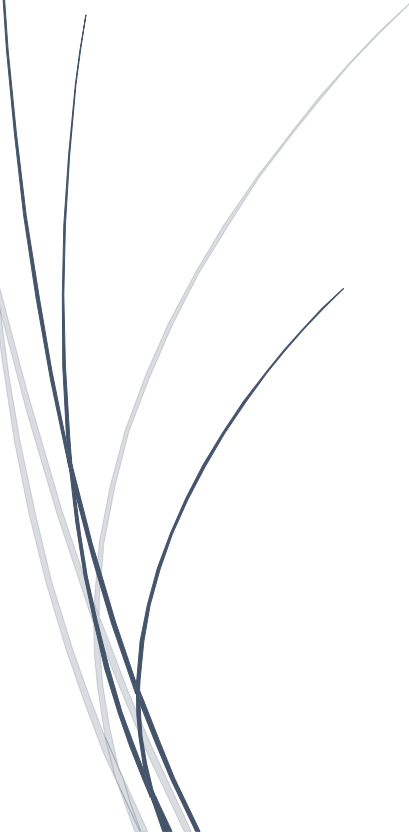


19.12.2021

Платформа OMMG 3.0



Георгий Хазан
ОММДЖИ ТЕХНОЛОДЖИ

Требования к аппаратному обеспечению и системному ПО.

Для развёртывания платформы OMMG 3.0 (далее - Платформы) требуются:

- для развёртывания корпоративной Платформы при нагрузке не свыше 5000 соединений требуется/рекомендуется компьютер с 16/32 ГБ ОЗУ, 4/8 ядрами серверных процессоров класса Intel E5-2620v4 и 3/5 жёсткими дисками, объединёнными в RAID10.
- для развёртывания Платформы, рассчитанной на большие нагрузки, применяются отдельные сервера трёх видов:
 - о фронтенд-сервер (www) - сервер с 24 или 32 ядрами и достаточным объёмом памяти. Такие сервера нетребовательны к жёстким дискам. В качестве примера можно взять сервер на шасси DELL R420 с блоком питания 550W/, два процессора E5-2430, 64 ГБ памяти (8x8GB), два винчестера 300GB SAS/H710.
 - о сервер баз данных - с развитой дисковой подсистемой и большим объёмом памяти. Например, DELL R520 с блоком питания 750W/, двумя процессорами E5-24302, 192 ГБ памяти (12x16GB) и шестью винчестерами 2TB SATA/H710. Для поддержки 100 тысяч пользователей в онлайн понадобятся минимум два таких сервера, для миллиона - четыре.
 - о файловый сервер - соединяет высокую вычислительную мощь для конвертации закачанных звуковых и видео-сообщений в общий формат с большой дисковой ёмкостью. Типичная конфигурация: сервер на шасси DELL R720XD с блоком питания 750W/, двумя процессорами E5-2650, 128 ГБ памяти (8x16GB) и двенадцатью винчестерами 3TB SATA/H710. Обычно для конфигурации до 100 тысяч в онлайн хватает одного такого сервера, для миллиона в онлайн - в зависимости от объёма закачанных файлов. Может предусматриваться возможность стирания старых файлов (например, спустя месяц), если ёмкости дисков перестанет хватать.
- все компьютеры работают под управлением ОС Linux (Ubuntu/64 16.04.4 или совместимой с ней).

Развёртывание базы данных.

Для того, чтобы повысить нагрузочную способность Платформы, её база данных может быть разбита на 16 или 256 частей (шардов), каждая из которых может исполняться на отдельном компьютере. При возрастании нагрузки можно наращивать число физических компьютеров, обслуживающих базу данных, без остановки сервиса.

Установка сервера MySQL.

Эта операция производится на каждом сервере БД.

