

1. Настройка репликации postgres.

- Вынес настройки данных из volume docker в файловую систему посредством mount.
- Настройки postgresql.conf
 - `max_wal_senders = 4` - макс кол-во процессов отправляющих WAL (n реплик +1)
 - `max_replication_slots = 4` # количество слотов репликации (n реплик +1)
- Создал роль replicator:
postgres=# create role replicator with login replication password 'pass';
- Получить адрес подсети докера
docker network inspect social_network | grep Subnet
> "Subnet": "172.19.0.0/16"
- Добавил запись IP подсети в pg_hba.conf
 - **host replication replicator 172.19.0.0/16 md5**
- Примонтировал в контейнер докера мастер базы пути к данным для реплик, чтобы была возможность сделать бекап.
volumes:
 - `./pg-data-master:/var/lib/postgresql/data`
 - `./pg-data-slave_1:/var/lib/postgresql/data2`
 - `./pg-data-slave_2:/var/lib/postgresql/data3`
- Сделал бекапы в директории реплик
pg_basebackup -U replicator -Xs -P -R -D /var/lib/postgresql/data2/
 - **-Xs**: транслировать WAL во время создания резервной копии. Это помогает избежать проблем с удалением файлов WAL при наличии большого набора данных
 - **-P**: показывать прогресс backup.
 - **-R**: pg_basebackup автоматически создает standby.signal и подготавливает postgresql.auto.conf с настройками соединения.

```
root@6b0d7a11dec4:/var/lib/postgresql/data2# pg_basebackup -h postgres -D /var/lib/postgresql/data2/ -v -P --wal-method=stream -U replicator
Password:
pg_basebackup: initiating base backup, waiting for checkpoint to complete
pg_basebackup: checkpoint completed
pg_basebackup: write-ahead log start point: 0/2F000028 on timeline 1
pg_basebackup: starting background WAL receiver
pg_basebackup: created temporary replication slot "pg_basebackup_437729"
442559/442559 KB (100%), 1/1 tablespace
pg_basebackup: write-ahead log end point: 0/2F000170
pg_basebackup: waiting for background process to finish streaming ...
pg_basebackup: syncing data to disk ...
pg_basebackup: renaming backup_manifest.tmp to backup_manifest
pg_basebackup: base backup completed
root@6b0d7a11dec4:/var/lib/postgresql/data2#
root@6b0d7a11dec4:/var/lib/postgresql/data2# pg_basebackup -h postgres -D /var/lib/postgresql/data3/ -v -P --wal-method=stream -U replicator
Password:
pg_basebackup: initiating base backup, waiting for checkpoint to complete
pg_basebackup: checkpoint completed
pg_basebackup: write-ahead log start point: 0/30000028 on timeline 1
pg_basebackup: starting background WAL receiver
pg_basebackup: created temporary replication slot "pg_basebackup_437734"
442559/442559 KB (100%), 1/1 tablespace
pg_basebackup: write-ahead log end point: 0/30000100
pg_basebackup: waiting for background process to finish streaming ...
pg_basebackup: syncing data to disk ...
pg_basebackup: renaming backup_manifest.tmp to backup_manifest
pg_basebackup: base backup completed
root@6b0d7a11dec4:/var/lib/postgresql/data2# ls
```

- Создал файл, чтобы реплики узнали что они реплики:
touch ./pg-data-slave_1/standby.signal
touch ./pg-data-slave_2/standby.signal
- Добавил строки подключения primary_conninfo для реплик в файлах postgresql.conf:
 - **primary_conninfo = 'host=postgres port=5432 user=replicator password=pass application_name=postgres_slave_1'**

- **primary_conninfo = 'host=postgres port=5432 user=replicator password=pass application_name=postgres_slave_2'**
- Включил синхронную репликацию на мастере postgresql.conf:
synchronous_commit = on synchronous_standby_names = 'FIRST 1 (postgres_slave_1, postgres_slave_2)'
- Перезагрузил конфиг мастера
psql select pg_reload_conf();
- **psql# select application_name, sync_state from pg_stat_replication;**

```
social_network=# select application_name, sync_state from pg_stat_replication;
 application_name | sync_state
-----+-----
 postgres_slave_2 | potential
 postgres_slave_1 | sync
(2 rows)

social_network=#
```

