

ЛР 3. НА Postgres Cluster

Опираясь на гайд, мы подняли кластер Postgres

```
C:\Users\User1\Desktop\lab_1>docker compose up --build -d
[+] Building 2.05 (16/18)
--> [pg-master] internal load build definition from Dockerfile
--> >> transferring dockerfile: 479B
--> [pg-slave internal] load build definition from Dockerfile
--> >> transferring dockerfile: 479B
--> [pg-master internal] load metadata for docker.io/library/postgres:15
--> [pg-slave internal] load .dockerignore
--> >> transferring context: 28
--> [pg-master internal] load .dockerignore
--> >> transferring context: 28
--> [pg-slave internal] FROM docker.io/library/postgres:15@sha256:24d6c206bb8c0440bcbe24a8d4bf642f60bf7aea94887051ea5761d29c22323
--> [pg-slave internal] load build context
--> >> transferring context: 688
--> [pg-master internal] load build context
--> >> transferring context: 688
--> CACHED [pg-master 2/5] RUN apt-get update -y && apt-get install -y netcat-openbsd python3-pip curl python3-psycopg2 python3-venv iputils-ping
--> CACHED [pg-master 3/5] COPY patroni-venv /opt/patroni-venv && /opt/patroni-venv/bin/pip install --upgrade pip && /opt/patroni-venv/bin/pip install patroni[zookeeper] psycopg2-binary
--> CACHED [pg-slave 5/5] COPY postgres1.yml /postgres1.yml
--> [pg-master] exporting to image
--> >> exporting layers
--> > exporting manifest sha256:84d39b5c38b9ded1e80cd293d5e8d401d8a9ded6c1e24c80a4a763b4fbba7
--> > exporting config sha256:1e5f652073729fc00cf2ed58cc0ff2e1abbb8c32c7e29e77cad24bd4d466073df
--> > exporting attestation manifest sha256:6928d908eb989dac019fffa2373a5a24df5402f2b029fe06c7b45cb4598208dc06
--> > exporting manifest list sha256:3dea06e2a2a241fbfb6d7dabb1eb44ff39aae6c1e661b5f39b8d495eb948f
--> > naming to docker.io/library/lab_1_pg-master:latest
--> > unpacking to docker.io/library/lab_1_pg-master:latest
--> [pg-slave] exporting to image
--> > exporting layers
--> > exporting manifest sha256:a75cfaf589ac03f7a00782a514fe82a7a02b3e529a90c96f8d6f1905ee0aeac4d
--> > exporting config sha256:608beb6bc92b2a6f66c71f1b1bede51f3df03d047791e8c4150c074c90f92
--> > exporting attestation manifest sha256:e229fd1ea4aae4e5da2db4bc25c8296c39aacf4e593b16550e35342195c609a
--> > exporting manifest list sha256:608beb6bc92b2a6f66c71f1b1bede51f3df03d047791e8c4150c074c90f92
--> > naming to docker.io/library/lab_1_pg-slave:latest
--> > unpacking to docker.io/library/lab_1_pg-slave:latest
--> [pg-master] resolving provenance for metadata file
--> [pg-slave] resolving provenance for metadata file

Container zoo          Started
Container pg-slave      Started
Container pg-master     Started

C:\Users\User1\Desktop\lab_1>docker compose logs -f
```

Ответы на вопросы:

При обычном перезапуске композ-проекта, будет ли сбилден заново образ?

Нет

А если предварительно отредактировать файлы postgresX.yml?

Нет

А если содержимое самого Dockerfile?

Нет

Почему?

Пока явно не попросить build или up --build, докер считает, что существующий образ годится

Проверяем репликацию:

Dashboard Properties SQL Statistics Dependencies Dependents postgres/postgres@db2*

postgres/postgres@db2

No limit

Query History

```
1 CREATE TABLE my_first_replication (
2     id int,
3     my_data varchar,
4     my_comment varchar
5 );
6
7 INSERT INTO my_first_replication VALUES('1',
8                                         'my important data',
9                                         'is this line replicated?')
10
```

Data output Messages Notifications

INSERT 0 1

Query returned successfully in 56 msec.

