



Инструкция по установке экземпляра ПО
" Модульная система «Цифровой город»"



Инструкция по установке экземпляра ПО " Модульная система «Цифровой город»"



Оглавление

Настройка сервера	3
Настройка окружения для Debian	3
1.1. <i>Установка</i>	3
1.2. <i>Установка пакетов</i>	4
1.3. <i>Конфигурация Nginx</i>	6
1.4. <i>Конфигурация PHP</i>	8
1.5. <i>Конфигурация</i>	10
1.6. <i>Конфигурация MariaDB</i>	12
1.7. <i>Конфигурация push-server</i>	13
1.8. <i>Конфигурация портала</i>	18
Настройка корпоративного портала	21
Установка экземпляра ПО	23
Контактная информация	24

Настройка сервера

Для корректной работы на устройстве заказчика должно быть установлено ПО «1С-Битрикс24» (Запись в реестре от 20.02.2016 №25, Произведена на основании приказа Министерства цифрового развития, связи и массовых коммуникаций Российской Федерации от 20.02.2016 №63, <https://reestr.digital.gov.ru/reestr/301374/> , <https://www.1c-bitrix.ru/products/bitrix24/about.php>) на ОС Debian. ПО «1С-Битрикс24» приобретается заказчиком самостоятельно.

Рекомендуемая конфигурация:

- **До 12 пользователей:**
2 ядра 5,0 ГГц / 4 Гб оперативной памяти / Диск 120 Гб NVMe SSD
- **До 50 пользователей:**
3 ядра 5,0 ГГц / 6 Гб оперативной памяти / Диск 240 Гб NVMe SSD
- **50 – 100 пользователей:**
4 ядра 5,0 ГГц / 8 Гб оперативной памяти / Диск 360 Гб NVMe SSD
- **100 – 300 пользователей:**
4 ядра 5,0 ГГц / 8 Гб оперативной памяти / Диск 540 Гб NVMe SSD
- **300 – 500 пользователей:**
5 ядер 5,0 ГГц / 10 Гб оперативной памяти / Диск 840 Гб NVMe SSD

Настройка окружения для Debian

1.1. Установка

Установку необходимо выполнять с диска с минимальным набором ПО, остальное будет установлено по сети во время настройки.

В процессе установки выберите сервер с минимальной настройкой, в противном случае получите десктоп. Дальнейшая настройка будет показана на базе такой установки.

В качестве менеджера пакетов используется **apt/apt-get**. Обновите систему до последней стабильной версии. Отключите selinux:



```
apt update && apt upgrade
```

```
echo 'SELINUX=disabled' > /etc/selinux/config
```

```
reboot
```

Настройка пакетов

Обязательно нужно открыть порты:

- 22 – ssh доступ;
- 80 / 443 – http / https web-сервер;

Остальные порты для:

- ntlm
- сервера мгновенных сообщений;
- xmpp-сервера

нужно открывать, если только они используются. Можно выбрать произвольные порты, а можно те же, что используются в Debian :

- 8890 / 8891 – http/https ntlm;
- 8893 / 8894 – http/https сервер мгновенных сообщений;
- 5222 / 5223 – http/https.

1.2. Установка пакетов

В дефолтном репозитории нет PHP 8.0, но есть сторонние, которые позволяют поставить необходимое ПО.

```
apt install -y lsb-release ca-certificates apt-transport-https software-properties-common gnupg2
```

Используйте репозиторий: packages.sury.org.



Настройте репозиторий, симпортируйте ключ репозитория, обновите список пакетов:

```
echo "deb https://packages.sury.org/php/ $(lsb_release -sc) main" | \

sudo tee /etc/apt/sources.list.d/sury-php.list

wget -qO - https://packages.sury.org/php/apt.gpg | \

sudo apt-key add -

apt update
```

Установите **apache 2.4** и **php 8.0**:

```
# apt install apache2 -y

# apt install php8.0 php8.0-cli \

php8.0-common php8.0-gd php8.0-ldap \

php8.0-mbstring php8.0-mysql \

php8.0-opcache \

php-pear php8.0-apcu php-geoip \

php8.0-mcrypt php8.0-memcache\

php8.0-zip php8.0-pspell php8.0-xml -y
```

Установите **nginx** (1.18 версия):

```
apt install nginx
```



Установите **MariaDB** сервер (10.5 версия):

```
apt -y install mariadb-server mariadb-common
```

Установите **node** и **npm** (push-server) - 12.22

```
apt install nodejs npm -y
```

Установите **redis** - 6.0:

```
apt install redis -y
```

1.3. *Конфигурация Nginx*

Рабочий каталог для сайта - **/var/www/html/bx-site**. Пользователь для web окружения - **nginx**, группа **apache**.