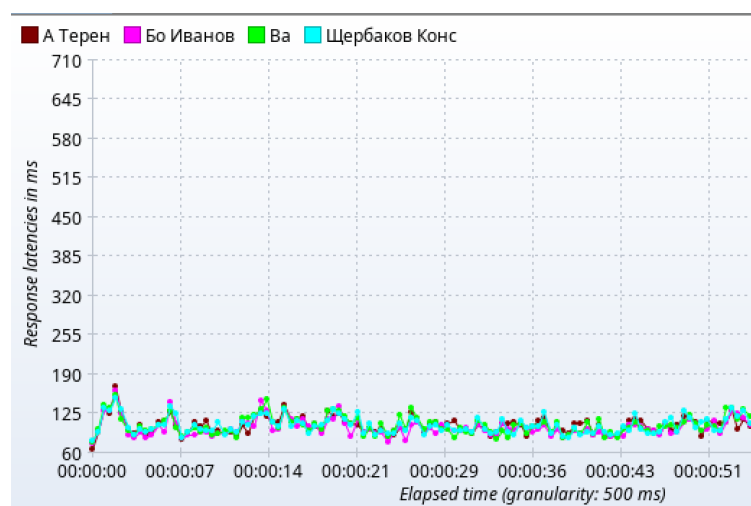
2. Нагрузочный тест до перевода методов на слейв

Для чистоты эксперимента я запустил генератор сидов тоесть операции на вставку и параллельно запустил нагрузочное тестирование метода /search.

Для нагрузочного тестирования использовалось 800 пользователей.

aggregate_report_statistics_before

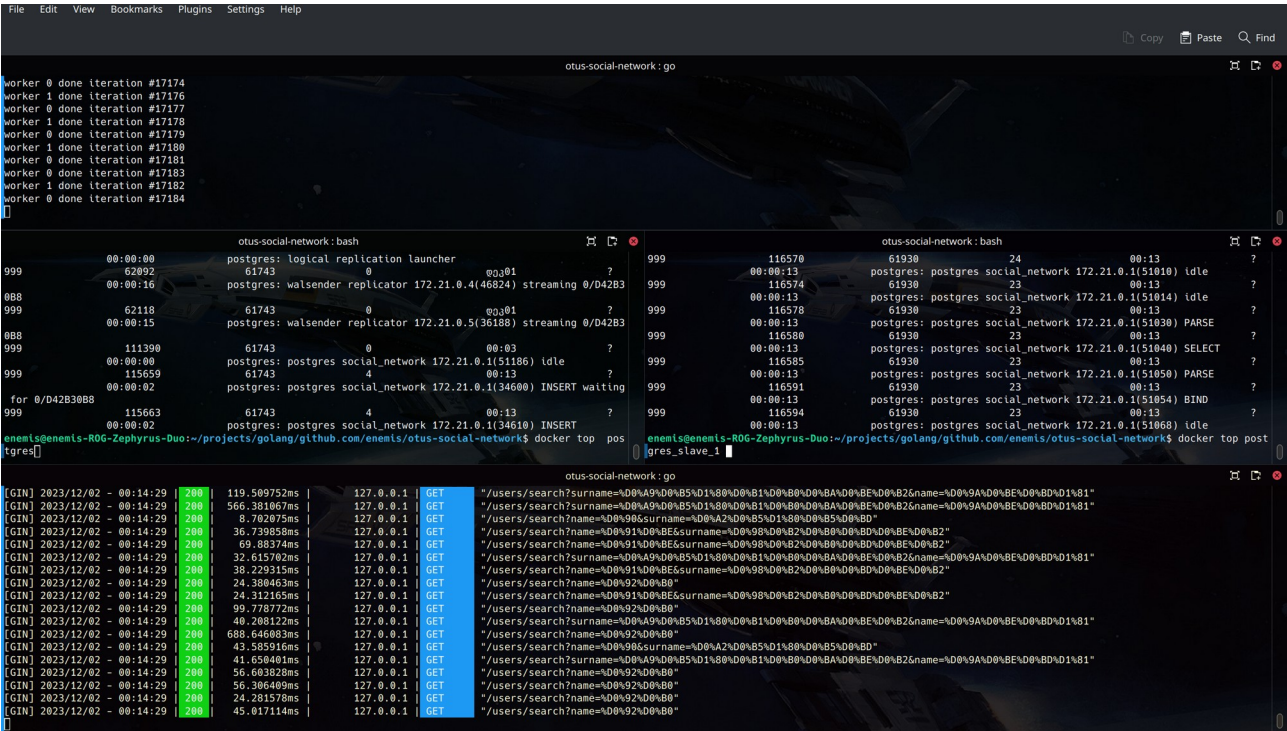
	# Samples	Average	Median	90% Line	95% Line	99% Line	Min	Max	Error %	Throughput	Received KB/sec
Бо Иванов	136145	104	50	199	377	1028	1	3897	0.137%	1862.27037	3811.41
Ва	136125	107	52	204	385	1064	0	3877	0.137%	1861.94586	94527.39
Щербаков Конс	136154	107	51	204	389	1056	0	4412	0.156%	1862.31706	1122.43
А Терен	136141	107	52	205	386	1060	0	3892	0.149%	1862.19018	88192.93



