OSM Tile Server Ubuntu 24.04

Введение

Установка производиться на устройство

- **Материнская плата Supermicro MBD-H11SSL-I**
- **Процессор AMD EPYC™ 7551P OEM**
- **Raid5 массив 1tb SSD M.2 Samsung 970 EVO Plus**
- **256GB DDR4 3200MHz DIMM ECC Reg Micron**
- **ОС Ubuntu Server 24.04 LTS**

Все ниже написанное относиться к Ubuntu Server 24.04 (Noble Numbat) и протестированы в августе 2024 года.

Общая теория

Tile Server OSM представляет собой набор программ и библиотек, которые работают вместе для создания тайлового (плиточного) сервера. Он состоит из 5 основных компонентов:

- **mod_tile**
- **renderd**
- **mapnik**
- **osm2pgsql**
- **postgresql/postgis**

Postgresql/postgis - базы данных. Mod_tile — это модуль apache, который обслуживает кэшированные тайлы и решает, какие тайлы нуждаются в повторном рендеринге — либо потому, что они еще не кэшированы, либо потому, что они устарели. Mapnik — это программная библиотека, которая осуществляет реальный рендеринг с использованием открытого рендеринга.
Предполагается, что вы запускаете все от пользователя без регистрации root через «sudo». Имя пользователя без root, повторяемое по умолчанию, — «alisa». Если вы указываете иного пользователя, то вам необходимо будет заменить во всех командах на свое имя и добавить его в группу пользователей, которые могут использовать sudo для получения root-прав. Из вашей обычной учетной записи пользователя без регистрации root.

Обновление программного обеспечения

Устанавливаем и обновляем Ubuntu

```
login as: alisa
alisa@192.168.1.64's password:
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-39-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Fri Aug  2 07:27:30 PM UTC 2024

  System load:  0.0               Processes:             168
  Usage of /:   3.5% of 121.44GB  Users logged in:       1
  Memory usage: 1%                IPv4 address for eth0: 192.168.1.64
  Swap usage:   0%


Expanded Security Maintenance for Applications is not enabled.

40 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status
```

Всегда полезно обновлять программное обеспечение сервера перед выполнением какой-либо серьезной работы на сервере. Войдите на сервер через SSH и выполните следующую команду.

```
sudo apt --yes update; sudo apt --yes upgrade -y
```

```
alisa@smmap:~$ sudo apt install libboost-all-dev git tar unzip wget bzip2 build-essential autoconf libtool libxml2-de
v libgeos-dev libgeos++-dev libpq-dev libbz2-dev libproj-dev munin-node munin protobuf-c-compiler libfreetype6-dev li
btiff5-dev libicu-dev libgdal-dev libcairo2-dev libcairomm-1.0-dev apache2 apache2-dev libagg-dev liblua5.2-dev fonts
-unifont lua5.1 liblua5.1-0-dev
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
bzip2 is already the newest version (1.0.8-5build1).
bzip2 set to manually installed.
```

Установка необходимых библиотек

Итак, приступим:

```
sudo apt --yes install screen locate git tar unzip wget bzip2 lua5.1 gdal-
bin npm node-carto net-tools curl
```

```
alisa@smmap:~$ sudo apt --yes install screen locate git tar unzip wget bzip2 lua5.1 gdal-bin npm node-carto net-tools
 curl
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
screen is already the newest version (4.9.1-1build1).
screen set to manually installed.
git is already the newest version (1:2.43.0-1ubuntu7.1).
git set to manually installed.
tar is already the newest version (1.35+dfsg-3build1).
tar set to manually installed.
wget is already the newest version (1.21.4-1ubuntu4.1).
wget set to manually installed.
curl is already the newest version (8.5.0-2ubuntu10.1).
curl set to manually installed.
```

Процесс достаточно продолжительный и можно попить кофейку...
Установка необязытельных, но рекомендованных openstreetmap-carto элементов

```
sudo apt --yes install ca-certificates gnupg build-essential clang nodejs
fonts-hanazono fonts-noto-cjk fonts-noto-hinted fonts-noto-unhinted
```

```
alisa@smmap:~$ sudo apt --yes install ca-certificates gnupg build-essential clang nodejs fonts-hanazono fonts-noto-cj
k fonts-noto-hinted fonts-noto-unhinted
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ca-certificates is already the newest version (20240203).
ca-certificates set to manually installed.
gnupg is already the newest version (2.4.4-2ubuntu17).
gnupg set to manually installed.
build-essential is already the newest version (12.10ubuntu1).
build-essential set to manually installed.
nodejs is already the newest version (18.19.1+dfsg-6ubuntu5).
nodejs set to manually installed.
```

Установка postgresql/postgis

В Ubuntu есть предварительно упакованные версии как postgis, так и postgresql, поэтому их можно просто установить через диспетчер пакетов Ubuntu. Проверим подходящую версию на странице пакета postgresql в Ubuntu

Пакет postgresql-16-postgis-3

- noble (24.04LTS) (разное): Поддержка географических объектов для PostgreSQL 16 [**universe**]
  3.4.2+dfsg-1ubuntu3: amd64 arm64 armhf ppc64el riscv64 s390x
- oracular (разное): Поддержка географических объектов для PostgreSQL 16 [**universe**]
  3.4.2+dfsg-2: amd64 arm64 armhf ppc64el riscv64 s390x

Другие совпадения

Пакет postgresql-16-postgis-3-scripts

- noble (24.04LTS) (разное): Поддержка географических объектов для PostgreSQL 16 -- SQL-скрипты [**universe**]
  3.4.2+dfsg-1ubuntu3: все
- oracular (разное): Поддержка географических объектов для PostgreSQL 16 -- SQL-скрипты [**universe**]
  3.4.2+dfsg-2: все

```
sudo apt --yes install postgresql postgresql-contrib postgis postgresql-16-postgis-3 postgresql-16-postgis-3-scripts
```

```
alisa@smmap:~$ sudo apt --yes install postgresql postgresql-contrib postgis postgresql-16-postgis-3 postgresql-16-pos
tgis-3-scripts
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
```

Здесь «postgresql» — это база данных, в которой мы собираемся хранить данные карты, а «postgis» добавляет к ней дополнительную графическую поддержку. У команды apt параметр –yes отвечает за автоматические подстановки подтверждения ДА при установке. Проверим список установленных пакетов postgres

```
dpkg -l | grep postgres
```

```
alisa@smmap:~$ dpkg -l | grep postgres
ii  postgresql                                    16+257build1                          all          object-re
lational SQL database (supported version)
ii  postgresql-16                                 16.3-0ubuntu0.24.04.1                 amd64        The World
's Most Advanced Open Source Relational Database
ii  postgresql-16-postgis-3                       3.4.2+dfsg-1ubuntu3                   amd64        Geographi
c objects support for PostgreSQL 16
ii  postgresql-16-postgis-3-scripts               3.4.2+dfsg-1ubuntu3                   all          Geographi
c objects support for PostgreSQL 16 -- SQL scripts
ii  postgresql-client-16                          16.3-0ubuntu0.24.04.1                 amd64        front-end
 programs for PostgreSQL 16
ii  postgresql-client-common                      257build1                             all          manager f
or multiple PostgreSQL client versions
ii  postgresql-common                             257build1                             all          PostgreSQ
L database-cluster manager
ii  postgresql-contrib                            16+257build1                          all          additiona
l facilities for PostgreSQL (supported version)
ii  postgresql-postgis                            3.4.2+dfsg-1ubuntu3                   amd64        Geographi
c objects support for PostgreSQL -- Metapackage
ii  postgresql-postgis-scripts                    3.4.2+dfsg-1ubuntu3                   all          Geographi
c objects support for PostgreSQL -- SQL scripts metapackage
alisa@smmap:~$
```

Примечание! Для полного удаления воспользуйтесь командой

```
sudo apt-get --purge --yes remove postgresql postgresql-doc postgresql-common postgresql-16  postgresql-client  postgresql-client-16 postgresql-client-common postgresql-common  postgresql-contrib postgresql-contrib-16 postgresql-postgis-scripts postgresql-16-postgis-3-scripts
```

Перезапускаем postgresql

```
sudo /etc/init.d/postgresql restart
```

```
alisa@smmap:~$ sudo /etc/init.d/postgresql restart
Restarting postgresql (via systemctl): postgresql.service.
alisa@smmap:~$
```

Проверим статус работы postgresql

```
sudo /etc/init.d/postgresql status
```

```
alisa@smmap:~$ sudo /etc/init.d/postgresql status
● postgresql.service - PostgreSQL RDBMS
     Loaded: loaded (/usr/lib/systemd/system/postgresql.service; enabled; preset: enabled)
     Active: active (exited) since Sun 2024-08-04 13:36:12 UTC; 44s ago
    Process: 18719 ExecStart=/bin/true (code=exited, status=0/SUCCESS)
   Main PID: 18719 (code=exited, status=0/SUCCESS)
        CPU: 2ms

Aug 04 13:36:12 smmap systemd[1]: Starting postgresql.service - PostgreSQL RDBMS...
Aug 04 13:36:12 smmap systemd[1]: Finished postgresql.service - PostgreSQL RDBMS.
alisa@smmap:~$
```

Создаем базу данных postgis. По умолчанию в различных программах предполагается, что база данных называется gis, и тут мы будем использовать то же соглашение, хотя это и не обязательно. Замените свое имя пользователя вместо alisa, где это используется ниже. Это должно быть имя пользователя, которое будет отображать карты с помощью Mapnik.

```
sudo -u postgres -i
```

```
postgres@smmap:~$ createuser alisa
postgres@smmap:~$
```

```
createuser alisa
```

```
postgres@smmap:~$ createuser alisa
postgres@smmap:~$
```

если ошибки нет, то продолжаем установку

Если Ошибка

```
postgres@smmap:~$ createuser alisa
createuser: error: connection to server on socket "/var/run/postgresql/.s.PGSQL.5432" failed: No such file or directo
ry
        Is the server running locally and accepting connections on that socket?
```

перезапускаем postgresql

```
/etc/init.d/postgresql restart
```

```
postgres@smmap:~$ /etc/init.d/postgresql restart
Restarting postgresql (via systemctl): postgresql.service==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-unit
s ===
Authentication is required to restart 'postgresql.service'.
Authenticating as: alisa
Password:
==== AUTHENTICATION COMPLETE ===
.
```

добавляем пользователя

```
createuser alisa
```

```
postgres@smmap:~$ createuser alisa
postgres@smmap:~$
```

проверяем

```
pg_isready
```

```
postgres@smmap:~$ createuser alisa
postgres@smmap:~$ pg_isready
/var/run/postgresql:5432 - accepting connections
postgres@smmap:~$
```

ответ «/var/run/postgresql:5432 - accepting connections»

Продолжаем установку

```
createdb -E UTF8 -O alisa gis
```

```
postgres@smmap:~$ createdb -E UTF8 -O alisa gis
postgres@smmap:~$
```

Продолжая работать как пользователь «postgres», настройте PostGIS в б PostgreSQL (опять же, меняя alisa на свое имя пользователя ниже):

```
psql
```

```
postgres@smmap:~$ psql
psql (16.3 (Ubuntu 16.3-0ubuntu0.24.04.1))
Type "help" for help.

postgres=#
```

(ответ «postgres=#»)

```
\c gis
```

```
postgres=# \c gis
You are now connected to database "gis" as user "postgres".
gis=#
```

(ответ: “You are now connected to database ‘gis’ as user ‘postgres’”.)

```
CREATE EXTENSION postgis;
CREATE EXTENSION hstore;
ALTER TABLE geometry_columns OWNER TO alisa;
ALTER TABLE spatial_ref_sys OWNER TO alisa;
```

```
gis=# CREATE EXTENSION postgis;
CREATE EXTENSION hstore;
ALTER TABLE geometry_columns OWNER TO alisa;
ALTER TABLE spatial_ref_sys OWNER TO alisa;
CREATE EXTENSION
CREATE EXTENSION
ALTER TABLE
ALTER TABLE
gis=#
```

ответ:
CREATE EXTENSION
CREATE EXTENSION
ALTER TABLE
ALTER TABLE

```
\q
```

```
gis=# \q
postgres@smmap:~$
```

(выход из psql и приглашение к Linux)

```
exit
```

```
postgres@smmap:~$ exit
logout
alisa@smmap:~$
```

(это вернет нас к пользователю, которым мы были до того, как сделали «sudo -u postgres -i» выше)
Если вы еще не создали пользователя, создайте пользователя Unix и для этого пользователя, выбрав пароль при появлении запроса:

```
sudo useradd -m alisa
```

```
alisa@smmap:~$ sudo useradd -m alisa
useradd: user 'alisa' already exists
alisa@smmap:~$
```

```
sudo passwd alisa
```

```
alisa@smmap:~$ sudo passwd alisa
New password:
Retype new password:
passwd: password updated successfully
alisa@smmap:~$
```

Опять же, выше замените “alisa” на имя пользователя без регистрации root, которого вы выбрали.

Установка osm2pgsql

Далее мы установим osm2pgsql:

```
sudo apt --yes install osm2pgsql
```

```
alisa@smmap:~$ sudo apt --yes install osm2pgsql
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
```

Ошибок обычно быть не должно

```
No containers need to be restarted.

User sessions running outdated binaries:
 alisa @ session #1: login[1065]
 alisa @ session #3: sshd[1253]
 alisa @ user manager service: systemd[1193]

No VM guests are running outdated hypervisor (qemu) binaries on this host.
alisa@smmap:~$
```

Mapnik и python3

Далее мы установим Mapnik python3, используя версию по умолчанию в Ubuntu 24.04:

```
sudo apt --yes install mapnik-utils python3-mapnik python3-psycopg2 python3-yaml
```

```
alisa@smmap:~$ sudo apt --yes install mapnik-utils python3-mapnik python3-psycopg2 python3-yaml
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
python3-psycopg2 is already the newest version (2.9.9-1build1).
python3-psycopg2 set to manually installed.
python3-yaml is already the newest version (6.0.1-2build2).
python3-yaml set to manually installed.
```

Проверим правильность установки Mapnik:

```
python3
```

```
alisa@smmap:~$ python3
Python 3.12.3 (main, Jul 31 2024, 17:43:48) [GCC 13.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

```
import mapnik
```

```
>>> import mapnik
>>>
```

(Ответ: »> и без ошибок, то библиотека Mapnik была найдена Python.)