Конфигурация Nginx сервера:

```
/etc/nginx/nginx.conf          # основной конфигурационный файл

    _conf.d/upstreams.conf      # конфигурация для upstream серверов: apache
&& push-server

    _conf.d/maps-composite_settings.conf  # переменные используемые для кеша

    _conf.d/maps.conf           # дополнительные переменные

    _conf.d/http-add_header.conf  # CORS заголовки

    _sites-available/*.conf      # подключаем сайты
```



```
    |_default.conf          # сайт по умолчанию (настраиваем только 80 порт)

    |_conf.d/bx_temp.conf          # конфигурация
BX_TEMPORARY_FILES_DIRECTORY

    |_conf.d/bitrix.conf # дефолтная конфигурация сайта

    |_rtc.conf          # проксирование запросов на push-server
(публикация)Копировать
```

Дефолтная конфигурация сайта:

```
conf.d/bitrix.conf          # основной блоки со включенным по умолчанию
кешем в файлах

    |_conf.d/bitrix_general.conf # отдача статики, быстрая отдача для внешних
хранилищ и прочее

    |_conf.d/errors.conf      # обработка ошибок

    |_conf.d/im_subscrider.conf # проксирование запросов на push-server
(получение)

    |_conf.d/bitrix_block.conf # блокировки по умолчанию
```

Конфигурация взята из виртуальной машины и может показаться избыточной, но фактически поддерживает все возможности, что и виртуальная машина.

Все конфигурационные файлы можно [скачать в архиве](#).

```
su -
```

```
rsync -av debian/nginx/ /etc/nginx/
```



В сервисе используются имена для проксирования на определенные службы:

- **httpd** - проксирование запросов на apache,
- **push** - проксирование запросов на push-server.

Чтобы заработала конфигурация, необходимо прописать службы в локальных адресах. Если сервисы расположены на другом хосте, то укажите здесь правильный адрес.

```
echo "127.0.0.1 push httpd" >> /etc/hosts
```

По умолчанию в Debian Apache2 сервер использует 80 порт и поставлен на автозапуск. Поэтому перед запуском nginx сервера, на время выключите Apache2 (на данный момент он еще не настроен). Остановите Apache2:

```
systemctl stop apache2
```

Запустите Nginx:

```
systemctl --now enable nginx
```

1.4. Конфигурация PHP

В данной версии установки централизованное хранилище конфигов:

```
/etc/php/8.0
```

```
|---- apache2
```

```
|   |-> conf.d/
```

```
|   |-> php.ini
```

```
|---- cli  
  
|   |-> conf.d/  
  
|   |-> php.ini  
  
|---- mods-available  
  
    |-> .ini
```

Файлы **conf.d** внутри каталогов **/apache2** и **/cli** содержат ссылки на **mods-available**. То есть в дефолтной конфигурации и модуль **apache2** и командная строка будут содержать одинаковый набор модулей с одинаковыми параметрами.

Добавьте настройки для следующих опций:

- для модуля **opcache**:

- **opcache.max_accelerated_files = 100000**

```
opcache.revalidate_freq = 0
```

- настройки **bitrexenv.ini**:

- **display_errors = Off**
- **error_reporting = E_ALL**
- **error_log = '/var/log/php/error.log'**
- **; Set some more PHP parameters**
- **enable_dl = Off**

- `short_open_tag = On`
- `allow_url_fopen = On`
- `# Security headers`
- `mail.add_x_header = Off`
- `expose_php = Off`

Конфигурационные файлы для PHP расположены в папке [debian/php.d](#).

`su -`

```
rsync -av debian/php.d/ /etc/php/8.0/mods-available/
```

```
In -sf /etc/php/8.0/mods-available/zbx-bitrix.ini /etc/php/8.0/apache2/conf.d/99-bitrix.ini
```

```
In -sf /etc/php/8.0/mods-available/zbx-bitrix.ini /etc/php/8.0/cli/conf.d/99-bitrix.ini
```

1.5. Конфигурация

По умолчанию конфигурация **Apache** устроена следующим образом:

```
# /etc/apache2/

# |-- apache2.conf

# | `-- ports.conf

# |-- mods-enabled

# | |-- *.load

# | `-- *.conf
```



```
# |-- conf-enabled

# | `-- *.conf

# `-- sites-enabled

# |-- 000-default.conf

# `-- *.conf
```

Основное, что нужно изменить:

- каталог для сайта **/var/www/html/bx-site**,
- порт, который слушает сервис (так как в качестве внешнего сервиса используется **nginx**)
- для сайта импортируйте настройки из виртуальной машины **000-default.conf**.