В результате тестирования нагрузка оказалась почти предельная для железа и начали возникать ошибки отказа.

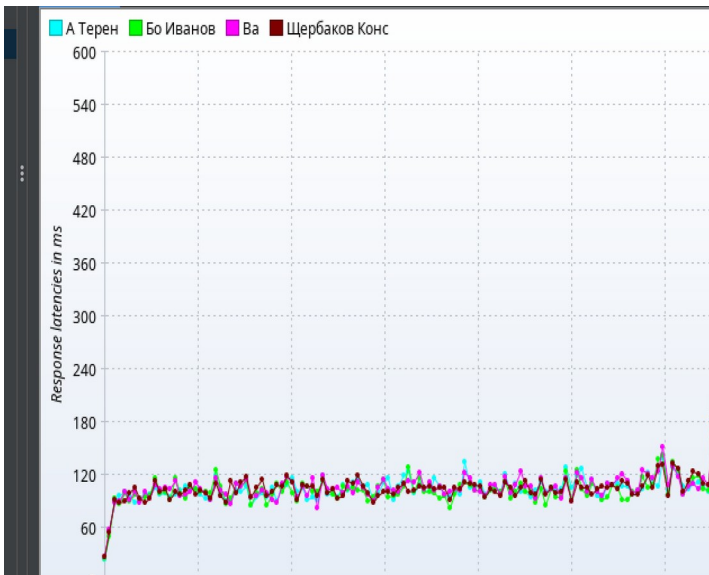
3. Нагрузочный тест после перевода методов на слейв

Для чистоты эксперимента я запустил генератор сидов тоесть операции на вставку и параллельно запустил нагрузочное тестирование метода /search. В результате тестирования нагрузка оказалась почти предельная для железа и начали возникать ошибки отказа. Для нагрузочного тестирования использовалось 800 пользователей.



Процесс тестирования

aggregate_report_statistics_after											
	# Samples	Average	Median	90% Line	95% Line	99% Line	Min	Max	Error %	Throughput	
Ва	160000	109	52	208	400	1053	2	5482	0.000%	1737.18554	
Бо Иванов	160000	106	50	205	396	1050	1	5436	0.000%	1737.33645	
А Терен	160000	108	52	205	397	1049	2	5472	0.000%	1737.39304	
Щербаков Конс	160000	108	52	206	391	1044	3	5408	0.000%	1737.2044	



По результатам тестирования можно сделать следующие выводы:

- Процент ошибок отказа снизится, в моем случае он стал равен 0, так как суммарная нагрузка разделится на между мастером и репликой
- Среднее время ответа незначительно вырастит из-за того что возрастет нагрузка на IO сервера. Так как кол-во открытых соединений удвоится, но снизится частота запросов на каждую из баз.
- Самое главное, чуда не случится так как узкое место базы postgres - это хранилище. И так как скорость его относительно константна то прироста производительности не могло произойти. Поэтому на мой взгляд держать реплику на одном и том же жестком диске особо не имеет смысла.