- От имени root необходимо исполнить команду:

```
#apt-get install -y mysql-server
```

- Отредактируйте настройки сервера MySQL (имя файла настроек приведено по умолчанию, оно может изменяться в зависимости от версии MySQL и конкретной установки):

```
#vim /etc/mysql/mysql.conf.d/mysqld.cnf
```

- если БД устанавливается на несколько серверов, то необходимо убрать привязку MySQL к локальному интерфейсу:

```
# Instead of skip-networking the default is now to listen only on
```

```
# localhost which is more compatible and is not less secure.
```

```
# bind-address          = 127.0.0.1
```

- если у вас версия MySQL младше 8.0, добавьте настройку sql-mode:

```
sql-mode=ONLY_FULL_GROUP_BY,NO_AUTO_CREATE_USER,NO_ENGINE_SUBSTITUTION
```

- если у вас версия MySQL 8.0 или старше, добавьте настройку sql-mode:

```
sql-mode=ONLY_FULL_GROUP_BY,NO_ENGINE_SUBSTITUTIONесли
```

- если сервер устанавливается на Ubuntu 16.x или старше, поправьте количество открытых файлов для сервера MySQL:

```
#mkdir /etc/systemd/system/mysql.service.d
```

```
#vim /etc/systemd/system/mysql.service.d/nofile.conf
```

```
[Service]
```

```
LimitNOFILE=20000
```

```
#systemctl daemon-reload
```

- перезапустите сервер MySQL:

```
#systemctl restart mysql
```

- создайте конфигурационный файл для MySQL:

```
#vim ~/.my.cnf
```

```
[client]
```

```
#port = 3306
```

```
socket = /var/run/mysqld/mysqld.sock
```

```
user = root
```

```
password = xxxxxxxxxxxxxxxx
```

- Войдите в командную строку:

```
#mysql
```

```
Welcome to the MySQL monitor.  Commands end with ; or \g.
```

```
Your MySQL connection id is 11820667
```

```
Server version: 5.5.50-0ubuntu0.14.04.1 (Ubuntu)
```

```
Copyright (c) 2000, 2016, Oracle and/or its affiliates. All rights reserved.
```

```
Oracle is a registered trademark of Oracle Corporation and/or its  
affiliates. Other names may be trademarks of their respective  
owners.
```

```
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

```
mysql>
```

- Создайте пользователя для доступа к БД:

```
mysql>create user db@localhost identified by 'xxxxxxxxxxxx';
```

- если база данных и веб-фронтенд находятся на разных машинах, добавьте этого же пользователя для удалённых подключений:

```
mysql>create user db@`%` identified by 'xxxxxxxxxxxx';
```

- Для версии MySQL 8.0 или старше:

- Выдайте пользователю db права на доступ к данным:

```
mysql>grant all privileges on *.* to 'db'@'localhost';
```

```
mysql>grant all privileges on *.* to 'db'@'%';
```

- Для версии MySQL менее 8.0:

- Получите список пользователей и найдите там хэш пароля пользователя db:

```
mysql> select User, Host, authentication_string from mysql.user;
```

```
+-----+-----+-----+  
| db | % | *XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX |  
+-----+-----+-----+
```

- Выдайте пользователю db права на доступ к данным, используя этот хэш:

```
mysql>grant all privileges on *.* to 'db'@'localhost' identified by password  
'*XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX';
```

```
mysql>grant all privileges on *.* to 'db'@'%` identified by password  
'*XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX';
```

- залейте структуру базу данных из скриптов, входящих в дистрибутив системы:

```
#cd /var/www/ommg3_api/mysql
```

```
#cat db.sql | mysql db
```

```
#cat dbglobal.sql | mysql db
```

```
#cat dbstats.sql | mysql dbstats
```

```
#cat stickers.sql | mysql stickers
```

```
#for i in `seq 1 16`; do cat dbshard.sql | mysql db$i; done
#for i in `seq 1 16`; do cat sprocs.sql | mysql db$i; done
```

Развёртывание веб-фронтенда

Данный набор операций производится на каждой из машин, предназначенных для функционирования в качестве www-сервера.

Установка memcached

Memcached применяется в Платформе для кэширования слабонагруженных элементов, таких, как регистрация пользователей.

- для установки отдайте команду:

```
#apt-get install -y memcached
```
- для установки платформы на нескольких серверах установите memcached только на одном сервере и прокомментируйте настройку `-l` в файле `/etc/memcached.conf`