```
quit()
```

```
>>> quit()
alisa@smmap:~$
```

выход

Установка Apache, mod_tile и render

- Далее мы установим Apache, mod_tile и renderd. «mod_tile» — это модуль Apache, который обрабатывает запросы на тайлы; «render» — это демон, который фактически отображает тайлы, когда «mod_tile» их запрашивает. Мы будем использовать ветку «switch2osm»
- https://github.com/SomeoneElseOSM/mod_tile, которая сама является ответвлением
- https://github.com/openstreetmap/mod_tile, но модифицирована для поддержки Ubuntu 20.04 и с пара других изменений для работы на стандартном сервере Ubuntu, а не на одном из серверов рендеринга OSM.

Установка пакетов

В репозитории Ubuntu по умолчанию нет пакетов mod_tileи renderd, но мы можем установить их из OSM PPA.

```
sudo apt install software-properties-common
sudo add-apt-repository ppa:osmadmins/ppa
```

```
alisa@smmap:~$ sudo apt install software-properties-common
sudo add-apt-repository ppa:osmadmins/ppa
[sudo] password for alisa:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
software-properties-common is already the newest version (0.99.48).
software-properties-common set to manually installed.
```

Нажмем [ENTER] для продолжения.

```
More info: https://launchpad.net/~osmadmins/+archive/ubuntu/ppa
Adding repository.
Press [ENTER] to continue or Ctrl-c to cancel.
Hit:1 http://security.ubuntu.com/ubuntu noble-security InRelease
Get:2 https://ppa.launchpadcontent.net/osmadmins/ppa/ubuntu noble InRelease [18.1 kB]
Get:3 https://ppa.launchpadcontent.net/osmadmins/ppa/ubuntu noble/main amd64 Packages [892 B]
Get:4 https://ppa.launchpadcontent.net/osmadmins/ppa/ubuntu noble/main Translation-en [756 B]
Hit:5 http://ru.archive.ubuntu.com/ubuntu noble InRelease
Get:6 http://ru.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Hit:7 http://ru.archive.ubuntu.com/ubuntu noble-backports InRelease
Fetched 146 kB in 1s (110 kB/s)
Reading package lists... Done
alisa@smmap:~$
```

Установим веб-сервер Apache. Конфигурационный файл renderd будет создан в /etc/apache2/conf-available/renderd.conf

```
sudo apt --yes install apache2 apache2-dev  libapache2-mod-tile renderd
```

```
alisa@smmap:~$ sudo apt --yes install apache2 apache2-dev  libapache2-mod-tile renderd
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils autoconf automake autopoint autotools-dev debhelper debugedit
  dh-autoreconf dh-strip-nondeterminism dwz gettext intltool-debian libaprl-dev libaprlt64 libaprutill-dbd-sqlite3
  libaprutill-dev libaprutill-ldap libaprutillt64 libarchive-cpio-perl libarchive-zip-perl libdebhelper-perl
  libexpat1-dev libfile-stripnondeterminism-perl libiniparserl libldap-dev libldap2-dev libltdl-dev liblua5.4-0
  libmail-sendmail-perl libpcre2-16-0 libpcre2-32-0 libpcre2-dev libpcre2-posix3 libsctp-dev libsctpl
  libsub-override-perl libsys-hostname-long-perl libtool m4 po-debconf uuid-dev
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom www-browser autoconf-archive gnu-standards
```

Включим mod-tile модуль.

```
sudo a2enmod tile
```

```
alisa@smmap:~$ sudo a2enmod tile
Module tile already enabled
alisa@smmap:~$
```

если установили из репозитория, то компиляцию пропускаем

Скомпилируем исходный код mod_tile

Создадим в домашнем каталоге папку src

```
mkdir ~/src
```

```
alisa@smmap:~$ mkdir ~/src
alisa@smmap:~$
```

- перейдем в эту папку

```
cd ~/src
```

```
alisa@smmap:~$ cd ~/src
alisa@smmap:~/src$
```

- Обновим установленные пакеты

```
sudo apt update && sudo apt --yes upgrade
```

```
alisa@smmap:~$ sudo apt update && sudo apt --yes upgrade
Hit:1 http://ru.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://ru.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://ru.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu noble-security InRelease
0% [Waiting for headers]
```

- Установим зависимости сборки (libmemcached-dev и librados-dev не являются обязательными)

```
sudo apt --no-install-recommends --yes install \
  apache2 \
  apache2-dev \
  cmake \
  curl \
  g++ \
  gcc \
  git \
```

```
  libcairo2-dev \
  libcurl4-openssl-dev \
  libglib2.0-dev \
  libiniparser-dev \
  libmapnik-dev \
  libmemcached-dev \
  librados-dev
```

```
alisa@smmap:~$ sudo apt --no-install-recommends --yes install \
  apache2 \
  apache2-dev \
  cmake \
  curl \
  g++ \
  gcc \
  git \
  libcairo2-dev \
  libcurl4-openssl-dev \
  libglib2.0-dev \
  libiniparser-dev \
  libmapnik-dev \
  libmemcached-dev \
  librados-dev
```

- Загрузим, создадим, протестируем и установим `mod_tile`

```
export CMAKE_BUILD_PARALLEL_LEVEL=$(nproc)
rm -rf /tmp/mod_tile_src /tmp/mod_tile_build
mkdir /tmp/mod_tile_src /tmp/mod_tile_build
cd /tmp/mod_tile_src
git clone --depth 1 https://github.com/openstreetmap/mod_tile.git .
cd /tmp/mod_tile_build
cmake -B . -S /tmp/mod_tile_src \
  -DCMAKE_BUILD_TYPE:STRING=Release \
  -DCMAKE_INSTALL_LOCALSTATEDIR:PATH=/var \
  -DCMAKE_INSTALL_PREFIX:PATH=/usr \
  -DCMAKE_INSTALL_RUNSTATEDIR:PATH=/run \
  -DCMAKE_INSTALL_SYSCONFDIR:PATH=/etc \
  -DENABLE_TESTS:BOOL=ON
cmake --build .
ctest
sudo cmake --install . --strip
```

```
alisa@smmap:~$ export CMAKE_BUILD_PARALLEL_LEVEL=$(nproc)
rm -rf /tmp/mod_tile_src /tmp/mod_tile_build
mkdir /tmp/mod_tile_src /tmp/mod_tile_build
cd /tmp/mod_tile_src
git clone --depth 1 https://github.com/openstreetmap/mod_tile.git .
cd /tmp/mod_tile_build
cmake -B . -S /tmp/mod_tile_src \
  -DCMAKE_BUILD_TYPE:STRING=Release \
  -DCMAKE_INSTALL_LOCALSTATEDIR:PATH=/var \
  -DCMAKE_INSTALL_PREFIX:PATH=/usr \
  -DCMAKE_INSTALL_RUNSTATEDIR:PATH=/run \
  -DCMAKE_INSTALL_SYSCONFDIR:PATH=/etc \
  -DENABLE_TESTS:BOOL=ON
cmake --build .
ctest
sudo cmake --install . --strip
```

- Должно ответить 100% тестом

```
142/143 Test #142: bad_httpd_config_25 ....................................   Passed    0.06 sec
        Start 143: bad_httpd_config_26
143/143 Test #143: bad_httpd_config_26 ....................................   Passed    0.06 sec

100% tests passed, 0 tests failed out of 143

Total Test time (real) = 137.94 sec
-- Install configuration: "Release"
-- Up-to-date: /var/cache/renderd/tiles
-- Up-to-date: /run/renderd
-- Installing: /etc/apache2/mods-available/tile.load
-- Installing: /etc/renderd.conf
-- Installing: /usr/lib/apache2/modules/mod_tile.so
-- Installing: /usr/bin/render_expired
-- Installing: /usr/bin/render_list
-- Installing: /usr/bin/render_old
-- Installing: /usr/bin/render_speedtest
-- Installing: /usr/bin/renderd
-- Installing: /usr/share/man/man1/render_expired.1
-- Installing: /usr/share/man/man1/render_list.1
-- Installing: /usr/share/man/man1/render_old.1
-- Installing: /usr/share/man/man1/render_speedtest.1
-- Installing: /usr/share/man/man1/renderd.1
-- Installing: /usr/share/man/man5/renderd.conf.5
alisa@smmap:/tmp/mod_tile_build$
```

- Включить конфигурацию модуля пакета a2enmod

```
a2enmod tile
```

```
alisa@smmap:~$ a2enmod tile
Module tile already enabled
alisa@smmap:~$
```

- Запустим сервис

```
sudo apache2ctl start
```

```
alisa@smmap:~$ sudo apache2ctl start
Invoking 'systemctl start apache2'.
Use 'systemctl status apache2' for more info.
alisa@smmap:~$
```

- Перезапустим сервис веб-сервера apache2

```
sudo service apache2 reload
```

```
alisa@smmap:~$ sudo service apache2 reload
alisa@smmap:~$
```

- Проверим статус сервиса веб-сервера apache2

```
sudo systemctl status apache2
```

```
alisa@smmap:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled)
     Active: active (running) since Sun 2024-08-04 17:18:36 UTC; 1h 23min ago
       Docs: https://httpd.apache.org/docs/2.4/
    Process: 72516 ExecReload=/usr/sbin/apachectl graceful (code=exited, status=1/FAILURE)
   Main PID: 23009 (apache2)
      Tasks: 55 (limit: 11839)
     Memory: 9.7M (peak: 15.9M)
        CPU: 1.050s
     CGroup: /system.slice/apache2.service
             ├─23009 /usr/sbin/apache2 -k start
             ├─27608 /usr/sbin/apache2 -k start
             └─27609 /usr/sbin/apache2 -k start
```

Конфигурация таблицы стилей

- Теперь, когда все необходимое программное обеспечение установлено, вам нужно загрузить и настроить таблицу стилей.
- Здесь мы будем использовать тот же стиль, что и на «стандартной» карте на веб-сайте openstreetmap.org.

Создадим, если ранее этого не делали каталог src в домашней директории

```
mkdir ~/src
```

```
alisa@smmap:~$ mkdir ~/src
alisa@smmap:~$
```

Перейдем в каталог src

```
cd ~/src
```

```
alisa@smmap:~$ cd ~/src
alisa@smmap:~/src$
```

клонируем репозитарий

```
git clone https://github.com/gravitystorm/openstreetmap-carto
```

```
alisa@smmap:~/src$ git clone https://github.com/gravitystorm/openstreetmap-carto
Cloning into 'openstreetmap-carto'...
remote: Enumerating objects: 18194, done.
remote: Counting objects: 100% (349/349), done.
remote: Compressing objects: 100% (227/227), done.
remote: Total 18194 (delta 205), reused 230 (delta 118), pack-reused 17845
Receiving objects: 100% (18194/18194), 17.25 MiB | 9.36 MiB/s, done.
Resolving deltas: 100% (12573/12573), done.
alisa@smmap:~/src$
```

переходим в каталог openstreetmap-carto

```
cd openstreetmap-carto
```

```
alisa@smmap:~/src$ cd openstreetmap-carto
alisa@smmap:~/src/openstreetmap-carto$
```

установку подходящей версию компилятора «carto» пропустим, т.к. это сделали на первоначальном этапе загрузки необходимых пакетов.

