ NULL  │ NULL   │                 0 │            0 │
│ backend   │ NULL      │ dictGet          │ dicts    │ NULL  │ NULL   │                 0 │            0 │
│ developer │ NULL      │ dictGet          │ dicts    │ NULL  │ NULL   │                 0 │            0 │
└───────────┴───────────┴──────────────────┴──────────┴───────┴────────┴───────────────────┴──────────────┘

5 rows in set. Elapsed: 0.008 sec.
```

Вопросы?

Ставим “+”,
если вопросы есть

Ставим “–”,
если вопросов нет

Профиль (settings_profile)

Управление

создать

CREATE SETTINGS PROFILE [IF NOT EXISTS | OR REPLACE] name1 [ON CLUSTER cluster_name1]
[, name2 [ON CLUSTER cluster_name2] ...]
[SETTINGS variable [= value] [MIN [=] min_value] [MAX [=] max_value]
[CONST|READONLY|WRITABLE|CHANGEABLE_IN_READONLY] | INHERIT 'profile_name'] [,...]

ALTER с таким же синтаксисом

удалить

DROP SETTINGS PROFILE [IF EXISTS] name1

установить на пользователя/роль:

ALTER USER/ROLE ... PROFILE 'profile_name']

разбор

набор пар ключ=значение для настроек пользователя

[SETTINGS variable [= value] [MIN [=] min_value] [MAX [=] max_value]

унаследовать настройки из другого профиля

INHERIT 'profile_name'

Текущие профили можно посмотреть в ***system.settings_profiles***

полезные настройки

log_queries=1 логировать запросы

max_query_size - максимальный размер запроса
max_ast_elements - сложность запроса для парсера
max_bytes_before_external_group_by - порог по памяти для группировки на диске
max_bytes_before_external_sort - тоже самое для сортировки

max_result_rows, max_result_bytes - ограничения на результат запроса
max_execution_time - ограничение на время выполнения запроса

max_rows_in_join - ограничение на размер JOIN
max_bytes_in_set - ограничение на размер WHERE IN (....)

Вопросы?

Ставим “+”,
если вопросы есть

Ставим “–”,
если вопросов нет

Квота (quota)

Управление

создается/удаляется/изменяется запросами: ***CREATE/DROP/ALTER QUOTA name***

запросы CREATE и ALTER имеют одинаковый синтаксис

Синтаксис CREATE-запроса:

```
CREATE QUOTA [IF NOT EXISTS | OR REPLACE] name [ON CLUSTER cluster_name]
    [KEYED BY {user_name | ip_address | client_key | client_key,user_name | client_key,ip_address} | NOT KEYED]
    [FOR [RANDOMIZED] INTERVAL number {second | minute | hour | day | week | month | quarter | year}
        {MAX { {queries | query_selects | query_inserts | errors | result_rows | result_bytes | read_rows | read_bytes |
execution_time} = number } [,...] |
         NO LIMITS | TRACKING ONLY} [,...]]
    [TO {role [,...] | ALL | ALL EXCEPT role [,...]}]
```

CREATE QUOTA / разбор

отдельный счетчик на каждый IP-адрес/пользователя

[KEYED BY {user_name | ip_address | client_key | client_key,user_name | client_key,ip_address} | NOT KEYED]

интервал, можно указать в количестве секунд, можно использовать специальный тип INTERVAL как синтаксический сахар.

FOR [RANDOMIZED] INTERVAL number {second | minute | hour | day | week | month | quarter | year}

счетчики, можно указать несоклько через «,»

MAX { {queries | query_selects | query_inserts | errors | result_rows | result_bytes | read_rows | read_bytes | execution_time} = number }

не ограничивать, но вести счетчик

NO LIMITS | TRACKING ONLY

к пользователи, роли, к которым применяется квота

[TO {role [,...] | ALL | ALL EXCEPT role [,...]}]

список счетчиков по квотам