Примечание: В дефолтной конфигурации **/sites-enabled/000-default.conf** - это ссылка на файл в каталоге **/sites-available**.

Конфигурационные файлы для apache можно найти в папке [debian/apache2](#).

```
su -
```

```
rsync -av debian/apache2/ /etc/apache2/
```

Настройте следующие файлы:

- **ports.conf** - смена Listen на 8090
- **sites-available/000-default.conf** - настройки сайта

Отключите листинг каталогов в **Apache**:

```
a2dismod --force autoindex
```



Включите модуль **rewrite**:

```
a2enmod rewrite
```

Запустите сервис:

```
systemctl --now enable apache2
```

1.6. Конфигурация MariaDB

Что нужно добавить:

- **transaction-isolation** поменять в READ-COMMITTED.
- **innodb_flush_method** желательно должно быть равным O_DIRECT
- **innodb_flush_log_at_trx_commit** желательно должно быть равным 2.

Конфигурационные файлы для БД расположены в папке [debian/mysql](#).

```
su -
```

```
rsync -av debian/mysql/ /etc/mysql/
```

Измените следующие файлы:

- **my.cnf** - добавляем загрузку настроек из каталога **/etc/mysql/my-bx.d/**,
- **my-bx.d/zbx-custom.cnf** - сюда прописываем настройки, указанные выше.

Запустите сервис:

```
systemctl --now enable mariadb
```



systemctl restart mariadb

Настройте сервис через *mysql_secure_installation*:

```
mysql_secure_installation
```

```
...
```

```
Switch to unix_socket authentication [Y/n] n
```

```
... skipping.
```

```
Change the root password? [Y/n] y
```

```
New password:
```

```
Re-enter new password:
```

```
Password updated successfully!
```

```
Reloading privilege tables..
```

```
... Success!
```

```
Remove anonymous users? [Y/n] y
```

```
... Success!
```

```
Disallow root login remotely? [Y/n] y
```

```
... Success!
```

1.7. Конфигурация push-server

Схема работы:



```
-----  
-----  
  
| nginx: 0.0.0.0:80 | -> /bitrix/sub/bitrix/subws -> | node server.js --config push-server-sub-  
80XX.json |  
  
-----  
-----  
  
-----  
-----  
  
| nginx: 127.0.0.1:8895 | -> /bitrix/pub -> | node server.js --config push-server-pub-90XX.json |  
  
-----
```

Nginx проксирует запрос на push-сервис выбранного типа. Запросы получения сообщений (например, sub) - публичные и проксируются со стандартных портов 80/443, запросы публикации (pub) доступны только с внутреннего адреса сервера.

Nodejs-процессы делятся на два типа:

- процессы, отвечающие за подключение пользователя к выбранному каналу и получение им сообщений: слушают порты 8010-8015,
- процессы, отвечающие за отправку сообщения в канал: слушают порты 9010-9011.

Для запуска **push-server** понадобится:

- nodejs && npm,
- архив сервиса и его модулей.

Скачайте архив:

```
su -
```

```
cd /opt
```

```
wget https://repo.bitrix.info/vm/push-server-0.3.0.tgz
```



```
npm install --production ./push-server-0.3.0.tgz
```

Установка должна закончиться строкой:

```
added 1 package, and audited 145 packages in 13s
```

```
16 packages are looking for funding
```

```
run `npm fund` for details
```

Для удобства можно сделать так:

```
In -sf /opt/node_modules/push-server/etc/push-server /etc/push-server
```

Скопируйте файлы сервиса и основной конфигурационный сайт:

```
su -
```

```
cd /opt/node_modules/push-server
```

```
cp etc/init.d/push-server-multi /usr/local/bin/push-server-multi
```

```
mkdir /etc/sysconfig
```

```
cp etc/sysconfig/push-server-multi /etc/sysconfig/push-server-multi
```

```
cp etc/push-server/push-server.service /etc/systemd/system/
```

```
In -sf /opt/node_modules/push-server /opt/push-server
```

Отредактируйте конфигурационный файл `/etc/sysconfig/push-server-multi`.
Исправьте/добавьте параметры:



- SECURITY_KEY - секретный ключ для подписи соединения между клиентом и push-сервером,
- RUN_DIR - используется для хранения pid файлов процесса,
- USER/GROUP - пользователь, под которым будет запущен сервис,
- REDIS SOCK - сокет, который использует Redis сервис.