```
sudo apt install npm
```

```
alisa@smmap:~/src/openstreetmap-carto$ sudo apt install npm
[sudo] password for alisa:
```

размер примерно в 600mb, т.ч. перекур
Скомпилируем «carto»

```
sudo npm install -g carto
```

```
alisa@smmap:~/src/openstreetmap-carto$ sudo apt install npm
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
npm is already the newest version (9.2.0~ds1-2).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
alisa@smmap:~/src/openstreetmap-carto$ sudo npm install -g carto
(                    ) ⠸ idealTree:lib: timing idealTree:#root Completed in 2000ms
```

вывод на экран

```
alisa@smmap:~/src/openstreetmap-carto$ sudo npm install -g carto

added 64 packages in 7s

1 package is looking for funding
  run `npm fund` for details
alisa@smmap:~/src/openstreetmap-carto$
```

Тут разработчики просят профинансировать работу с пакетом openstreetmap-carto

```
npm fund
```

```
alisa@smmap:~/src/openstreetmap-carto$ npm fun
openstreetmap-carto

alisa@smmap:~/src/openstreetmap-carto$
```

Далее применяем заплатку, если видим «npm audit fix –force», иначе пропускаем этот раздел и переходим к проверки версии «carto»

```
npm audit fix --force
```

```
alisa@smmap:~/src/openstreetmap-carto$ npm audit fix --force
npm WARN using --force Recommended protections disabled.
npm ERR! code ENOLOCK
npm ERR! audit This command requires an existing lockfile.
npm ERR! audit Try creating one first with: npm i --package-lock-only
npm ERR! audit Original error: loadVirtual requires existing shrinkwrap file

npm ERR! A complete log of this run can be found in:
npm ERR!     /home/alisa/.npm/_logs/2024-08-01T19_17_39_474Z-debug-0.log
alisa@smmap:~/src/openstreetmap-carto$
```

```
npm i --package-lock-only
```

```
alisa@smmap:~/src/openstreetmap-carto$ npm i --package-lock-only
npm ERR! code ENOENT
npm ERR! syscall open
npm ERR! path /home/alisa/src/openstreetmap-carto/package.json
npm ERR! errno -2
npm ERR! enoent ENOENT: no such file or directory, open '/home/alisa/src/openstreetmap-carto/package.json'
npm ERR! enoent This is related to npm not being able to find a file.
npm ERR! enoent

npm ERR! A complete log of this run can be found in:
npm ERR!     /home/alisa/.npm/_logs/2024-08-01T19_23_34_489Z-debug-0.log
alisa@smmap:~/src/openstreetmap-carto$
```

```
npm audit fix --force
```

```
alisa@smmap:~/src/openstreetmap-carto$ npm audit fix --force
npm WARN using --force Recommended protections disabled.

up to date, audited 1 package in 346ms

found 0 vulnerabilities
alisa@smmap:~/src/openstreetmap-carto$
```

```
npm audit fix
```

```
alisa@smmap:~/src/openstreetmap-carto$ npm audit fix

up to date, audited 1 package in 334ms

found 0 vulnerabilities
alisa@smmap:~/src/openstreetmap-carto$
```

Проверим версию пакета «carto»

```
carto -v
```

```
alisa@smmap:~/src/openstreetmap-carto$ carto -v
```

Это должно ответить номером версии, которая должна быть не меньше: 1.2.0

```
alisa@smmap:~/src/openstreetmap-carto$ carto -v
1.2.0
alisa@smmap:~/src/openstreetmap-carto$
```

преобразуем проект carto во что-то, понятное Mapnik:

```
carto project.mml > mapnik.xml
```

```
alisa@smmap:~/src/openstreetmap-carto$ carto project.mml > mapnik.xml
Warning: style/admin.mss:23:19 Styles do not match layer selector #admin-low-zoom.
Warning: style/admin.mss:22:18 Styles do not match layer selector #admin-low-zoom.
Warning: style/admin.mss:18:6 Styles do not match layer selector #admin-low-zoom.
alisa@smmap:~/src/openstreetmap-carto$
```

Теперь у вас есть таблица стиля Mapnik XML в /home/alisa/src/openstreetmap-carto/mapnik.xml.
Примечание: все предепреждения
Warning: style/admin.mss:22:18 Styles do not match layer selector #admin-low-zoom.
Warning: style/admin.mss:18:6 Styles do not match layer selector #admin-low-zoom.
Игнорируем

/home/alisa/src/openstreetmap-carto/

Имя	Размер	Изменено	Права	Владел...
CARTOGRAPHY.md	5 KB	04.08.2024 18:54:03	rw-rw-r--	alisa
CHANGELOG.md	42 KB	04.08.2024 18:54:03	rw-rw-r--	alisa
CODE_OF_CONDUCT.md	6 KB	04.08.2024 18:54:03	rw-rw-r--	alisa
CONTRIBUTING.md	9 KB	04.08.2024 18:54:03	rw-rw-r--	alisa
DOCKER.md	11 KB	04.08.2024 18:54:03	rw-rw-r--	alisa
docker-compose.yml	1 KB	04.08.2024 18:54:03	rw-rw-r--	alisa
Dockerfile	2 KB	04.08.2024 18:54:03	rw-rw-r--	alisa
Dockerfile.db	1 KB	04.08.2024 18:54:03	rw-rw-r--	alisa
Dockerfile.import	1 KB	04.08.2024 18:54:03	rw-rw-r--	alisa
external-data.yml	4 KB	04.08.2024 18:54:03	rw-rw-r--	alisa
indexes.sql	3 KB	04.08.2024 18:54:03	rw-rw-r--	alisa
indexes.yml	2 KB	04.08.2024 18:54:03	rw-rw-r--	alisa
INSTALL.md	6 KB	04.08.2024 18:54:03	rw-rw-r--	alisa
LICENSE.txt	1 KB	04.08.2024 18:54:03	rw-rw-r--	alisa
mapnik.xml	3 187 KB	04.08.2024 19:17:37	rw-rw-r--	alisa
openstreetmap-carto.lua	13 KB	04.08.2024 18:54:03	rw-rw-r--	alisa
openstreetmap-carto.style	3 KB	04.08.2024 18:54:03	rw-rw-r--	alisa
preview.png	361 KB	04.08.2024 18:54:03	rw-rw-r--	alisa
project.mml	95 KB	04.08.2024 18:54:03	rw-rw-r--	alisa

Загрузка данных

в домашнем каталоге создаем папку data

```
mkdir ~/data
```

```
alisa@smmap:~/src/openstreetmap-carto$ mkdir ~/data
alisa@smmap:~/src/openstreetmap-carto$
```

и переходим в эту папку

```
cd ~/data
```

```
alisa@smmap:~/src/openstreetmap-carto$ cd ~/data
alisa@smmap:~/data$
```

на странице загрузки http://download.geofabrik.de/ выбираем нужную нам карту для и копируем с кнопки ее ссылку, процесс не быстрый, идем гулять пол часика с собачкой...
У кого собачки нет, том может для примера загрузить карту Азербайджана размером 32Mb...

```
wget https://download.geofabrik.de/asia/azerbaijan-latest.osm.pbf
```

```
alisa@smmap:~/data$ wget https://download.geofabrik.de/asia/azerbaijan-latest.osm.pbf
--2024-08-04 16:30:40--  https://download.geofabrik.de/asia/azerbaijan-latest.osm.pbf
Resolving download.geofabrik.de (download.geofabrik.de)... 65.109.50.43, 65.109.48.72, 2a01:4f9:5a:25c3::2, ...
Connecting to download.geofabrik.de (download.geofabrik.de)|65.109.50.43|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 39383181 (38M) [application/octet-stream]
Saving to: 'azerbaijan-latest.osm.pbf'

azerbaijan-latest.osm.pbf     100%[=========================================>]  37.56M  3.99MB/s    in 6.1s

2024-08-04 16:30:46 (6.15 MB/s) - 'azerbaijan-latest.osm.pbf' saved [39383181/39383181]

alisa@smmap:~/data$
```

Следующая команда вставит загруженные ранее данные OpenStreetMap в базу данных. Этот шаг очень интенсивно использует дисковый ввод-вывод; импорт всей планеты может занять много часов, дней или недель в зависимости от аппаратного обеспечения.

```
cd ~/data
osm2pgsql -d gis --create --slim  -G --hstore --tag-transform-script
~/src/openstreetmap-carto/openstreetmap-carto.lua -C 4000 --number-processes
2 -S ~/src/openstreetmap-carto/openstreetmap-carto.style ~/data/azerbaijan-
latest.osm.pbf
```

```
alisa@smmap:~/data$ cd ~/data
osm2pgsql -d gis --create --slim  -G --hstore --tag-transform-script ~/src/openstreetmap-carto/openstreetmap-carto.lu
a -C 4000 --number-processes 2 -S ~/src/openstreetmap-carto/openstreetmap-carto.style ~/data/azerbaijan-latest.osm.pb
f
2024-08-04 16:34:29  osm2pgsql version 1.11.0
2024-08-04 16:34:29  Database version: 16.3 (Ubuntu 16.3-0ubuntu0.24.04.1)
2024-08-04 16:34:29  PostGIS version: 3.4
2024-08-04 16:34:29  Storing properties to table '"public"."osm2pgsql_properties"'.
2024-08-04 16:34:30  Setting up table 'planet_osm_point'
2024-08-04 16:34:30  Setting up table 'planet_osm_line'
2024-08-04 16:34:30  Setting up table 'planet_osm_polygon'
2024-08-04 16:34:30  Setting up table 'planet_osm_roads'
```

Обратите внимание на параметр -c (-create) при импорте через osm2pgsql. С этим параметром удалится вся существующая информация с таблиц.

```
osm2pgsql -d gis --append --slim  -G --hstore --tag-transform-script
~/src/openstreetmap-carto/openstreetmap-carto.lua -C 4000 --number-processes
2 -S ~/src/openstreetmap-carto/openstreetmap-carto.style ~/data/russia-
latest.osm.pbf
```

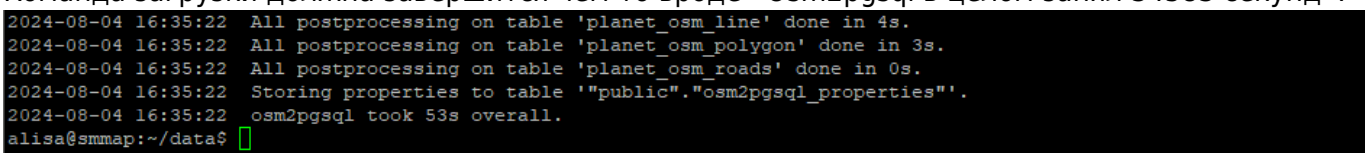
Если вам необходимо сделать импорт ещё нескольких стран, то вместо -c указывайте параметр -a (-append).

```
alisa@smmap:~/data$ osm2pgsql -d gis --append --slim  -G --hstore --tag-transform-script ~/src/openstreetmap-carto/op
enstreetmap-carto.lua -C 4000 --number-processes 2 -S ~/src/openstreetmap-carto/openstreetmap-carto.style ~/data/russ
ia-latest.osm.pbf
```

Командные опции:

- -d gis
- База данных, с которой нужно работать (раньше gis была по умолчанию, теперь нужно указать).
- –create
- Загружает данные в пустую базу данных, а не пытается добавить их в существующую (иначе –append).
- –slim
- osm2pgsql может использовать разные макеты таблиц; «тонкие» таблицы работают для рендеринга.
- -G
- Определяет, как обрабатываются мультиполигоны.
- –hstore
- Позволяет тегам, для которых нет явных столбцов базы данных, использоваться для рендеринга.
- –tag-transform-script
- Определяет сценарий lua, используемый для обработки тегов. Это простой способ обработки тегов OSM до того, как их обработает сам стиль, что значительно упрощает логику стиля.
- -C 4000
- Выделено 4,0 Гб памяти osm2pgsql для процесса импорта. Если у вас меньше памяти, вы можете попробовать меньшее число, а если процесс импорта завершится из-за нехватки памяти, вам нужно будет попробовать меньшее число или меньший размер карты OSM.
- –number-processes 2
- Используйте 2 ЦП. Если у вас есть больше доступных ядер, вы можете использовать больше.
- -S
- Создайте столбцы базы данных в этом файле (на самом деле они не изменились по сравнению с «openstreetmap-carto»)
- ~/src/openstreetmap-carto/openstreetmap-carto.style ~/data/russia-latest.osm.pbf
- Место сохранения и файл данных для загрузки.

Команда загрузки должна завершится чем-то вроде «Osm2pgsql в целом занял 34583 секунд».



Создание индексов

- Начиная с версии v5.3.0 некоторые дополнительные индексы теперь необходимо применять вручную .

переходим в папку openstreetmap-carto

```
cd ~/src/openstreetmap-carto/
```

```
alisa@smmap:~/data$ cd ~/src/openstreetmap-carto/
alisa@smmap:~/src/openstreetmap-carto$
```

- запускаем команду создание индексов

```
psql -d gis -f indexes.sql
```

```
alisa@smmap:~/src/openstreetmap-carto$ psql -d gis -f indexes.sql
```

- (ответ: «CREATE INDEX» 15 раз.)

```
alisa@smmap:~/src/openstreetmap-carto$ psql -d gis -f indexes.sql
CREATE INDEX
CREATE INDEX
CREATE INDEX
CREATE INDEX
CREATE INDEX
CREATE INDEX
CREATE INDEX
CREATE INDEX
CREATE INDEX
CREATE INDEX
CREATE INDEX
CREATE INDEX
CREATE INDEX
CREATE INDEX
CREATE INDEX
alisa@smmap:~/src/openstreetmap-carto$
```

- заняло примерно несколько минут

Загрузка Shape-файла

- Shape-файл — векторный формат для хранения объектов, описываемых геометрией и сопутствующими атрибутами.
- Несмотря на то, что большая часть данных для создания карты, берется из данных OpenStreetMap, загружается не все, некоторые шейп-файлы для таких вещей, как границы стран с малым масштабированием, по-прежнему необходимы. Чтобы загрузить и проиндексировать их:
- переходим в папку openstreetmap-carto

```
cd ~/src/openstreetmap-carto/
```

```
alisa@smmap:~$ cd ~/src/openstreetmap-carto/
alisa@smmap:~/src/openstreetmap-carto$
```

```
scripts/get-external-data.py
```

```
alisa@smmap:~/src/openstreetmap-carto$ scripts/get-external-data.py
INFO:root:Starting load of external data into database
INFO:root:Checking table simplified_water_polygons
```

- Этот процесс включает в себя загрузку и требует некоторого времени — во время работы на экране мало что освещает. На самом деле он загружает таблица водных полигонов и прочее и заполняет каталог «data» ниже «openstreetmap-carto».