Dashboard Properties SQL Statistics Dependencies Dependents postgres/post... postgres/postgres@db1*

postgres/postgres@db1

No limit

Query History

```
1 INSERT INTO my_first_replication VALUES('2',
2                                         'bla bla bla',
3                                         'work?')
```

Data output Messages Notifications

ERROR: cannot execute INSERT in a read-only transaction
SQL state: 25006

The screenshot shows the pgAdmin 4 interface with a database tree on the left. The tree structure is as follows:

- > 1..3 Sequences
- > Tables (1)
 - > my_first_replication
- > Trigger Functions
- > Types
- > Views
- > Subscriptions
- > Login/Group Roles
- > Tablespaces

A node labeled db2 is expanded, showing:

- > Databases (1)
 - > postgres
 - > Casts
 - > Catalogs
 - > Event Triggers
 - > Extensions
 - > Foreign Data Wrappers
 - > Languages
 - > Publications
 - > Schemas (1)
 - > public
 - > Aggregates
 - > Collations
 - > Domains
 - > FTS Configurations
 - > FTS Dictionaries
 - > FTS Parsers
 - > FTS Templates
 - > Foreign Tables
 - > Functions
 - > Materialized Views
 - > Operators
 - > Procedures
 - > 1..3 Sequences
 - > Tables (1)
 - > my_first_replication
 - >

В pgAdmin мы подключились к нодам постгреса, в pg-master создали таблицу и записали туда данные, на последнем (На данный момент) скриншоте видно, что таблица появилась сразу в обоих нодах.

На третьем скрине видно, что при попытке записать данные в pg-slave, возникает ошибка

Высокая доступность:

Следуя гайду, добавили HAProxy и перезапустили проект

```
=> [pg-master] exporting to image
=> => exporting layers
=> => exporting manifest sha256:84d39b5c38b9deed13e80cd3293d5e8d401d8a9ded6c1e24c80a4a763b4fbaa7
=> => exporting config sha256:1e5f652073729fc00cfef2d58c69f2e14b8bbc32c7e29e77cad24b4db466073df
=> => exporting attestation manifest sha256:700df709b29e731f17c04578162819b9708c2192f76a658cdd789990e41ac00d
=> => exporting manifest list sha256:db37225db9302f36deffc0e6ee976fd05959d8c4f3b09a250d43b4d25c353c03
=> => naming to docker.io/library/lab_1-pg-master:latest
=> => unpacking to docker.io/library/lab_1-pg-master:latest
=> [pg-slave] exporting to image
=> => exporting layers
=> => exporting manifest sha256:a75cfa589ac03f7a00782a514fe82a7a92b3e529a90c96f8d6f1905ee0aeac4d
=> => exporting config sha256:60ebbeb8c92b2a6f66c71fa1810e9e51f3df03d04f7f93e98c4150c074c90f982
=> => exporting attestation manifest sha256:15f9da51e4c977b4c4abbb3ed464bba7256266928b8df95863efe373a15d86c
=> => exporting manifest list sha256:f18ef13a64ead4e262196e3af9852255354dbdcf2aace63aca64f6ea97f3cf0e
=> => naming to docker.io/library/lab_1-pg-slave:latest
=> => unpacking to docker.io/library/lab_1-pg-slave:latest
=> [pg-slave] resolving provenance for metadata file
=> [pg-master] resolving provenance for metadata file
+| Running w/5
  █ Network lab_1_default      Created
  █ Container pg-master        Started
  █ Container zoo               Started
  █ Container pg-slave         Started
  █ Container postgres_entrypoint Started
C:\Users\User1\Desktop\lab_1>
```