```
SELECT *
FROM system.quota_limits
FORMAT Vertical

Query id: 1c920be4-2aba-46cb-a48d-617bc080c571

Row 1:
──────
quota_name:             default
duration:               3600
is_randomized_interval: 0
max_queries:            NULL
max_query_selects:      NULL
max_query_inserts:      NULL
max_errors:             NULL
max_result_rows:        NULL
max_result_bytes:       NULL
max_read_rows:          NULL
max_read_bytes:         NULL
max_execution_time:     NULL
max_written_bytes:      NULL

1 row in set. Elapsed: 0.004 sec.
```

список квот

```
SELECT *
FROM system.quotas
FORMAT Vertical

Query id: 51c88ec2-31c3-4532-8b39-09db788f3739

Row 1:
──────
name:           default
id:             e66c72d8-fbd2-c174-0df3-7cbfd0c3d635
storage:        users_xml
keys:           ['user_name']
durations:      [3600]
apply_to_all:   0
apply_to_list:  ['default']
apply_to_except: []

1 row in set. Elapsed: 0.003 sec.
```

Вопросы?

Ставим “+”,
если вопросы есть

Ставим “–”,
если вопросов нет

Политика (row-policy)

Управление

Синтаксис CREATE

```
CREATE [ROW] POLICY [IF NOT EXISTS | OR REPLACE] policy_name1 [ON CLUSTER cluster_name1] ON
[db1.]table1|db1.*
      [, policy_name2 [ON CLUSTER cluster_name2] ON [db2.]table2|db2.* ...]
    USING condition
    [AS {PERMISSIVE | RESTRICTIVE}]
    [TO {role1 [, role2 ...] | ALL | ALL EXCEPT role1 [, role2 ...]}]
```

Синтаксис ALTER такой же

Синтаксис DROP:

```
DROP [ROW] POLICY [IF EXISTS] name [,...] ON [database.]table [,...] [ON CLUSTER cluster_name]
```

CREATE синтаксис

создается на таблицу или база.*
  policy_name1 [ON CLUSTER cluster_name1] ON [db1.]table1|db1.*

условие для показа данных
  USING condition
поддерживается два варианта:
  column = константа
  column IN (‘константа1’,’константа2’)

режим AND/OR
  [AS {PERMISSIVE | RESTRICTIVE}]
PERMISSIVE - должна быть соблюдена любая из политик
RESTRICTIVE - должны быть соблюдены все политики

к кому применяется политика
  [TO {role1 [, role2 ...] | ALL | ALL EXCEPT role1 [, role2 ...]}]

Текущие политики можно посмотреть в ***system.row_policies***

Вопросы?

Ставим “+”,
если вопросы есть

Ставим “–”,
если вопросов нет

Домашнее задание

1) Создать пользователя jhon с паролем «qwery»

2) Создать роль devs

3) Выдать роли devs права на SELECT на любую таблицу

4) выдать роль devs пользователю jhon

5) предоставить результаты SELECT из system-таблиц соответсвующих созданным сущностям

Вопросы?

Ставим “+”,
если вопросы есть

Ставим “–”,
если вопросов нет

Рефлексия

Рефлексия

С какими впечатлениями уходите с вебинара?

Как будете применять на практике то,
что узнали на вебинаре?

Следующий вебинар

3 декабря 2024 (вторник)

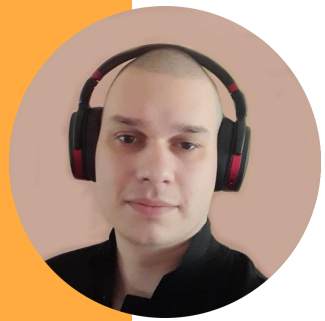
Storage Policy и резервное копирование

- Ссылка на вебинар будет в ЛК за 15 минут
- Материалы к занятию в ЛК — можно изучать
- Обязательный материал обозначен красной лентой

Заполните, пожалуйста, опрос о занятии по ссылке в чате

Спасибо за внимание!

Приходите на следующие вебинары



Senior SRE / ClickHouse DBA в VK

Занимаюсь эксплуатацией ClickHouse с первых версий: 5 лет в VK, до этого в AdNow, до этого занимался Vertica. Сотни серверов, десятки кластеров, десятки петабайт данных.