```
INFO:root:  Download complete (23866505 bytes)
INFO:root:  Decompressing file
INFO:root:  Importing into database
INFO:root:  Import complete
INFO:root:Checking table water_polygons
INFO:root:  Download complete (864102730 bytes)
INFO:root:  Decompressing file
INFO:root:  Importing into database
INFO:root:  Import complete
INFO:root:Checking table icesheet_polygons
INFO:root:  Download complete (52420603 bytes)
INFO:root:  Decompressing file
INFO:root:  Importing into database
INFO:root:  Import complete
INFO:root:Checking table icesheet_outlines
INFO:root:  Download complete (53083620 bytes)
INFO:root:  Decompressing file
INFO:root:  Importing into database
INFO:root:  Import complete
INFO:root:Checking table ne_110m_admin_0_boundary_lines_land
INFO:root:  Download complete (57325 bytes)
INFO:root:  Decompressing file
INFO:root:  Importing into database
INFO:root:  Import complete
alisa@smmap:~/src/openstreetmap-carto$
```

Шрифты

переходим в каталог openstreetmap-carto в папке src

```
cd ~/src/openstreetmap-carto/
```

```
alisa@smmap:~$ cd ~/src/openstreetmap-carto/
alisa@smmap:~/src/openstreetmap-carto$
```

и собственно загружаем шрифты

```
scripts/get-fonts.sh
```

```
alisa@smmap:~/src/openstreetmap-carto$ cd
alisa@smmap:~$ cd ~/src/openstreetmap-carto/
alisa@smmap:~/src/openstreetmap-carto$ scripts/get-fonts.sh
Warning: Failed to get filetime: No such file or directory
Warning: Illegal date format for -z, --time-cond (and not a file name).
Warning: Disabling time condition. See curl_getdate(3) for valid date syntax.
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0
100  555k  100  555k    0     0   398k      0  0:00:01  0:00:01 --:--:-- 1041k
Warning: Failed to get filetime: No such file or directory
Warning: Illegal date format for -z, --time-cond (and not a file name).
Warning: Disabling time condition. See curl_getdate(3) for valid date syntax.
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0
```

процесс занял 6 минут

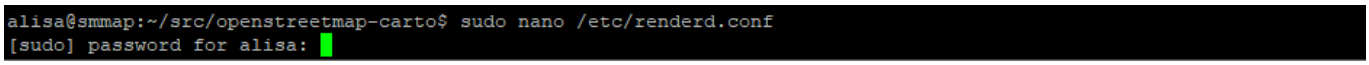
```
100 21858    0 21858    0     0  92165      0 --:--:-- --:--:-- --:--:-- 92618
[/tmp/get-fonts.YGfaQJL3w/Noto_Emoji.zip]
  End-of-central-directory signature not found.  Either this file is not
  a zipfile, or it constitutes one disk of a multi-part archive.  In the
  latter case the central directory and zipfile comment will be found on
  the last disk(s) of this archive.
unzip:  cannot find zipfile directory in one of /tmp/get-fonts.YGfaQJL3w/Noto_Emoji.zip or
        /tmp/get-fonts.YGfaQJL3w/Noto_Emoji.zip.zip, and cannot find /tmp/get-fonts.YGfaQJL3w/Noto_Emoji.zip.ZIP, per
iod.
alisa@smmap:~/src/openstreetmap-carto$
```

Настройка веб-сервера

визуализация

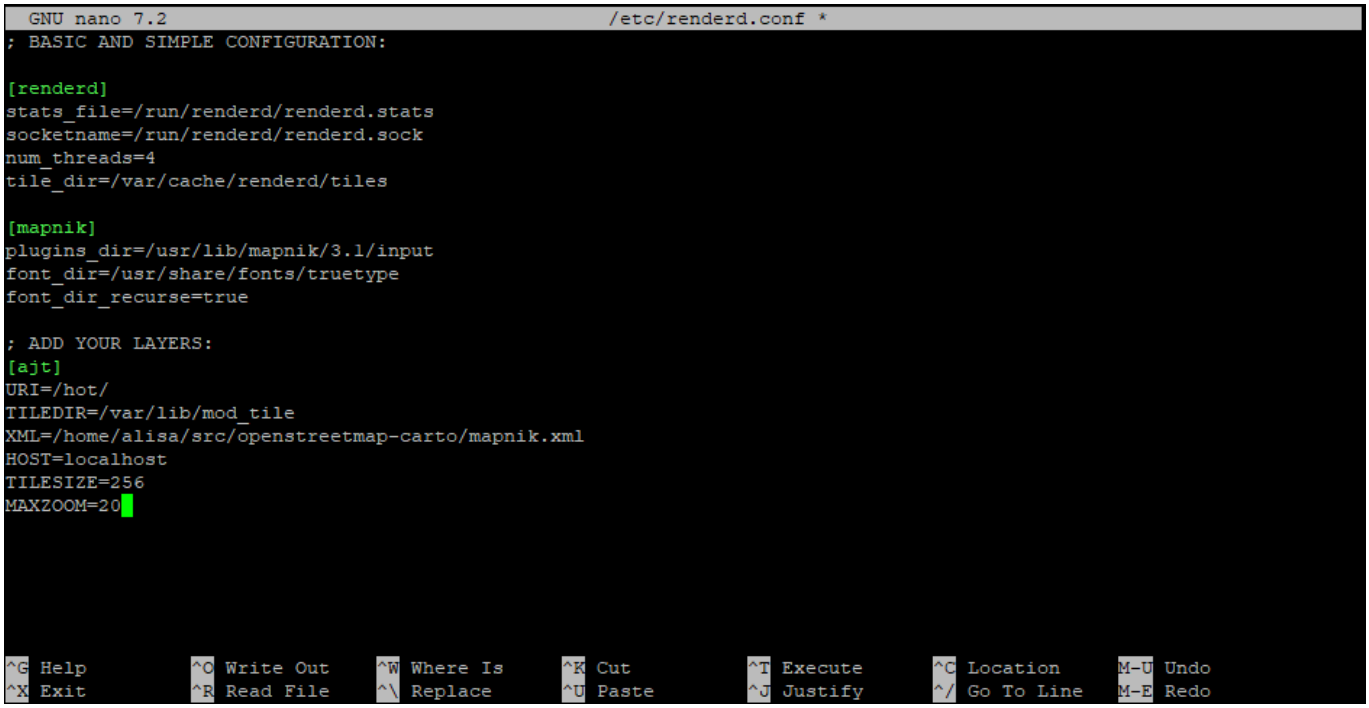
Отрываем с помощью текстового редактора nano файл конфигурации для «render» — «sudo nano /etc/renderd.conf»

```
sudo nano /etc/renderd.conf
```



Добавляем в конец файла нижеприведенные строки

```
[ajt]
URI=/hot/
TILEDIR=/var/lib/mod_tile
XML=/home/alisa/src/openstreetmap-carto/mapnik.xml
HOST=localhost
TILESIZE=256
MAXZOOM=20
```



- нажимаем последовательно **CTRL+0** , **ENTER** для сохранения и **CTRL+X** для выхода из редактора

Настройка Apache

создаем папку mod_tile

```
sudo mkdir /var/lib/mod_tile
```

```
alisa@smmap:~$ sudo mkdir /var/lib/mod_tile
alisa@smmap:~$
```

задаем права пользователя для этой папки

```
sudo chown alisa /var/lib/mod_tile
```

```
alisa@smmap:~$ sudo chown alisa /var/lib/mod_tile
alisa@smmap:~$
```

создаем папку renderd, если не создавали ранее

```
sudo mkdir /var/run/renderd
```

```
alisa@smmap:~$ sudo mkdir /var/run/renderd
```

задаем права пользователя для этой папки

```
sudo chown alisa /var/run/renderd
```

```
alisa@smmap:~$ sudo chown alisa /var/run/renderd
alisa@smmap:~$
```

сообщаем Apache о «mod_tile» редактируя файл mod_tile.conf добавляя следующую строку LoadModule tile_module /usr/lib/apache2/modules/mod_tile.so

```
sudo nano /etc/apache2/conf-available/mod_tile.conf
```

```
alisa@smmap:~$ sudo nano /etc/apache2/conf-available/mod_tile.conf
```

- вставляем строку **LoadModule tile_module /usr/lib/apache2/modules/mod_tile.so** в тело файла (в нашем случае это пустой файл)
- нажимаем последовательно **CTRL+0** , **ENTER** для сохранения и **CTRL+X** для выхода из редактора

```
  GNU nano 7.2                        /etc/apache2/conf-available/mod_tile.conf *
LoadModule tile_module /usr/lib/apache2/modules/mod_tile.so
^G Help          ^O Write Out     ^W Where Is      ^K Cut           ^T Execute       ^C Location      M-U Undo
^X Exit          ^R Read File     ^\ Replace       ^U Paste         ^J Justify       ^/ Go To Line    M-E Redo
```

выполняем команду для включения конфигурации mod_tile.

```
sudo a2enconf mod_tile
```

```
alisa@smmap:~$ sudo a2enconf mod_tile
Enabling conf mod_tile.
To activate the new configuration, you need to run:
  systemctl reload apache2
alisa@smmap:~$
```

- выполняем сообщение о необходимости перезапуска Apache «systemctl reload apache2», чтобы активировать новую конфигурацию;

```
sudo systemctl reload apache2
```

```
alisa@smmap:~$ sudo systemctl reload apache2
alisa@smmap:~$
```

- сообщаем Apache о «renderd»
- открываем в редакторе файл 000-default.conf

```
sudo nano /etc/apache2/sites-available/000-default.conf
```

```
alisa@smmap:~$ sudo nano /etc/apache2/sites-available/000-default.conf
```

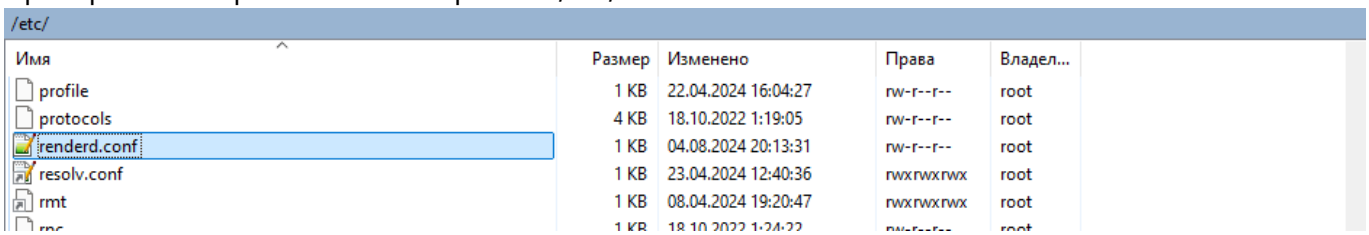
- вставляем следующин строки в тело файла между строками «ServerAdmin» и «DocumentRoot»

```
LoadTileConfigFile /etc/renderd.conf
ModTileRenderdSocketName /var/run/renderd/renderd.sock
# Timeout before giving up for a tile to be rendered
ModTileRequestTimeout 0
# Timeout before giving up for a tile to be rendered that is otherwise
missing
ModTileMissingRequestTimeout 30
```

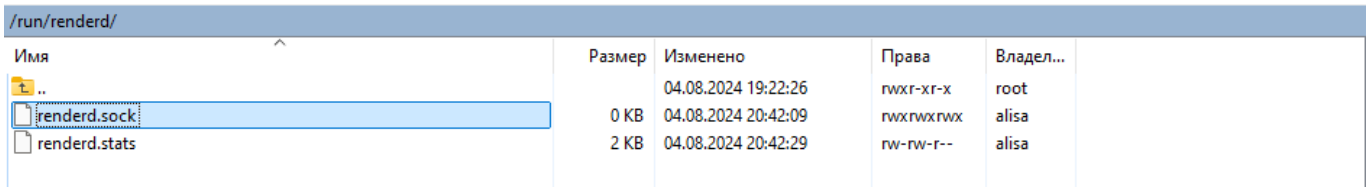
- нажимаем последовательно **CTRL+0** , **ENTER** для сохранения и **CTRL+X** для выхода из редактора

```
  GNU nano 7.2                    /etc/apache2/sites-available/000-default.conf *
<VirtualHost *:80>
        # The ServerName directive sets the request scheme, hostname and port that
        # the server uses to identify itself. This is used when creating
        # redirection URLs. In the context of virtual hosts, the ServerName
        # specifies what hostname must appear in the request's Host: header to
        # match this virtual host. For the default virtual host (this file) this
        # value is not decisive as it is used as a last resort host regardless.
        # However, you must set it for any further virtual host explicitly.
        #ServerName www.example.com

        ServerAdmin webmaster@localhost
LoadTileConfigFile /etc/renderd.conf
ModTileRenderdSocketName /var/run/renderd/renderd.sock
# Timeout before giving up for a tile to be rendered
ModTileRequestTimeout 0
# Timeout before giving up for a tile to be rendered that is otherwise missing
ModTileMissingRequestTimeout 30
        DocumentRoot /var/www/html

        # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
        # error, crit, alert, emerg.
        # It is also possible to configure the loglevel for particular
        # modules, e.g.
        #LogLevel info ssl:warn

        ErrorLog ${APACHE_LOG_DIR}/error.log

^G Help          ^O Write Out     ^W Where Is      ^K Cut           ^T Execute       ^C Location      M-U Undo
^X Exit          ^R Read File     ^\ Replace       ^U Paste         ^J Justify       ^/ Go To Line    M-E Redo
```

Проверим месторасположение файлов /etc/renderd.conf



и /var/run/renderd/renderd.sock



- запускаем и перезагрузите apache:

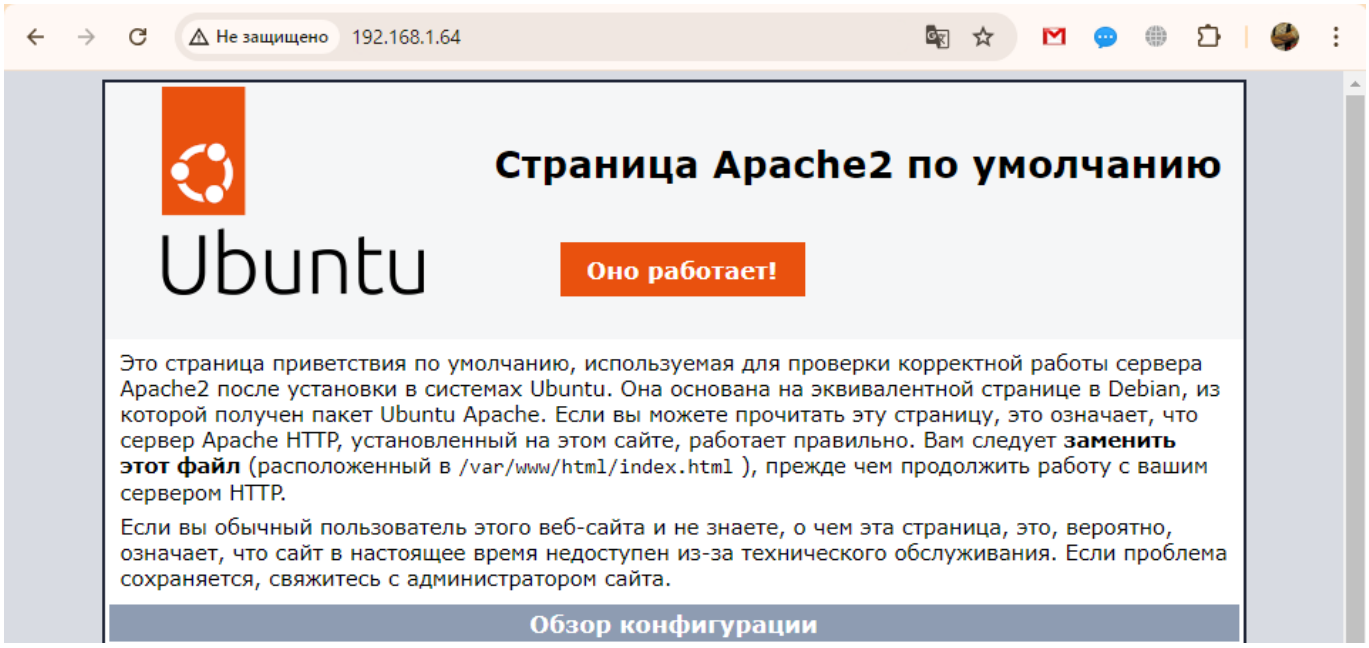
```
sudo service apache2 start
```