После отключения мастер ноды:

```
C:\Users\User1\Desktop\lab_1>docker stop pg-slave
pg-slave
C:\Users\User1\Desktop\lab_1>docker logs postgres_entrypoint
[NOTICE] (1) : New worker (8) forked
[NOTICE] (1) : Loading success.
[WARNING] (8) : Server postgres/pg_master is DOWN, reason: Layer4 connection problem, info: "Connection refused", check duration: 0ms. 1 active and 0 backup servers left. 0 sessions active, 0 queued, 0 remaining in queue.
[WARNING] (8) : Server postgres/pg_slave is DOWN, reason: layer4 connection problem, info: "Connection refused", check duration: 0ms. 0 active and 0 backup servers left. 0 sessions active, 0 queued, 0 remaining in queue.
[ALERT] (8) : proxy /postgres/pg_master is UP
[WARNING] (8) : proxy /postgres/pg_slave is UP, reason: Layer7 check passed, code: 200, check duration: 1ms. 1 active and 0 backup servers online. 0 sessions requested, 0 total in queue.
[WARNING] (8) : Server postgres/pg_master is UP, reason: Layer7 check passed, code: 200, check duration: 1ms. 2 active and 0 backup servers online. 0 sessions requested, 0 total in queue.
[WARNING] (8) : Server postgres/pg_slave is DOWN, reason: layer4 timeout, check duration: 3002ms. 1 active and 0 backup servers left. 0 sessions active, 0 queued, 0 remaining in queue.

C:\Users\User1\Desktop\lab_1>docker logs
"docker logs" requires exactly 1 argument.
See 'docker logs -help'.

Usage: docker logs [OPTIONS] CONTAINER
Fetch the logs of a container
```

```

C:\Users\User1\Desktop\lab_1>docker logs pg-master
2025-12-07 23:28:29,123 INFO: Connecting to zoo(172.18.0.4):2181, use_ssl: False
2025-12-07 23:28:29,123 WARNING: Connection dropped: socket connection error: Connection refused
2025-12-07 23:28:29,253 INFO: Connecting to zoo(172.18.0.4):2181, use_ssl: False
2025-12-07 23:28:29,254 WARNING: Connection dropped: socket connection error: Connection refused
2025-12-07 23:28:29,442 INFO: Connecting to zoo(172.18.0.4):2181, use_ssl: False
2025-12-07 23:28:29,442 WARNING: Connection dropped: socket connection error: Connection refused
2025-12-07 23:28:29,888 INFO: Connecting to zoo(172.18.0.4):2181, use_ssl: False
2025-12-07 23:28:29,888 WARNING: Connection dropped: socket connection error: Connection refused
2025-12-07 23:28:31,106 INFO: Connecting to zoo(172.18.0.4):2181, use_ssl: False
2025-12-07 23:28:31,106 WARNING: Connection dropped: socket connection error: Connection refused
2025-12-07 23:28:32,029 INFO: Connecting to zoo(172.18.0.4):2181, use_ssl: False
2025-12-07 23:28:32,029 WARNING: Connection dropped: socket connection error: Connection refused
2025-12-07 23:28:33,330 INFO: Connecting to zoo(172.18.0.4):2181, use_ssl: False
2025-12-07 23:28:33,352 INFO: Zookeeper connection established, state: CONNECTED
2025-12-07 23:28:33,364 INFO: No PostgreSQL configuration items changed, nothing to reload.
2025-12-07 23:28:33,368 INFO: Systemd integration is not supported
2025-12-07 23:28:33,407 WARNING: Postgresql is not running.
2025-12-07 23:28:33,407 INFO: Lock owner: None; I am postgresql0
2025-12-07 23:28:33,409 INFO: pg_controldata:
    pg_control version number: 1300
    Catalog version number: 202209061
    Database system identifier: 7581248061605298199
    Database cluster state: shut down in recovery
    pg_control last modified: Sun Dec 7 23:28:13 2025
    Latest checkpoint location: 0/80000108
    Latest checkpoint's REDO location: 0/800000D0
    Latest checkpoint's REDO WAL file: 00000006000000000000000000000008
