



Содержание

| | |
|---|----|
| Об авторе | 10 |
| О рецензентах | 11 |
| Предисловие | 14 |
| Глава 1. Установка NGINX и сторонних модулей | 19 |
| Установка NGINX с помощью менеджера пакетов | 19 |
| CentOS | 20 |
| Debian | 21 |
| Сборка NGINX из исходного кода | 21 |
| Подготовка среды для сборки | 22 |
| Компиляция исходного кода | 22 |
| Настройка для работы в качестве веб-сервера или почтового сервера | 24 |
| Параметры configure для почтового прокси-сервера | 24 |
| Параметры configure для определения путей | 25 |
| Включение модулей | 26 |
| Отключение неиспользуемых модулей | 28 |
| Поиск и установка сторонних модулей | 30 |
| Полный пример | 31 |
| Резюме | 32 |
| Глава 2. Руководство по настройке | 33 |
| Основы формата конфигурационного файла | 33 |
| Глобальные конфигурационные параметры NGINX | 34 |
| Включаемые файлы | 35 |
| Секция с описанием HTTP-сервера | 36 |
| Клиентские директивы | 36 |

| | |
|---|-----------|
| Директивы, относящиеся к вводу-выводу | 38 |
| Директивы, относящиеся к хеш-таблицам | 39 |
| Директивы, относящиеся к сокетам | 40 |
| Пример конфигурации | 40 |
| Секция с описанием виртуального сервера | 41 |
| Местоположения – где, когда и как | 45 |
| Секция с описанием почтового сервера | 48 |
| Полный пример конфигурации | 49 |
| Резюме | 50 |
| Глава 3. Почтовый модуль | 51 |
| Простая служба проксирования | 51 |
| Служба POP3 | 53 |
| Служба IMAP | 54 |
| Служба SMTP | 55 |
| Использование SSL/TLS | 56 |
| Полный пример конфигурации почтового модуля | 58 |
| Служба аутентификации | 60 |
| Использование в связке с memcached | 67 |
| Интерпретация журналов | 70 |
| Ограничения операционной системы | 72 |
| Резюме | 73 |
| Глава 4. NGINX как обратный прокси-сервер | 75 |
| Введение в технологию обратного проксирования | 76 |
| Модуль проху | 77 |
| Унаследованные серверы с куками | 81 |
| Модуль upstream | 82 |
| Кэширование соединений | 83 |
| Алгоритмы балансировки нагрузки | 84 |
| Типы проксируемых серверов | 85 |
| Единственный проксируемый сервер | 85 |
| Несколько проксируемых серверов | 86 |
| Проксируемые серверы, работающие по протоколу, отличному от HTTP | 87 |
| Проксируемые серверы memcached | 88 |
| Проксируемые серверы FastCGI | 88 |
| Проксируемые серверы SCGI | 89 |
| Проксируемые серверы uWSGI | 89 |

| | |
|--|----|
| Преобразование конфигурации с «if» в более современную форму | 89 |
| Использование документов с описанием ошибок для обработки ошибок проксирования | 93 |
| Определение истинного IP-адреса клиента | 94 |
| Резюме | 95 |

Глава 5. Обратное проксирование, дополнительные вопросы..... 97

| | |
|---|-----|
| Безопасность за счет разделения | 98 |
| Шифрование трафика по протоколу SSL..... | 98 |
| Аутентификация клиентов по протоколу SSL..... | 100 |
| Блокирование трафика на основе IP-адреса отправителя | 103 |
| Обеспечение масштабируемости за счет изоляции компонентов приложения..... | 105 |
| Оптимизация производительности обратного прокси-сервера | 108 |
| Буферизация | 108 |
| Кэширование..... | 111 |
| Сохранение..... | 116 |
| Сжатие | 117 |
| Резюме | 120 |

Глава 6. NGINX как HTTP-сервер 121

| | |
|---|-----|
| Архитектура NGINX..... | 121 |
| Базовый модуль HTTP | 122 |
| Директива server | 123 |
| Протоколирование | 124 |
| Поиск файлов | 127 |
| Разрешение имен | 129 |
| Взаимодействие с клиентами | 131 |
| Установка предельных значений для предотвращения недобросовестного использования..... | 133 |
| Ограничение доступа | 136 |
| Потоковая передача мультимедийных файлов | 140 |
| Предопределенные переменные..... | 141 |
| Использование NGINX совместно с PHP-FPM | 143 |
| Пример конфигурации для Drupal..... | 147 |