```
alisa@smmap:~$ sudo service apache2 start
alisa@smmap:~$
```

```
sudo service apache2 reload
```

```
alisa@smmap:~$ sudo service apache2 reload
alisa@smmap:~$
```

- проверим наш Apache набрав в браузере адрес нашего виртуального сервера (в нашем случае: http://192.168.1.64/)



Перечень команд работы с сервисом рендер

- sudo systemctl start renderd
- sudo systemctl restart renderd
- sudo systemctl enable renderd
- sudo systemctl status renderd
- sudo systemctl stop renderd
- sudo systemctl daemon-reload

Перезапустим и провеим статус сервиса

```
sudo systemctl daemon-reload
sudo systemctl enable renderd
sudo systemctl restart renderd
sudo systemctl status renderd
```

```
alisa@smmap:~$ sudo systemctl daemon-reload
sudo systemctl enable renderd
sudo systemctl restart renderd
sudo systemctl status renderd
Synchronizing state of renderd.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable renderd
● renderd.service - Renderd
     Loaded: loaded (/usr/lib/systemd/system/renderd.service; enabled; preset: enabled)
     Active: active (running) since Sun 2024-08-04 19:53:06 UTC; 25ms ago
    Process: 80821 ExecStart=/etc/init.d/renderd start (code=exited, status=0/SUCCESS)
   Main PID: 80832 (renderd)
      Tasks: 6 (limit: 11839)
     Memory: 9.0M (peak: 11.0M)
        CPU: 106ms
     CGroup: /system.slice/renderd.service
             └─80832 /usr/bin/renderd

Aug 04 19:53:06 smmap renderd[80831]: Initialising unix server socket on /run/renderd/renderd.sock
Aug 04 19:53:06 smmap renderd[80831]: Created server socket 4
Aug 04 19:53:06 smmap renderd[80831]: Renderd is using mapnik version 3.1.0
Aug 04 19:53:06 smmap renderd[80831]: Loading fonts from directory '/usr/share/fonts'
```

«Настройки конфигов»

```
sudo nano /etc/renderd.conf
sudo -u 'alisa' renderd -f -c /etc/renderd.conf
sudo renderd -f -c /etc/renderd.conf
```

```
sudo renderd -f


renderd.service
systemctl start renderd.service
systemctl enable renderd.service
systemctl status renderd.service
systemctl status

sudo systemctl start renderd
sudo systemctl restart renderd
sudo systemctl enable renderd
sudo systemctl status renderd
sudo systemctl stop renderd
sudo systemctl daemon-reload

sudo service renderd start

sudo chown alisa /run/renderd/ -R
sudo chown alisa /var/cache/renderd/tiles/ -R



sudo apt install renderd


sudo chown alisa /home/alisa/src/openstreetmap-carto/ -R
sudo chown alisa /home/alisa/src/openstreetmap-carto/mapnik.xml



sudo nano /etc/renderd.conf
[ajt]
URI=/hot/
TILEDIR=/var/lib/mod_tile
XML=/home/alisa/src/openstreetmap-carto/mapnik.xml
HOST=localhost
TILESIZE=256
MAXZOOM=20

sudo nano /etc/apache2/conf-available/mod_tile.conf
LoadModule tile_module /usr/lib/apache2/modules/mod_tile.so

sudo nano /etc/apache2/sites-available/000-default.conf
LoadTileConfigFile /etc/renderd.conf
ModTileRenderdSocketName /var/run/renderd/renderd.sock
# Timeout before giving up for a tile to be rendered
ModTileRequestTimeout 0
# Timeout before giving up for a tile to be rendered that is otherwise
missing
```

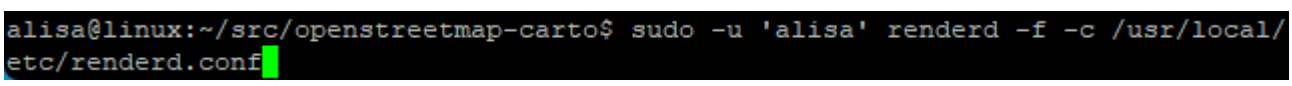
```
ModTileMissingRequestTimeout 30

sudo nano ~/src/mod_tile/debian/renderd.init

sudo nano /usr/lib/systemd/system/renderd.service

[Unit]
Description=Renderd
After=network.target auditd.service

[Service]
Type=forking
ExecStart=/etc/init.d/renderd start
ExecReload=/etc/init.d/renderd restart
ExecStop=/etc/init.d/renderd stop
Environment=G_MESSAGES_DEBUG=all

[Install]
WantedBy=multi-user.target



[Unit]
Description=Daemon that renders map tiles using mapnik
Documentation=man:renderd
After=network.target auditd.service

[Service]
ExecStart=/usr/bin/renderd -f
User=alisa

[Install]
WantedBy=multi-user.target
```

- Дальше не продолжаем если не дождались загрузки данных в базу и Создание индексов

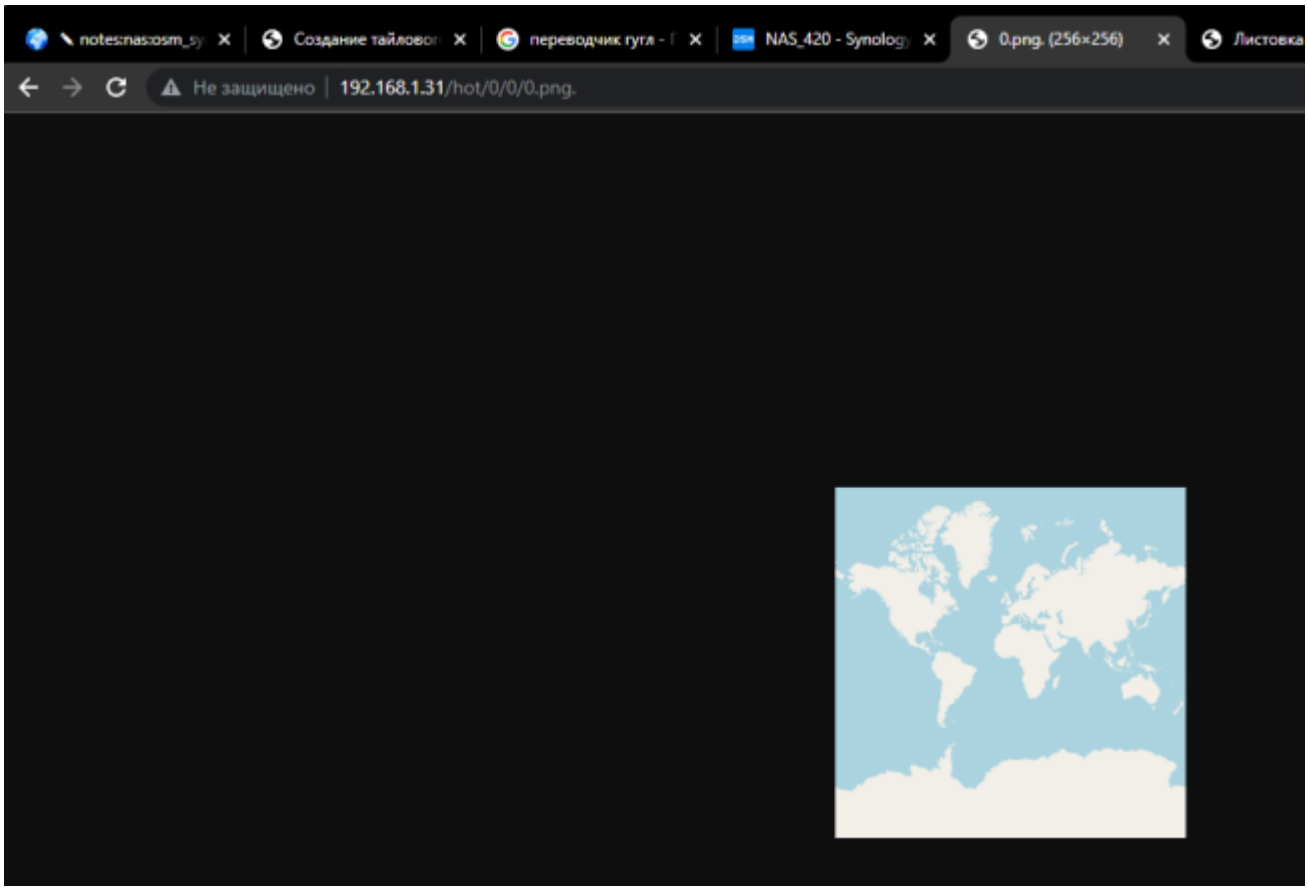
Запуск рендера в первый раз

запустим renderd, чтобы попытаться отрендерить некоторые плитки. Сначала мы запустим его на переднем плане, чтобы видеть любые ошибки по мере их возникновения:

```
sudo -u 'alisa' renderd -f -c /usr/local/etc/renderd.conf
```



- Откройте веб-браузер по адресу: http://yourserveripaddress/hot/0/0/0.png (в нашем случае http://192.168.1.31/hot/0/0/0.png)



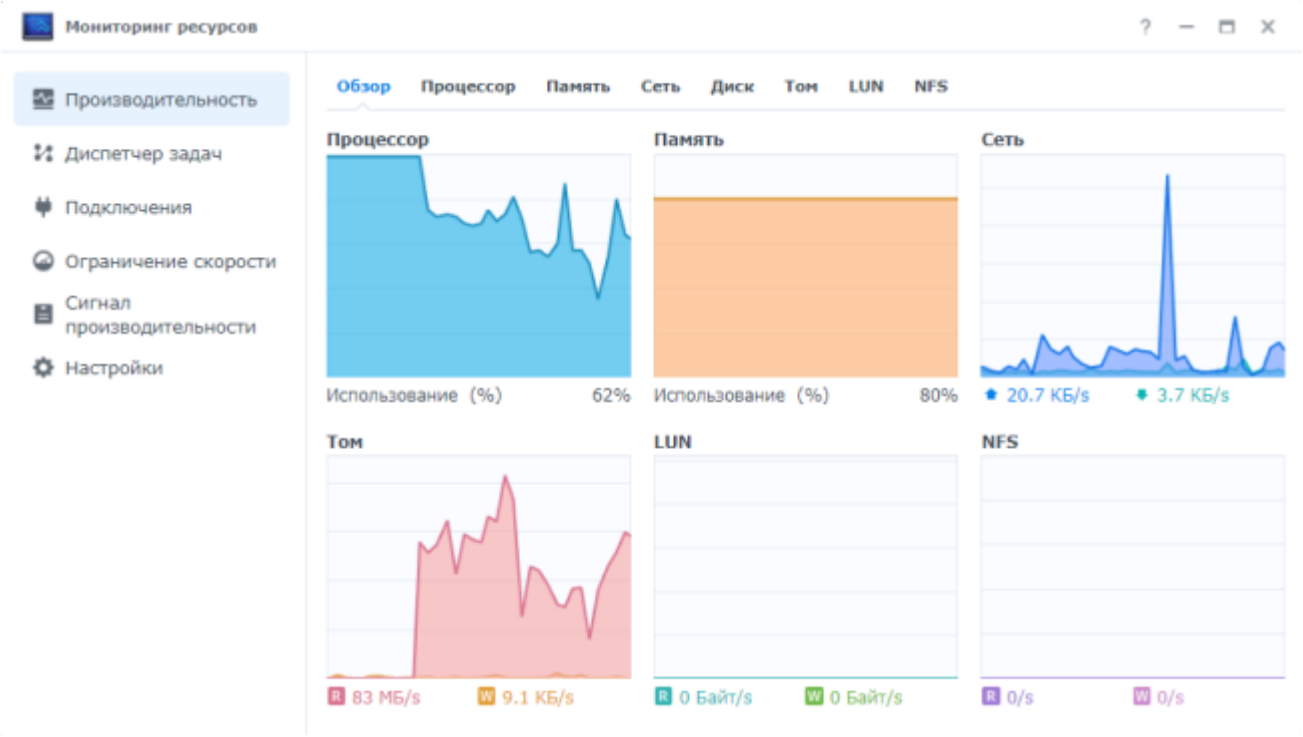
- Если все это работает, нажмите control-c, чтобы завершить процесс рендеринга переднего плана.
- при возникновении необходимости остановить renderd