Пример:

```
GROUP=www-data
```

```
SECURITY_KEY="PUTTHEPRIVATEKEYHERE"
```

```
RUN_DIR=/tmp/push-server
```

```
REDIS_SOCK=/var/run/redis/redis.sock
```

Создайте пользователя:

```
useradd -g www-data bitrix
```

Каждый nodejs процесс будет запущен как отдельный процесс. Сгенерируйте конфигурационные файлы:

```
/usr/local/bin/push-server-multi configs pub
```

```
/usr/local/bin/push-server-multi configs sub
```

Создайте каталог через **tmpfiles.d**.

```
echo 'd /tmp/push-server 0770 bitrix www-data -' > /etc/tmpfiles.d/push-server.conf
```



```
systemd-tmpfiles --remove --create
```

Создайте каталог логов:

```
[[ ! -d /var/log/push-server ]] && mkdir /var/log/push-server
```

```
chown bitrix:www-data /var/log/push-serve
```

Изменяйте пользователя в конфигурационном файле сервиса **`/etc/systemd/system/push-server.service`**:

```
[Service]
```

```
User=bitrix
```

```
Group=www-data
```

```
ExecStart=/usr/local/bin/push-server-multi systemd_start
```

```
ExecStop=/usr/local/bin/push-server-multi sto
```

Переконфигурируйте файл:

```
systemctl daemon-reload
```

Запустите сервис

```
systemctl --now enable push-server
```

Перейдите в конфигурацию **push-модуля** (настройки сайта), включите использование локального push-сервера (последняя версия). Дополнительно нужно указать секретный ключ, который настраивали в файле `/etc/sysconfig/push-server-multi`.

1.8. Конфигурация портала

Создайте рабочий каталог:

```
mkdir /var/www/html/bx-site  
  
cd /var/www/html/bx-site  
  
wget https://www.1c-bitrix.ru/download/scripts/bitrixsetup.php  
  
chown www-data:www-data /var/www/html/bx-site -R
```

Аналогичным образом можно скачать нужный дистрибутив и установить его в каталог: `/var/www/html/bx-site`.

Создайте базу данных и пользователя:

```
create database portal;  
  
CREATE USER 'bitrix'@'localhost' IDENTIFIED BY 'PASSWORD';
```

```
GRANT ALL PRIVILEGES ON portal.* to 'bitrix'@'localhost';
```

Необходимо заменить PASSWORD на пароль, который будете использовать для доступа к БД.

Перейдите в http и откройте страницу: http://IP_ADDRESS/bitrixsetup.php. В качестве сервера БД используйте следующие настройки:



```
'host' => '127.0.0.1',  
  
'database' => 'portal',  
  
'login' => 'bitrix',
```

```
'password' => 'PASSWORD'
```

Пароль измените на тот, который вводили на этапе создания аккаунта

Для работы портала необходимо настроить **push-server**. Настройки могут быть выполнены через [административный раздел портала](#), а можно добавить их в конфигурационный файл. Покажем как это делается вторым способом.

Исправьте конфигурационный файл `/var/www/html/bx-site/bitrix/.settings.php`, добавив следующую секцию:

```
return array (  
  
'pull' => Array(  
  
'value' => array(  
  
'path_to_listener' => 'http://#DOMAIN#/bitrix/sub/',  
  
'path_to_listener_secure' => 'https://#DOMAIN#/bitrix/sub/',  
  
'path_to_modern_listener' => 'http://#DOMAIN#/bitrix/sub/',  
  
'path_to_modern_listener_secure' => 'https://#DOMAIN#/bitrix/sub/',  
  
'path_to_mobile_listener' => 'http://#DOMAIN#:8893/bitrix/sub/',  
  
'path_to_mobile_listener_secure' => 'https://#DOMAIN#:8894/bitrix/sub/',
```

```
'path_to_websocket' => 'ws://#DOMAIN#/bitrix/subws/',  
  
'path_to_websocket_secure' => 'wss://#DOMAIN#/bitrix/subws/',  
  
'path_to_publish' => 'http://localhost:8895/bitrix/pub/',  
  
'path_to_publish_web' => 'http://#DOMAIN#/bitrix/rest/',  
  
'path_to_publish_web_secure' => 'https://#DOMAIN#/bitrix/rest/',  
  
'nginx_version' => '4',  
  
'nginx_command_per_hit' => '100',  
  
'nginx' => 'Y',  
  
'nginx_headers' => 'N',  
  
'push' => 'Y',  
  
'websocket' => 'Y',  
  
'signature_key' => 'PUTTHEPRIVATEKEYHERE',  
  
'signature_algo' => 'sha1',  
  
'guest' => 'N',  
  
,  
  
,  
  
...
```