4. Тестирование падения мастера

- Запустил создание сидов на мастере.

```
otus-social-netwo
worker 1 done iteration #7695
Марамзина Юна (Maramzina.Yuna7695@gmail.com)
worker 0 done iteration #7696
Язов Магистриан (Yazov.Magistrian7696@gmail.com)
worker 1 done iteration #7697
Плавильщикова Аяна (Plavil'shnikova.Ayana7697@gmail.com)
worker 0 done iteration #7698
Ануров Пахомий (Anurov.Paxomij7698@gmail.com)
worker 1 done iteration #7699
Зарубина Елизавета (Zarubina.Elizaveta7699@gmail.com)
worker 0 done iteration #7700
Карамзин Платон (Karamzin.Platon7700@gmail.com)
worker 1 done iteration #7701
Валентинова Валерия (Valentinova.Valeriya7701@gmail.com)
worker 0 done iteration #7702
Сергеев Пров (Sergeev.Prov7702@gmail.com)
worker 1 done iteration #7703
Долинская Ада (Dolinskaya.Ada7703@gmail.com)
worker 0 done iteration #7704
```

- Как только остановил контейнер мастера, реплики начали сыпать сообщения в логи о недоступности мастера.

```
otus-social-network: docker
social_network exited with code 137
postgres | 2023-12-02 20:46:06.301 UTC [27] LOG: checkpoint starting: time
postgres | 2023-12-02 20:46:06.315 UTC [27] LOG: checkpoint complete: wrote 3 buffers (0.0%); 0 WAL file(s) added, 0 removed, 0 recycled; write=0.002 s, sync=0.001 s, total=0.014 s; sync files=2, longest=0.001 s, average=0.001 s; distance=0 KB, estimated=0 KB
postgres_slave.1 | 2023-12-02 20:46:06.602 UTC [27] LOG: restartpoint starting: time
postgres_slave.1 | 2023-12-02 20:46:06.613 UTC [27] LOG: restartpoint complete: wrote 1 buffers (0.0%); 0 WAL file(s) added, 0 removed, 0 recycled; write=0.001 s, sync=0.001 s, total=0.011 s; sync files=0, longest=0.000 s, average=0.000 s; distance=0 KB, estimated=0 KB
postgres_slave.2 | 2023-12-02 20:46:06.707 UTC [27] LOG: recovery restart point at 0/D674BF88
postgres_slave.2 | 2023-12-02 20:46:06.718 UTC [27] LOG: restartpoint starting: time
postgres_slave.2 | 2023-12-02 20:46:06.718 UTC [27] LOG: restartpoint complete: wrote 1 buffers (0.0%); 0 WAL file(s) added, 0 removed, 0 recycled; write=0.001 s, sync=0.001 s, total=0.003 s; sync files=0, longest=0.000 s, average=0.000 s; distance=0 KB, estimated=0 KB
postgres_slave.2 | 2023-12-02 20:53:22.658 UTC [30] FATAL: could not receive data from WAL stream: server closed the connection unexpectedly
postgres_slave.2 | This probably means the server terminated abnormally
postgres_slave.2 | before or while processing the request.
postgres_slave.2 | 2023-12-02 20:53:22.658 UTC [29] LOG: invalid resource manager ID 55 at 0/DAB59E20
postgres_slave.2 | 2023-12-02 20:53:22.658 UTC [30] FATAL: could not receive data from WAL stream: server closed the connection unexpectedly
postgres_slave.2 | This probably means the server terminated abnormally
postgres_slave.2 | before or while processing the request.
postgres_slave.2 | 2023-12-02 20:53:22.658 UTC [29] LOG: invalid resource manager ID 55 at 0/DAB59E20
postgres_slave.2 | 2023-12-02 20:53:22.653 UTC [30] FATAL: could not connect to the primary server: connection to server at "postgres" (172.21.0.2), port 5432 failed: Connection refused
postgres_slave.2 | Is the server running on that host and accepting TCP/IP connections?
postgres_slave.2 | 2023-12-02 20:53:22.653 UTC [29] LOG: waiting for WAL to become available at 0/DAB59E38
postgres_slave.2 | 2023-12-02 20:53:22.654 UTC [35] FATAL: could not connect to the primary server: connection to server at "postgres" (172.21.0.2), port 5432 failed: Connection refused
postgres_slave.2 | Is the server running on that host and accepting TCP/IP connections?
postgres_slave.2 | 2023-12-02 20:53:22.654 UTC [29] LOG: waiting for WAL to become available at 0/DAB59E38
postgres_slave.2 | 2023-12-02 20:53:27.654 UTC [36] FATAL: could not connect to the primary server: could not translate host name "postgres" to address: Temporary failure in name resolution
postgres_slave.2 | 2023-12-02 20:53:27.654 UTC [29] LOG: waiting for WAL to become available at 0/DAB59E38
postgres_slave.1 | 2023-12-02 20:53:27.654 UTC [39] FATAL: could not connect to the primary server: could not translate host name "postgres" to address: Temporary failure in name resolution
postgres_slave.1 | 2023-12-02 20:53:27.654 UTC [29] LOG: waiting for WAL to become available at 0/DAB59E38
postgres_slave.2 | 2023-12-02 20:53:32.658 UTC [37] FATAL: could not connect to the primary server: could not translate host name "postgres" to address: Temporary failure in name resolution
postgres_slave.2 | 2023-12-02 20:53:32.658 UTC [29] LOG: waiting for WAL to become available at 0/DAB59E38
postgres_slave.1 | 2023-12-02 20:53:32.658 UTC [40] FATAL: could not connect to the primary server: could not translate host name "postgres" to address: Temporary failure in name resolution
postgres_slave.1 | 2023-12-02 20:53:32.658 UTC [29] LOG: waiting for WAL to become available at 0/DAB59E38
```