| | |
|---|------------|
| Интеграция NGINX и uWSGI..... | 152 |
| Пример конфигурации для Django | 153 |
| Резюме | 155 |
| Глава 7. NGINX для разработчика | 156 |
| Интеграция с механизмом кэширования | 156 |
| Приложения без кэширования | 157 |
| Кэширование в базе данных | 158 |
| Кэширование в файловой системе | 161 |
| Динамическое изменение содержимого | 164 |
| Модуль addition..... | 164 |
| Модуль sub | 165 |
| Модуль xslt..... | 166 |
| Включение на стороне сервера | 167 |
| Принятие решений в NGINX..... | 170 |
| Создание безопасной ссылки | 173 |
| Генерация изображений | 174 |
| Отслеживание посетителей сайта | 178 |
| Предотвращение случайного выполнения кода | 179 |
| Резюме | 180 |
| Глава 8. Техника устранения неполадок..... | 181 |
| Анализ журналов | 181 |
| Форматы записей в журнале ошибок | 181 |
| Примеры записей в журнале ошибок | 183 |
| Настройка расширенного протоколирования | 186 |
| Отладочное протоколирование..... | 186 |
| Переключение двоичного файла во время выполнения..... | 186 |
| Использование журналов доступа для отладки | 193 |
| Типичные ошибки конфигурирования..... | 194 |
| Использование if вместо try_files..... | 195 |
| Использование if для ветвления по имени хоста | 196 |
| Неоптимальное использование контекста server | 196 |
| Ограничения операционной системы | 198 |
| Ограничение на количество файловых дескрипторов | 198 |
| Сетевые лимиты | 200 |
| Проблемы с производительностью | 201 |
| Использование модуля Stub Status..... | 203 |
| Резюме | 204 |

| | |
|--|-----|
| Приложение А. Справочник директив | 205 |
| Приложение В. Руководство по правилам переписывания | 254 |
| Введение в модуль rewrite | 254 |
| Создание новых правил переписывания..... | 259 |
| Преобразование правил из формата Apache | 261 |
| Рекомендация 1: заменить проверки существования каталогов и файлов директивой <code>try_files</code> | 261 |
| Рекомендация 2: заменить сравнение с <code>REQUEST_URI</code> секцией <code>location</code> | 262 |
| Рекомендация 3: заменить сравнение с <code>HTTP_HOST</code> секцией <code>server</code> | 263 |
| Рекомендация 4: заменить <code>RewriteCond</code> проверкой переменной в директиве <code>if</code> | 264 |
| Резюме | 265 |
| Приложение С. Сообщество NGINX | 266 |
| Список рассылки..... | 266 |
| IRC-канал | 266 |
| Веб-ресурсы | 267 |
| Как правильно составить отчет об ошибке..... | 267 |
| Резюме | 268 |
| Приложение D. Сохранение сетевых настроек в Solaris | 269 |
| Предметный указатель | 272 |



Об авторе

Димитрий Айвалиотис работает системным архитектором в компании, предоставляющей хостинг в Цюрихе, Швейцария. Начав карьеру с построения вычислительной сети на базе Linux для школы, он затем занимался созданием инфраструктуры высокодоступных двоярных центров обработки данных для банков и онлайн-порталов. Решая проблемы заказчиков – в течение десяти лет – он открыл для себя NGINX и с тех пор использует эту программу в качестве веб-сервера, прокси-сервера и для организации потоковой передачи мультимедийных данных.

Димитрий с отличием закончил бакалавриат физического факультета Политехнического института Ренсселера, а затем получил степень магистра по информационно-управляющим системам в Университете штата Флорида.

Это его первая книга.

Я благодарен Джону Блэкуэллу и Филу Марголису, прочитавшим ранние варианты рукописи. Их советы и критические замечания оказались очень полезны и позволили сделать книгу лучше. Хочу также поблагодарить технических рецензентов за конструктивную критику и указание на допущенные мной ошибки. Все оставшиеся ошибки – целиком моя вина.

Коллектив издательства Packt Publishing немало способствовал претворению этого проекта в жизнь. Их вера в меня как в писателя не давала мне впасть в отчаяние в тяжелые моменты, когда казалось, что все сроки будут сорваны.

Сотрудники компании NGINX, Inc. помогли заполнить пробелы в моем понимании внутренних механизмов работы NGINX. Без них я бы не смог написать эту книгу.

Отдельная благодарность семье. Мои жена и дети вынуждены были мириться с тем, что я тратил немало времени на сочинение книги. Я высоко ценю их терпение на протяжении этого непростого периода.



О рецензентах

Ясир Аднан (Yasir Adnan) живет в столице Бангладеш Дакке. Он изучает информатику и одновременно работает вольнонаемным программистом. Ему доводилось разрабатывать мобильные и веб-приложения, но в настоящее время он занимается в основном мобильными. С ним можно связаться по адресу yasiradnan@outlook.com.

Андрей Алексеев – соучредитель высокотехнологичной компании NGINX, Inc., стоящей за разработкой веб-сервера NGINX. До прихода в NGINX, Inc. в начале 2011 года Андрей работал в Интернет-индустрии в отделах информационно-коммуникационных технологий различных предприятий. Андрей получил диплом инженера-электроника в Санкт-Петербургском государственном электротехническом университете и окончил курс по программе MBA для руководителей в школе менеджмента Университета Антверпена.