```
# -l 127.0.0.1
```

Установка php-fpm

- установите серверную часть php:

```
$sudo apt-get install -y php-fpm
```
- настройте его и перезапустите сервер

```
$vim /etc/php/7.0/fpm/pool.d/www.conf
;listen = /var/run/php7-fpm.sock
listen = 127.0.0.1:9001
```
- перезапустите сервер

```
$sudo service php7.0-fpm restart
```

Установка php

В настоящий момент Платформа использует php версии 7.x или совместимый с ним:

- установите сам php:

```
$sudo apt-get install -y php
```
- установите расширения php:

```
$sudo apt-get install -y php-memcache php-curl php-json php-mcrypt \
php-mysql php-imagick
```
- отконфигурируйте файлы `/var/www/ommg3_api/conf/default.php` и `/var/www/ommg3_api/conf/sconfig.php` таким образом, чтобы там были верные названия доменов, имена пользователей, пароли на базу данных и прочее.
- создайте каталог для логов:

```
#mkdir /var/log/php_log
```

```
#chown www-data /var/log/php_log
#chgrp www-data /var/log/php_log
#chmod 775 /var/log/php_log
```

- запустите конфигуратор базы данных:

```
$cd /var/www/ommg3_api/test
$php autoinc.php
```

- убедитесь, что php имеет нормальный доступ к MySQL, и вышеозначенный скрипт выполнен без ошибок.

Установка nginx

Для функционирования платформы требуется nginx версии 1.21.1 или старше. Установка производится из исходного кода.

- скачайте исходный код nginx и распакуйте его:

```
$cd /opt
$wget https://nginx.org/download/nginx-1.21.1.tar.gz
$tar -xzvf nginx-1.21.1.tar.gz
```

- скопируйте исходный код модулей:

```
$cp -R /var/www/nginx/modules/* /opt
```

- установите нужные библиотеки:

```
$sudo apt-get install -y nginx libssl-dev libxslt1-dev libpcre3 \
libpcre3-dev libgd-dev libgeoip-dev
```

- если у вас не установлен компилятор C++, установите его:

```
apt install -y build-essential
```

- сконфигурируйте nginx:

```
$cd /opt/nginx-1.21.1
$./configure --prefix=/etc/nginx --sbin-path=/usr/sbin/nginx \
--modules-path=/usr/lib/nginx/modules \
--conf-path=/etc/nginx/nginx.conf \
--error-log-path=/var/log/nginx/error.log \
--http-log-path=/var/log/nginx/access.log \
--pid-path=/var/run/nginx.pid \
--lock-path=/var/run/nginx.lock \
--http-client-body-temp-path=/var/cache/nginx/client_temp \
--http-proxy-temp-path=/var/cache/nginx/proxy_temp \
--http-fastcgi-temp-path=/var/cache/nginx/fastcgi_temp \
--http-uwsgi-temp-path=/var/cache/nginx/uwsgi_temp \
--http-scgi-temp-path=/var/cache/nginx/scgi_temp --user=nginx \
--group=nginx --with-http_ssl_module \
--with-http_realip_module --with-http_addition_module \
--with-http_sub_module --with-http_dav_module \
--with-http_flv_module --with-http_mp4_module \
```