Обратите внимание: signature_key должен содержать тот же ключ, который был указан в /etc/sysconfig/push-server-multi в соответствующем ключе. Если все хорошо, то после перезапуска httpd:



```
systemctl restart httpd2
```

Вы увидите запросы к push-server:

```
Request URL: ws://sitename/bitrix/subws/?CHANNEL_ID=....
```

```
Request Method: GET
```

```
Status Code: 101 Switching Protocols
```

Настройка корпоративного портала

Создайте рабочий каталог:

```
mkdir /var/www/html/bx-site  
  
cd /var/www/html/bx-site  
  
wget https://www.1c-bitrix.ru/download/scripts/bitrixsetup.php  
  
chown www-data:www-data /var/www/html/bx-site -R
```

Аналогичным образом скачайте нужный дистрибутив и установите его в каталог: `/var/www/html/bx-site`.

Создайте базу данных и пользователя:

```
create database portal;  
  
CREATE USER 'bitrix'@'localhost' IDENTIFIED BY 'PASSWORD';  
  
GRANT ALL PRIVILEGES ON portal.* to 'bitrix'@'localhost';
```



Необходимо заменить PASSWORD на пароль, который будет использоваться для доступа к БД.

Push-server

Для работы портала необходимо настроить **push-server**. Настройки могут быть выполнены через административный раздел портала, а можно добавить их в конфигурационный файл. Покажем как это делается вторым способом.

Исправьте конфигурационный файл `/var/www/html/bx-site/bitrix/.settings.php`, добавив следующую секцию:

```
return array (  
  
    'pull' => Array(  
  
        'value' => array(  
  
            'path_to_listener' => 'http://#DOMAIN#/bitrix/sub/',  
  
            'path_to_listener_secure' => 'https://#DOMAIN#/bitrix/sub/',  
  
            'path_to_modern_listener' => 'http://#DOMAIN#/bitrix/sub/',  
  
            'path_to_modern_listener_secure' => 'https://#DOMAIN#/bitrix/sub/',  
  
            'path_to_mobile_listener' => 'http://#DOMAIN#:8893/bitrix/sub/',  
  
            'path_to_mobile_listener_secure' =>  
            'https://#DOMAIN#:8894/bitrix/sub/',  
  
            'path_to_websocket' => 'ws://#DOMAIN#/bitrix/subws/',  
  
            'path_to_websocket_secure' => 'wss://#DOMAIN#/bitrix/subws/',  
  
            'path_to_publish' => 'http://localhost:8895/bitrix/pub/',  
  
            'path_to_publish_web' => 'http://#DOMAIN#/bitrix/rest/',  
  
            'path_to_publish_web_secure' => 'https://#DOMAIN#/bitrix/rest/',  
  
            'nginx_version' => '4',  
  
            'nginx_command_per_hit' => '100',  
  

```

```
'nginx' => 'Y',  
  
'nginx_headers' => 'N',  
  
'push' => 'Y',  
  
'websocket' => 'Y',  
  
'signature_key' => 'PUTTHEPRIVATEKEYHERE',  
  
'signature_algo' => 'sha1',  
  
'guest' => 'N',  
  
,  
  
,  
  
...
```

Внимание! `signature_key` должен содержать тот же ключ, который указан в `/etc/sysconfig/push-server-multi` в соответствующем ключе. Если все хорошо, то после перезапуска службы `httpd`:

```
systemctl restart apache2
```

Вы увидите запросы к push-server:

```
Request URL: ws://sitename/bitrix/subws/?CHANNEL_ID=....
```

```
Request Method: GET
```

```
Status Code: 101 Switching Protocols
```

Установка экземпляра ПО

В папку `/var/www/html/bx-site/` необходимо загрузить архив `CD.zip`, который получает заказчик и распаковать его.

В браузере открыть страницу портала [https://\(Имя портала\)/install.php](https://(Имя портала)/install.php), после чего будет произведена установка нужных таблиц на портале.



Инструкция по установке экземпляра ПО
" Модульная система «Цифровой город»"

Битрикс24 
помогает бизнесу работать

После установки таблиц необходимо обновить страницу портала и все функции станут доступны.

Контактная информация

ООО «Цифровой город»

123022, Г.Москва, вн.тер. г. Муниципальный Округ Пресненский, ш Звенигородское,
дом 18/20, корпус 1, помещение II, комната 18,23,24

Почта: info@city-digital.ru

Телефон: +79183973179