```
systemctl stop renderd
```

- иногда рендер идет с командой, т.е. без sudo -u 'alisa'

```
renderd -f -c /usr/local/etc/renderd.conf
```

- процессор был недолго в шоке
- на Synology DS720+ такого шока нет и все происходит быстрее в 3-4 раза



- Не знаю почему, но при наличии папки renderd и прав на неё, система ругается на её отсутствие (проверено при 3х установках на Synology Nas. Повторно выполним эти команды...
- создаем папку renderd

```
sudo mkdir /var/run/renderd
```

```
alisa@linux:~/src/openstreetmap-carto$ sudo mkdir /var/run/renderd
```

задаем права пользователя для этой папки

```
sudo chown alisa /var/run/renderd
```

```
alisa@linux:~/src/openstreetmap-carto$ sudo mkdir /var/run/renderd
alisa@linux:~/src/openstreetmap-carto$ sudo chown alisa /var/run/renderd
alisa@linux:~/src/openstreetmap-carto$
```

Запуск рендера в фоновом режиме

Далее мы настроим «рендеринг» для работы в фоновом режиме. Отредактируем файл «~/src/mod_tile/debian/renderd.init», чтобы «RUNASUSER» был установлен для учетной записи без имени root, который мы использовали ранее, например «alisa», и скопируем его в системный каталог. откроем файл в редакторе

```
vi ~/src/mod_tile/debian/renderd.init
```

- нажимаем **i** и меняем в строке RUNASUSER **renderaccount** на ваше имя пользователя (в нашем случае это alisa)
- нажимаем последовательно **esc** , **shift + :** , **w** , **q** , **!** для сохранения и выхода из редактора

```
# Short-Description: Mapnik rendering daemon
# Description:       Mapnik rendering daemon.
### END INIT INFO

# Do NOT "set -e"

# PATH should only include /usr/* if it runs after the mountnfs.sh script
PATH=/sbin:/usr/sbin:/bin:/usr/bin
DESC="Mapnik rendering daemon"
NAME=renderd
DAEMON=/usr/local/bin/$NAME
DAEMON_ARGS="-c /usr/local/etc/renderd.conf"
PIDSOCKDIR=/var/run/$NAME
PIDFILE=$PIDSOCKDIR/$NAME.pid
SCRIPTNAME=/etc/init.d/$NAME
RUNASUSER=alisa

# Exit if the package is not installed
[ -x "$DAEMON" ] || exit 0

# Read configuration variable file if it is present
[ -r /etc/default/$NAME ] && . /etc/default/$NAME

:wq!
```

- нажимаем **enter**
- скопируем это в системный каталог

```
sudo cp ~/src/mod_tile/debian/renderd.init /etc/init.d/renderd
```

```
alisa@linux:~$ sudo cp ~/src/mod_tile/debian/renderd.init /etc/init.d/renderd
```

```
sudo chmod u+x /etc/init.d/renderd
```

```
alisa@linux:~$ sudo chmod u+x /etc/init.d/renderd
```

```
sudo cp ~/src/mod_tile/debian/renderd.service /lib/systemd/system/
```

```
alisa@linux:~$ sudo cp ~/src/mod_tile/debian/renderd.service /lib/systemd/system
/
```

- Файл «render.service» является служебным файлом «systemd». Используемая здесь версия просто вызывает команды инициализации в старом стиле. Чтобы проверить, что команда запуска работает:

```
sudo /etc/init.d/renderd start
```

```
alisa@linux:~$ sudo /etc/init.d/renderd start
```

- (ответить «Starting renderd (через systemctl): renderd.service».)

```
Starting renderd (via systemctl): renderd.service.
alisa@linux:~$
```

- Чтобы это запускалось автоматически каждый раз:

```
sudo systemctl enable renderd
```

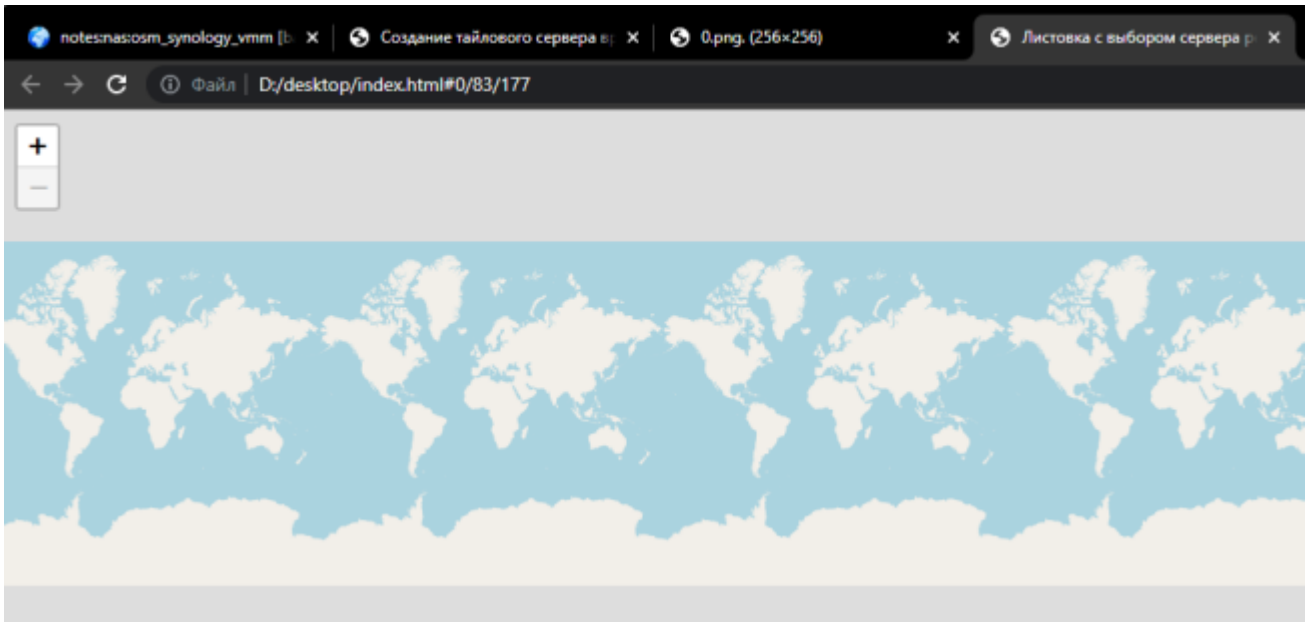
```
alisa@linux:~$ sudo systemctl enable renderd
```

Просмотр тайлов (плиток)

для просмотра плитки, используйте html-файл «sample_leaflet.html» в папке «extra» mod_tile или загрузите zip архив с этим файлом внизу страницы. Просто откройте его в веб-браузере на машине, где вы установили тайловый сервер. Если это невозможно, потому что вы устанавливаете на сервер без локального веб-браузера, вы можете отредактировать его, заменив «127.0.0.1» IP-адресом сервера и скопировав его ниже «/var/www/html». Мы запускаем его по сети с указанием IP-адреса виртуального сервера NAS Synology (192.168.1.31)

Пример загрузки карты

- Начало работы и загрузка 0 level zoom карты России

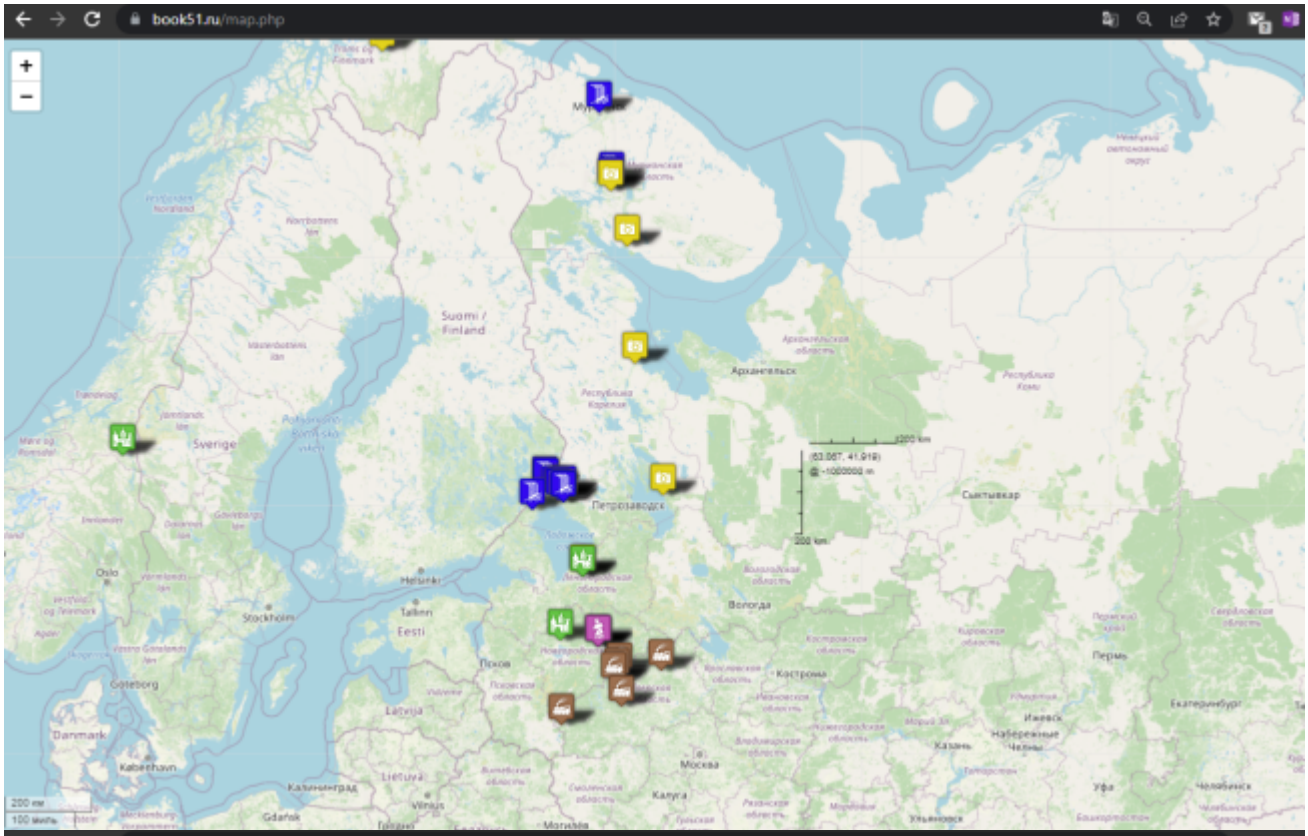


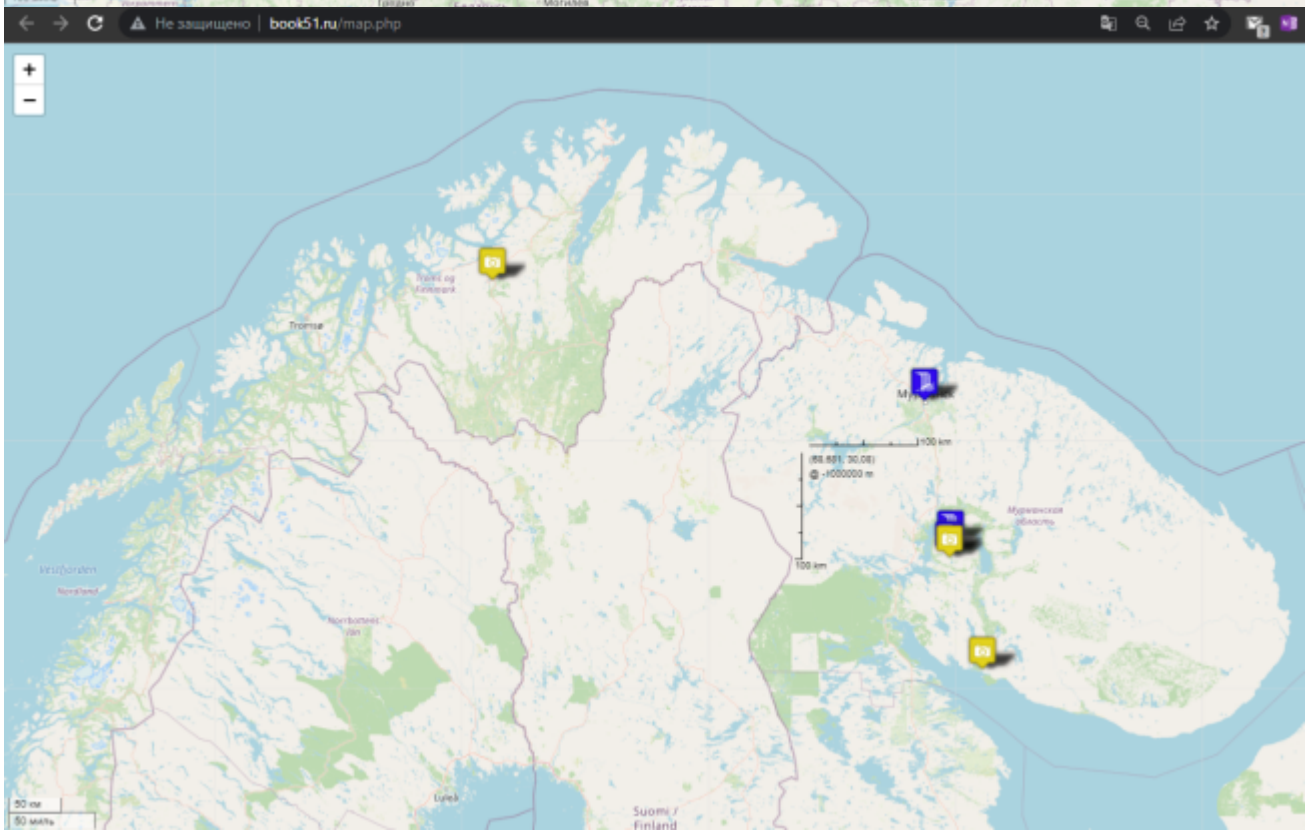
- маленькое видео, как за 1 час происходит визуальный рендеринг 9 level zoom карты России на 3,2gb на устройстве Synology Nas DS420+. Стоит перематывать и смотреть загрузку процессора, так как память неизменна в 80%. Уровни с 1 по 4 пишутся в кэш за 15-20 минут… 8 уровень около 30 минут… Ну а 9 на видео.. немногим более часа визуальный рендеринг карты России за 1 час на 9 уровне зума. Напомню что установлена дополнительная планка памяти на 8gb.

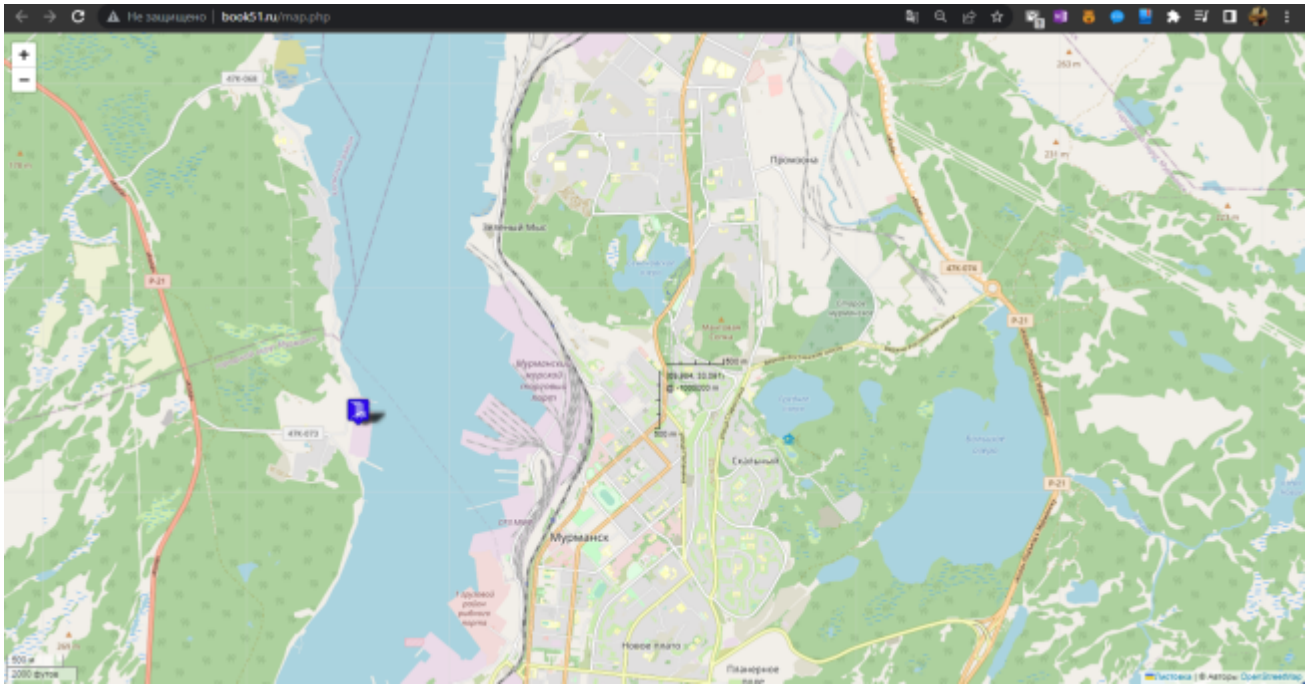
zoom_9.mp4

- в принципе, считаю проделанную работу, не плохим результатом! И OSM Tale Server имеет право существовать на устройстве Synology NAS DS 420+ для личного пользования или небольшой группы пользователей личного сервера OSM.
- Напомню, что данный Ubuntu Tile Server установлен на виртуальную машину Synology VMM на устройстве DS420+, где работает все, что было установлено ранее. Попробую

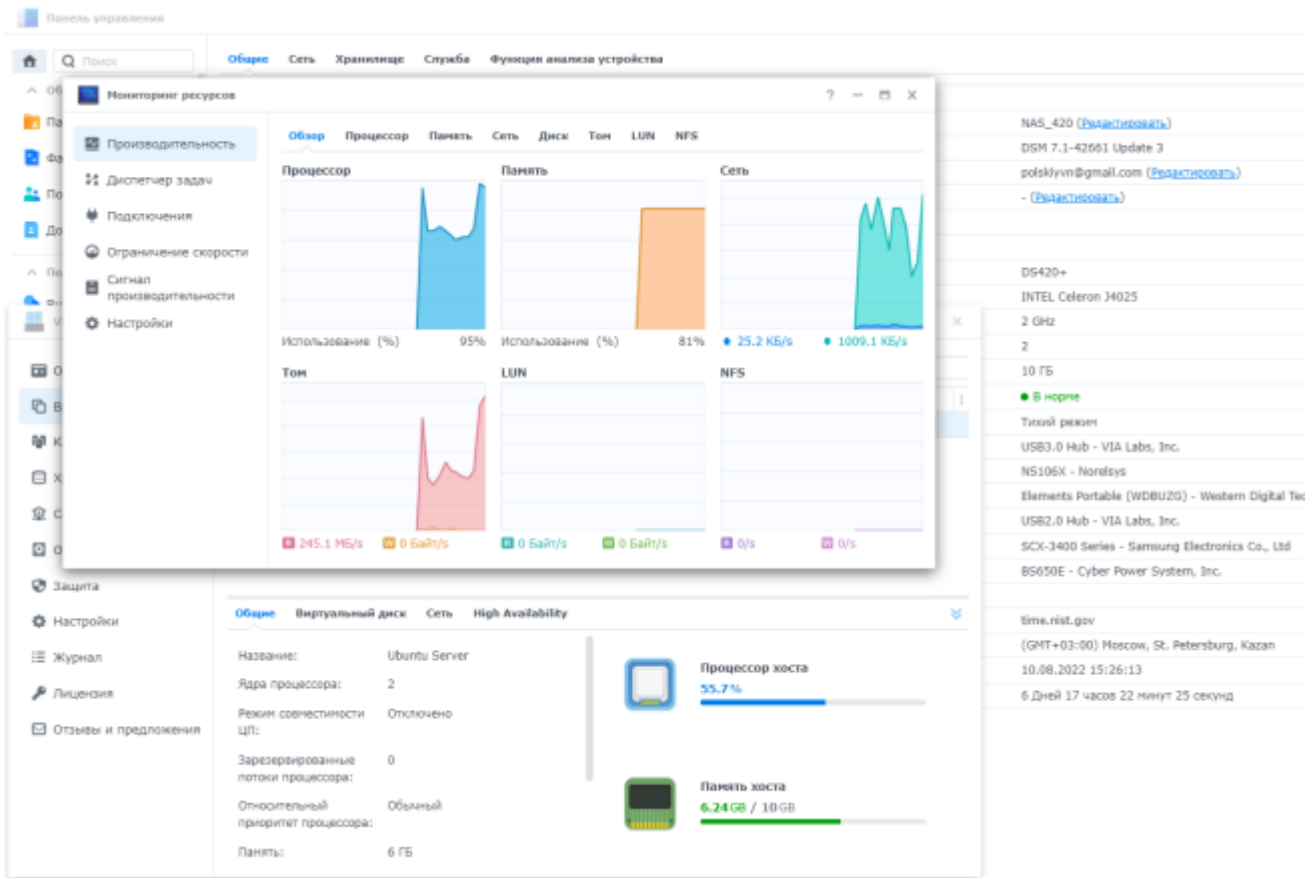
переключить мой личный сайт https://wwoss.ru/ с картой и слоями на вновь созданный виртуальный сервер.



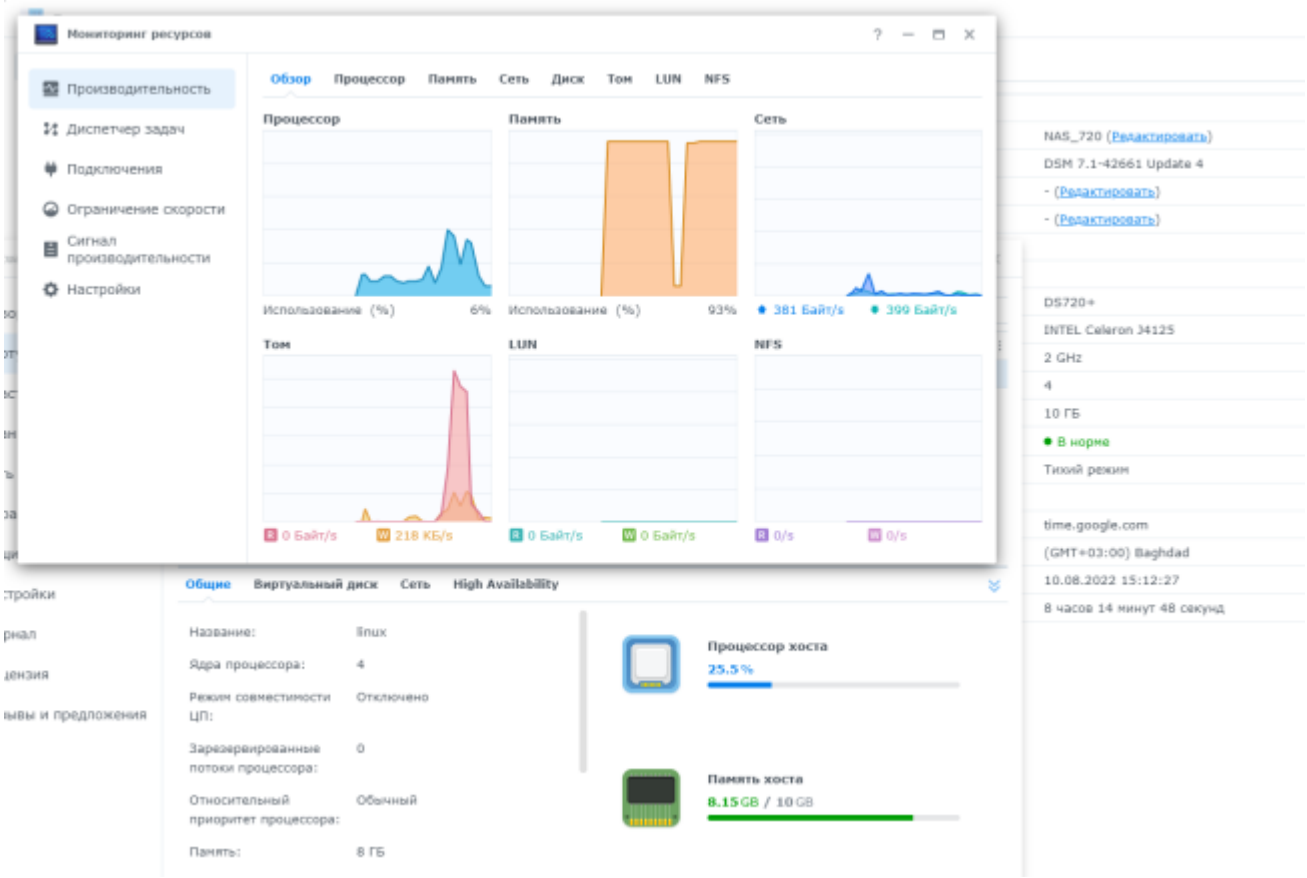




- Загрузка Synolody DS420+ при работающем OSM Tile Server на Synology VMM (8gb VMM не дает установить система, поэтому стоит 6gb)



- Для сравнения аналогичная работа OSM Tile Server на Synology VMM на Synolody DS720+ и загрузка процессора



О 32ядрах и 1tb ssd * 3шт в raid5

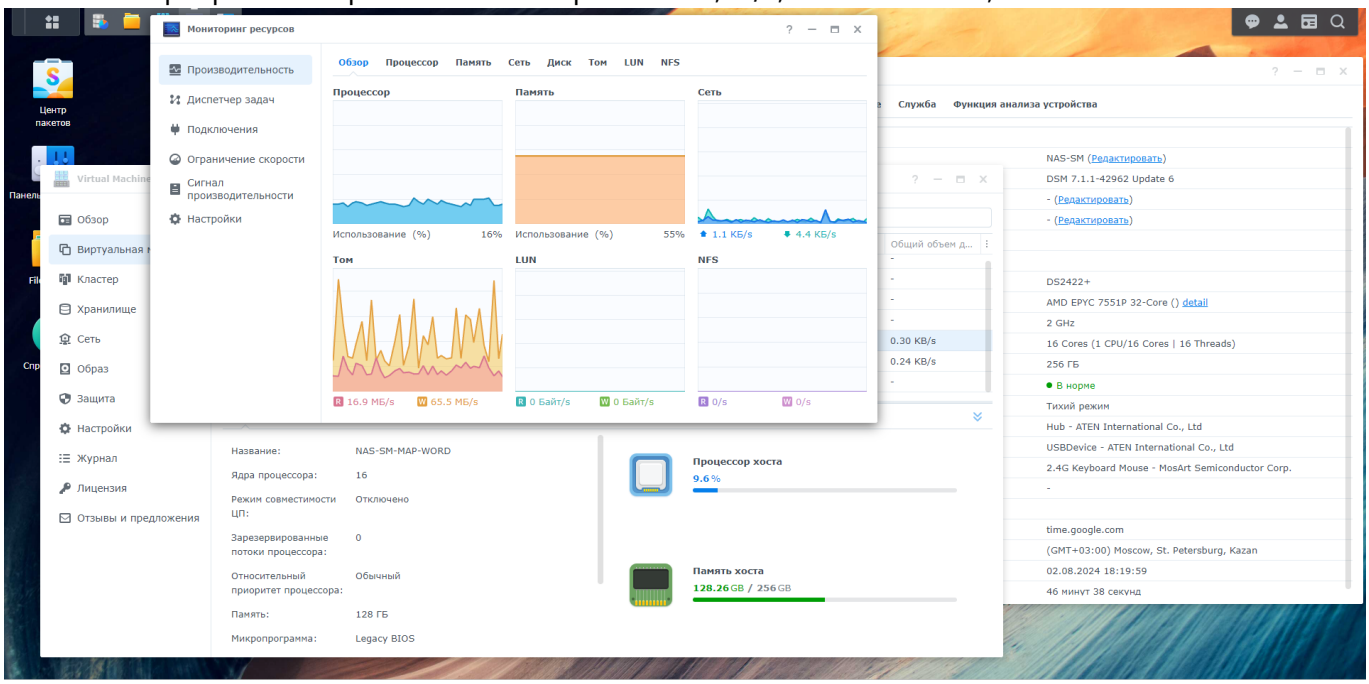
Загрузка всей карты с https://planet.openstreetmap.org/pbf/planet-latest.osm.pbf ограниченна пропускной способностью провайдера 12Mb/s - 5 часов. С соседнего сервера по внутренней сети 188Mb/s - 7 минут ограничение в скорости hdd соседнего сервера.