```
--with-http_gunzip_module --with-http_gzip_static_module \  
--with-http_random_index_module --with-http_secure_link_module \  
--with-http_stub_status_module --with-http_auth_request_module \  
--with-http_xslt_module=dynamic \  
--with-http_image_filter_module=dynamic \  
--with-http_geoip_module=dynamic \  
--with-threads --with-stream --with-stream_ssl_module \  
--with-http_slice_module --with-mail --with-mail_ssl_module \  
--with-http_v2_module \  
--add-module=/opt/headers-more-nginx-module-0.30 \  
--with-http_image_filter_module \  
--add-module=/opt/nginx_upload_module-2.2.0 \  
--add-module=/opt/nginx-upload-progress-module
```

- остановите предустановленный nginx:
`$sudo nginx -s stop`
- соберите и установите nginx:
`$make`
`$sudo make install`
- настройте виртуальные хосты:
`#cp /var/www/nginx/conf.d/* /etc/nginx/conf.d`
`#cp /var/www/nginx/nginx.conf /etc/nginx`
- создайте сертификат для ssl
`#mkdir /etc/nginx/ssl`
`#cd /etc/nginx/ssl`
`#openssl req -x509 -nodes -days 3650 -newkey rsa:2048 -keyout ommg.key \
-out ommg.crt`
`#cp /var/www/nginx/ssl/ssl.conf /etc/nginx/ssl`
`#vim /etc/nginx/ssl/ssl.conf`
- создайте временный каталог для файлов:
`#mkdir /var/cache/nginx`
- запустите nginx:
`#nginx`
- убедитесь в том, что после перезапуска nginx отсутствуют ошибки

Установка redis

Redis (ver_2.8.4) используется в Платформе для высоконагруженных кэшей прикладного сервера.

- установите собственно сервер:
`$sudo apt-get install -y redis-server`
- настройте порт для слушания:
`#vim /etc/redis/redis.conf`

```
# Accept connections on the specified port, default is 6379.
# If port 0 is specified Redis will not listen on a TCP socket.
port 6301
```

- настройте оповещение о событиях:

```
# By default all notifications are disabled because most users don't
need
# this feature and the feature has some overhead. Note that if you
don't
# specify at least one of K or E, no events will be delivered.
notify-keyspace-events "Ex"
```

- перезапустите сервер

```
service redis-server restart
```

- для особо нагруженных систем рекомендуется устанавливать по два экземпляра redis на машине, на портах 6301 и 6302.

- для установки платформы на несколько серверов отключите привязку redis к localhost, закомментировав настройку bind в файле /etc/redis/redis.conf:
bind 127.0.0.1

Установка node.js

В Платформе используется node.js версии 4.2.6 (входит по умолчанию в набор пакетов):

- установите nodejs и packet manager:

```
#apt-get install -y nodejs nodejs-legacy npm libzmq5 libzmq3-dev
```

- убедитесь, что установлена правильная версия node.js

```
#!/usr/bin/nodejs -v
4.2.6
```

- настройте файл /var/www/ommg3_nodejs/config.json

- скопируйте файлы сервисов:

```
#cp /var/www/ommg3_nodejs/systemd/* /etc/systemd/system
```

- запустите прикладной сервер:

```
#systemctl start ommg3_storage.service
#systemctl start ommg3.service
```

- если нужно, чтобы сервисы стартовали при запуске операционной системы, используйте команду enable:

```
#systemctl enable ommg3_storage.service
#systemctl enable ommg3.service
```

Установка файлового сервера

Если файловый сервер ставится на отдельный сервер, то установите сначала php и nginx, как описано в предыдущем разделе. Настройте файл

/var/www/ommg3_files/sconfig.php так же, как и для ommg3_api.

- редактирование настроек и заливка базы данных:

```
$cd /var/www/ommg3_files/sql  
$vim global.sql  
>замените домен после слов "storage #1" на нужный  
$cat global.sql | mysql db
```
- установка необходимых утилит

```
#apt-get install -y imagemagick libav-tools gawk  
#ln -s /usr/bin/avconv /usr/bin/ffmpeg  
#ln -s /usr/bin/avprobe /usr/bin/ffprobe
```
- запуск конвертора медиа-файлов

```
$cd /var/www/ommg3_files/scripts  
$php convert_multi.php &
```
- для загрузки больших файлов возможно потребуется внести изменения в настройки nginx и php:

```
#vim /etc/nginx/nginx.conf  
client_max_body_size 100M;  
#nginx -s reload  
#vim /etc/php/7.0/cli/php.ini  
; Maximum allowed size for uploaded files.  
; http://php.net/upload-max-filesize  
upload_max_filesize = 20M  
#vim /etc/php/7.0/fpm/php.ini  
; Maximum allowed size for uploaded files.  
; http://php.net/upload-max-filesize  
upload_max_filesize = 20M  
#service php7.0-fpm restart
```
- создание системного пользователя (0001@), от имени которого Платформа будет рассылать свои сообщения:

```
$cd /var/www/ommg3_api/pmi  
$php 0001.php
```