    Latest checkpoint's TimeLineID: 6
    Latest checkpoint's PrevTimeLineID: 6
    Latest checkpoint's full_page_writes: on
    Latest checkpoint's NextXID: 0:744
    Latest checkpoint's NextOID: 16394
    Latest checkpoint's NextMultiXactId: 1
    Latest checkpoint's NextMultiOffset: 0
    Latest checkpoint's oldestXID: 717
    Latest checkpoint's oldestXID's DB: 1
    Latest checkpoint's oldestActiveXID: 744
    Latest checkpoint's oldestMultiXid: 1
    Latest checkpoint's oldestMulti's DB: 1
    Latest checkpoint's oldestCommitTsXid: 0
    Latest checkpoint's newestCommitTsXid: 0
    Time of latest checkpoint: Sun Dec 7 23:26:13 2025
    Fake LSN counter for unlogged rels: 0/3E8
    Minimum recovery ending location: 0/9000000
    Min recovery ending loc's timeline: 6
    Backup start location: 0/0
    Backup end location: 0/0
    End-of-backup record required: no
    wal_level setting: replica
    wal_log_hints setting: on
    max_connections setting: 100

max_worker_processes setting: 8
max_wal_senders setting: 10
max_prepared_xacts setting: 0
max_locks_per_xact setting: 64
max_parallel_workers setting: off
Maximum data alignment: 8
Database block size: 8192
Blocks per segment for large relation: 131072
Maximal column length: 16384
Bytes per WAL segment: 16772716
Maximum length of identifiers: 64
Maximal columns in an index: 32
Maximal columns in a table: 1996
Size of a large-object chunk: 2048
Date/time type storage: 64-bit Integers
Float8 argument passing: by value
Double precision argument passing: by value
Mock authentication nonce: e9b0efbcfc57eb7912d5e15b2c25fae57c7deeb9277e3475257134c6096a503e

2025-12-07 23:28:33,436 INFO: Lock owner: None; I am postgresql0
2025-12-07 23:28:33,435 UTC [24] LOG:  starting as a secondary
2025-12-07 23:28:33,756 UTC [24] LOG:  starting PostgreSQL 15.15 (Debian 15.15-1+pgdg13+1) on x86_64-pc-linux-gnu, compiled by gcc (Debian 14.2.0-19) 14.2.0, 64-bit
2025-12-07 23:28:33,756 UTC [24] LOG:  listening on IPv4 address "0.0.0.0", port 5432
2025-12-07 23:28:33,756 UTC [24] LOG:  listening on Unix socket "/var/run/postgresql/.s.PGSQL.5432"
2025-12-07 23:28:33,774 UTC [28] LOG:  database system was shut down in recovery at 2025-12-07 23:28:13 UTC
2025-12-07 23:28:33,774 UTC [28] WARNING:  The database server will regularly poll the pg_wal subdirectory to check for files placed there.
2025-12-07 23:28:33,774 UTC [28] HINT:  The database is starting standby.
2025-12-07 23:28:33,774 UTC [28] FATAL:  the database system is starting up
localhost:5432: rejecting connection
localhost:5432: rejecting connection
localhost:5432: rejecting connection
2025-12-07 23:28:33,785 UTC [28] LOG:  redo starts at 0/8000000
2025-12-07 23:28:33,785 UTC [28] LOG:  invalid magic number 00000 in log segment 00000000000000000000000000000000, offset 0
2025-12-07 23:28:33,785 UTC [28] LOG:  content of recovery state checked at 0/8000000
2025-12-07 23:28:33,785 UTC [28] LOG:  invalid magic number 00000000000000000000000000000000, offset 0
2025-12-07 23:28:33,785 UTC [28] LOG:  waiting for WAL to become available at 0/9000000
2025-12-07 23:28:33,785 UTC [24] LOG:  database system is ready to accept read-only connections
localhost:5432: rejecting connection
localhost:5432: rejecting connection
localhost:5432: rejecting connection
2025-12-07 23:28:34,075 UTC [28] LOG:  establishing a new patroni heartbeat connection to postgres