```
alisa@sm-map:~/data$ wget http://192.168.1.112/openstreetmap/planet-latest/planet-latest.osm.pbf
--2024-08-02 14:49:44--  http://192.168.1.112/openstreetmap/planet-latest/planet-latest.osm.pbf
Connecting to 192.168.1.112:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 81733724534 (76G) [application/vnd.openstreetmap.data+xml]
Saving to: 'planet-latest.osm.pbf'

planet-latest.osm.pbf         100%[=============================================>]  76.12G   249MB/s    in 6m 55s

2024-08-02 14:56:39 (188 MB/s) - 'planet-latest.osm.pbf' saved [81733724534/81733724534]
```

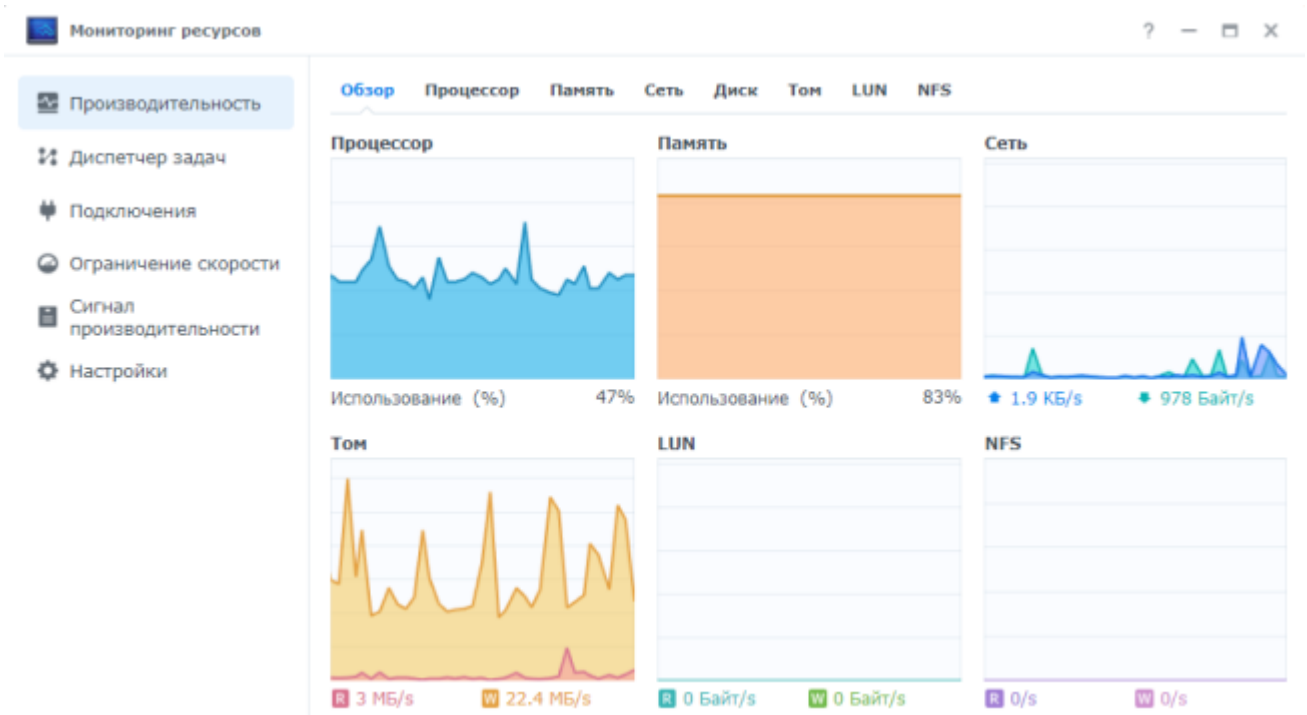
Использование парсера PBF

```
alisa@sm-map:~/data$ osm2pgsql -d gis --create --slim  -G --hstore --tag-transform-script ~/src/openstreetmap-carto/o
penstreetmap-carto.lua -C 128000 --number-processes 16 -S ~/src/openstreetmap-carto/openstreetmap-carto.style ~/data/
planet-latest.osm.pbf
osm2pgsql version 1.2.1 (64 bit id space)

Allocating memory for dense node cache
Allocating dense node cache in one big chunk
Allocating memory for sparse node cache
Sharing dense sparse
Node-cache: cache=128000MB, maxblocks=2048000*65536, allocation method=11
Mid: pgsql, cache=128000
Setting up table: planet_osm_nodes
Setting up table: planet_osm_ways
Setting up table: planet_osm_rels
Using lua based tag processing pipeline with script /home/alisa/src/openstreetmap-carto/openstreetmap-carto.lua
Using projection SRS 3857 (Spherical Mercator)
Setting up table: planet_osm_point
Setting up table: planet_osm_line
Setting up table: planet_osm_polygon
Setting up table: planet_osm_roads

Reading in file: /home/alisa/data/planet-latest.osm.pbf
Using PBF parser.
Processing: Node(302430k 292.8k/s) Way(0k 0.00k/s) Relation(0 0.00/s)
```

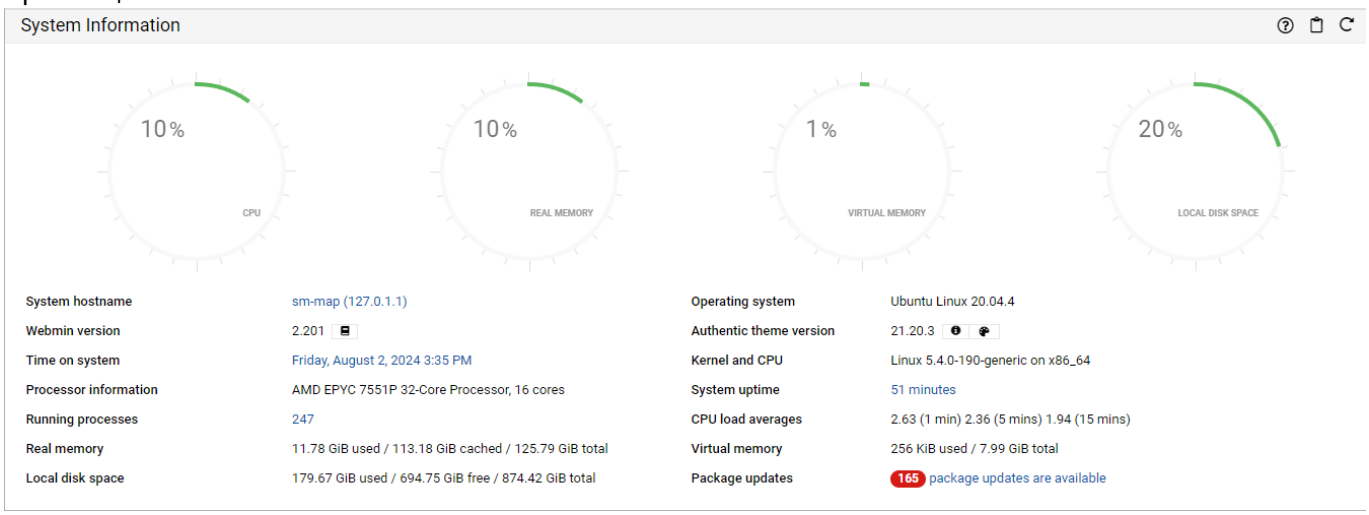
Россия в парсере Node при этих же настройка 516,5k/s, заняла 2 часа,



то же самое на Synology DS 720+



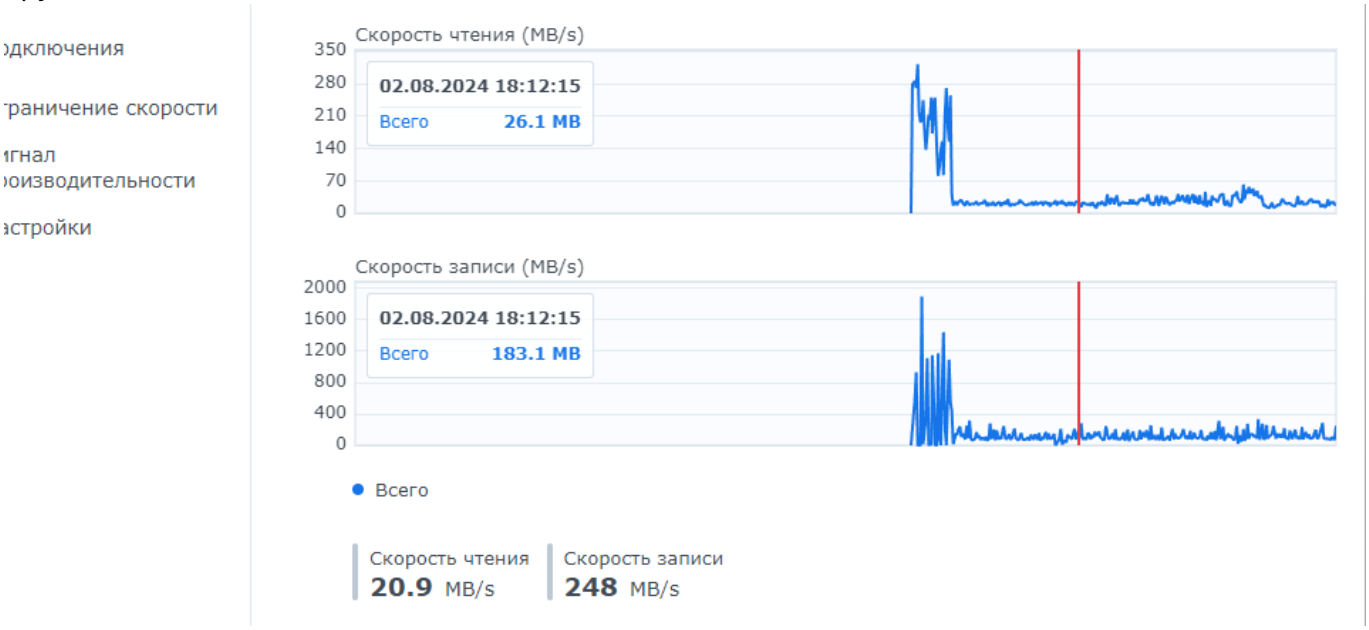
(Intel Celeron J4125 4*2,70 GHz / 2gb ddr4 + 8gb ddr4 / 2*1tb ssd) показала загрузку процессора в среднем 30-50% при выделенной виртуальной памяти в 8gb, которая занята соответственно на 85% от общего объема в 10gb. Реально используется только 12% озу от выделенных 128Gb, при общем объеме в 256Gb



При этом процессор от 9% до 17%



Трудятся только ssd



Файлы для загрузки

- Ubuntu Server 20.04.2 LTS
- html-файл для просмотра плиток
- Mod_tile: модуль веб-сервера Apache
- Leaflet — библиотека для интерактивных карт.
- OSM Carto: стиль стандартного слоя листов OSM
- water-polygons-split-3857.zip
- ne_110m_admin_0_boundary_lines_land.zip
- antarctica-icesheet-outlines-3857.zip
- antarctica-icesheet-polygons-3857.zip
- simplified-water-polygons-split-3857.zip

Карты OSM

- azerbaijan-latest.osm.pbf
- russia-latest.osm.pbf
- planet-240506.osm.pbf

Ссылки и литература

Создание тайлового сервера вручную (20.04 LTS)
Ubuntu 20.04 LTS: пользовательская документация
OSM2PGSQL: импортер данных OSM в базу данных PostgreSQL/PostGIS
PostgreSQL: база данных с открытым исходным кодом
Mapnik: отрисовка основного Slippy Map слоя для OSM
OSM Carto: стиль стандартного слоя листов OSM
Mod_tile: передача кэшированных тайлов
Leaflet — библиотека JS для интерактивных карт.
Synology Open Street Map Server
https://github.com/openstreetmap/mod_tile/blob/master/docs/build/building_on_ubuntu.md
https://switch2osm.org/serving-tiles/manually-building-a-tile-server-ubuntu-24-04-lts/
https://www.linuxbabe.com/linux-server/osm-openstreetmap-tile-server-ubuntu-22-04
https://ircama.github.io/osm-carto-tutorials/kosmtik-ubuntu-setup/
https://www.linuxbabe.com/debian/openstreetmap-osm-tile-server-debian-12-bookworm
https://github.com/openstreetmap/mod_tile

From:
https://wwoss.ru/ - **worldwide open-source software**

Permanent link:
https://wwoss.ru/doku.php?id=software:linux_server:tile_map_server_22.04.4

Last update: **2025/12/06 19:56**