- Транзакции которые были незакомичены были утеряны

```
Лукьянова Айсель (Luk'yanova.Aysel'7713@gmail.com)
worker 0 done iteration #7714
Сухов Иуда (Sukhov.Iuda7714@gmail.com)
worker 1 done iteration #7715
Задорнова Ляля (Zadornova.Lyalya7715@gmail.com)
worker 0 done iteration #7716
Чумаков Тарас (Chumakov.Taras7716@gmail.com)
read tcp 127.0.0.1:57516->127.0.0.1:5432: read: connection reset by peer
worker 0 done iteration #7718
Морковин Мирон (Morkovin.Miron7718@gmail.com)
read tcp 127.0.0.1:57520->127.0.0.1:5432: read: connection reset by peer
worker 0 done iteration #7719
Протасова Сафия (Protasova.Safiya7719@gmail.com)
read tcp 127.0.0.1:57522->127.0.0.1:5432: read: connection reset by peer
```

- Запросы поступающие позже были выполнены сервером с ошибкой о недоступности базы

```

Мамантова Зоряна (Mamantova.Zoryana7853@gmail.com)
read tcp 127.0.0.1:58678->127.0.0.1:5432: read: connection reset by peer
worker 1 done iteration #7852
Бунин Ждан (Bunin.Zhdan7852@gmail.com)
dial tcp 127.0.0.1:5432: connect: connection refused
worker 0 done iteration #7854
Веллер Венедикт (Veller.Venedikt7854@gmail.com)
dial tcp 127.0.0.1:5432: connect: connection refused
worker 1 done iteration #7855
Чипчикова Луиза (Chipchikova.Luiza7855@gmail.com)
dial tcp 127.0.0.1:5432: connect: connection refused

```

- Последняя запись на реплике была успешно выполненая запись в мастер

Language: English
PostgreSQL » postgres_slave_1 » social_network » public » Select: users

Adminer 4.8.1
DB: social_network
Schema: public
SQL command
Import
Export
Create table
select goose_db_version
select test
select users

Select: users
Select data
Show structure
Alter table
New item

Select
Search
Sort
Limit
50
Text length
100
Action
Select

SELECT * FROM "users" LIMIT 50 OFFSET 1487150 (0.338 s) Edit

	id	name	surname	email	birthday	biography	city
<input type="checkbox"/> Modify	d4ada386-9154-11ee-b3e9-0242ac150002	Тарас	Чумаков	Chumakov.Taras7716@gmail.com	2017-01-01	Accusamus deleniti vel tenetur impedit ab.	Louisville

Page
1 ... 29740 29741 29742 29743 29744
Whole result
☐ 1,487,151 rows
Modify
Selected (0)

Export (1,487,151)

Import