2025-12-07 23:28:34,075 UTC [28] LOG:  establishing a new patroni restapi connection to postgres
2025-12-07 23:28:34,075 UTC [28] LOG:  establishing response from postgresql http://pg-slave:8000/patroni: {"state": "running", "postmaster_start_time": "2025-12-07 23:28:33.770004+00:00", "role": "replica", "server_version": 150015, "xlog": {"received_xlog": "150995104", "replayed_xlog": "150995104", "replayed_lsn": null, "paused": false}, "timeline": 6, "cluster_unlocked": true, "ds_last_tseen": 1765150114, "database_system_identifier": "7581248061605298199", "patroni": {"version": "v0.12.0", "commit": "150995104", "cluster": "1765150114", "postmaster_start_time": "2025-12-07 23:28:33.770004+00:00"}, "wal_lsn": 0}
2025-12-07 23:28:34,075 UTC [28] LOG:  Wal position of postgresql1 is ahead of my wal position
2025-12-07 23:28:34,076 INFO: local timeline=6 lsn=0/9000000
2025-12-07 23:28:34,075 UTC [24] LOG:  received SIGABP, reloading configuration files
2025-12-07 23:28:34,075 UTC [24] LOG:  parameter "primary_conninfo" changed to "dbname=postgres user=replicator passfile=/tmp/pgpass0 host=pg-slave port=5432 sslmode=prefer application_name=postgresql0 gssencmode=prefer channel_binding=prefer"
2025-12-07 23:28:34,075 UTC [24] LOG:  parameter "primary_slot_name" changed to "postgresql0"
2025-12-07 23:28:34,087 UTC [24] LOG:  following a different leader because i am not the healthiest node
2025-12-07 23:28:34,087 UTC [46] LOG:  started streaming WAL from primary at 0/9000000 on timeline 6
2025-12-07 23:28:34,078 UTC [46] LOG:  replication terminated by primary server
2025-12-07 23:28:34,078 UTC [46] LOG:  DETAIL:  End of WAL reached on timeline 6 at 0/9000000
2025-12-07 23:28:34,079 UTC [46] LOG:  failed to find timeline history file for timeline 7 from primary server
2025-12-07 23:28:34,093 UTC [28] LOG:  new target timeline is 7
2025-12-07 23:28:34,093 UTC [28] LOG:  invalid record length at 0/9000000: wanted 24, got 0
2025-12-07 23:28:34,093 UTC [28] LOG:  invalid record length at 0/9000000 on timeline 7
2025-12-07 23:28:44,015 INFO: lock owner: postgresql1; I am postgresql0
2025-12-07 23:28:44,019 INFO: local timeline=7 lsn=0/9000108
2025-12-07 23:28:44,039 INFO: primary_timeline=
```

После падения pg-slave Patroni автоматически выполнил failover: pg-master был промoted в нового лидера и начал обслуживать запросы как мастер

После остановки pg-slave HAProxy начинает фиксировать таймауты к нему, она перестаёт использовать упавший pg-slave и оставляет в пуле только pg-master (уже как нового мастера). Поэтому все дальнейшие клиентские запросы через entrypoint идут на живой сервер

Мы попытались сделать запись в бд через entrypoint подключение:

The screenshot shows the pgAdmin 4 interface. The top navigation bar includes links for Dashboard, Properties, SQL, Statistics, Dependencies, Dependents, and a connection named 'postgres/postgres@proxy*'. Below the navigation is a toolbar with various icons for database management. The main area is divided into sections: 'Query' (underlined) and 'Query History'. A single-line query is entered: '1 INSERT INTO failover_test VALUES (2);'. The results section is titled 'Messages' and displays the output: 'INSERT 0 1' and 'Query returned successfully in 6 secs 305 msec.'. At the bottom, status bars show 'Total rows: 1 of 1' and 'Query complete 00:00:06.305'.

У нас получилось, но HAProxy ещё какое-то время пытался опрашивать мёртвый pg-slave и ждал таймаут перед тем, как переключиться на живой сервер