Антонио П.П. Альмейда (Antonio P.P. Almeida) (@perusio) увлекся NGINX и высокопроизводительными веб-технологиями еще с тех пор, как пытался разрабатывать приложения для Drupal на мало-мощном ноутбуке на базе процессора Centrino частотой 1,3 ГГц. Из-за прозорливости Apache он был просто вынужден обратиться к NGINX. Он научился выжимать из NGINX максимум возможного в приложениях самых разных типов и в частности освоил все тонкости языка настройки NGINX. Антонио живет в Париже. Помимо NGINX, у него есть и другие пристрастия: малоизвестная музыка позднего итальянского средневековья, кино и желание сделать Drupal еще лучше.

Райнер Даффнер (Rainer Duffner) окончил Университет прикладных наук в Констанце, Германия, по специальности «информационные системы» и в настоящее время работает системным инженером в компании EveryWare AG, где помогает заказчикам извлечь максимум пользы из выделенных серверов на платформах FreeBSD, Linux



и Solaris. Он живет в небольшом городке близ Цюриха и в свободное время катается на горном велосипеде в окрестностях Цюриха и по швейцарским горам.

Я благодарен Димитрию за возможность принять участие в рецензировании этой замечательной книги. Ее ценность невозможно переоценить.

*Посвящаю своему отцу, который всегда говорил,
что я могу добиться любой цели, которую поставил перед собой.*



Предисловие

NGINX – это высокопроизводительный веб-сервер, потребляющий очень мало системных ресурсов. В Сети немало руководств по его настройке и примеров конфигураций. Задача этой книги – очистить мутные воды конфигурирования NGINX. По ходу дела вы научитесь настраивать NGINX для решения различных задач, узнаете, что означают некоторые покрытые мраком параметры, и поймете, как разработать конфигурацию, отвечающую вашим целям.

Вам больше не потребуется копировать фрагменты найденного где-то конфигурационного скрипта, потому что вы будете знать, как создать файл, делающий в точности то, что нужно. Это умение достигается не сразу, по пути встретится немало ухабов, но благодаря приведенным в этой книге советам вы сможете свободно писать конфигурационные файлы NGINX самостоятельно. А если что-то пойдет наперекосяк, то сумеете найти причину ошибки сами или по крайней мере попросить помощи без чувства вины за то, что не пытались найти ответ своими силами.

Эта книга построена по модульному принципу – так чтобы максимально облегчить поиск нужной информации. Все главы более-менее независимы. Можете сразу переходить к интересующему вас вопросу. Если складывается ощущение, что пропущено что-то важное, вернитесь назад и прочитайте предшествующие главы. Они организованы так, чтобы можно было строить конфигурационный файл постепенно.

О содержании книги

В главе 1 «Установка NGINX и сторонних модулей» объясняется, как установить NGINX в различных операционных системах и как затем добавить сторонние модули.

В главе 2 «Руководство по настройке» рассказывается о формате конфигурационного файла NGINX. Вы узнаете, для чего предназна-

чены различные контексты, как задавать глобальные параметры и что такое «местоположение».

Глава 3 «Почтовый модуль» посвящена модулю проксирования почты, здесь рассматриваются все аспекты его настройки. В качестве примера включена настройка службы аутентификации.

В главе 4 «NGINX как обратный прокси-сервер» вводится понятие обратного проксирования и описывается использование NGINX в этой роли.

В главе 5 «Обратное проксирование, дополнительные вопросы» более глубоко рассматривается использование NGINX в качестве обратного прокси-сервера для решения проблем масштабирования и производительности.

В главе 6 «NGINX как HTTP-сервер» описывается использование различных модулей, включенных в NGINX для решения типичных задач веб-сервера.

В главе 7 «NGINX для разработчика» показано, как интегрировать NGINX с приложением для ускорения доставки содержимого пользователям.

В главе 8 «Техника устранения неполадок» рассматриваются типичные ошибки при настройке и способы их отладки, а также даются рекомендации по оптимизации производительности.

Приложение А «Справочник директив» содержит удобный справочник по директивам настройки – как описанным в книге, так и ранее не упоминавшимся.

В приложении В «Руководство по правилам переписывания» описано, как работать с модулем переписывания URL в NGINX и приведено несколько простых шагов преобразования правил переписывания из формата Apache в формат NGINX.

Приложение С «Сообщество NGINX» включает перечень сетевых ресурсов, где можно найти дополнительные сведения.

В приложении D «Сохранение сетевых настроек в Solaris» подробно рассказано о том, что необходимо для сохранения изменений различных сетевых параметров в операционной системе Solaris версии 10 и старше.

Что необходимо для чтения этой книги

В каждой главе, где используются примеры кода, приведены инструкции по установке. По существу, требуется следующее.

- **Среда сборки:** компилятор, файлы-заголовки и еще кое-что по мелочи.

- ❑ **NGINX:** последняя версия должна подойти.
- ❑ **Ruby:** лучше всего взять дистрибутив с сайта <https://rvm.io>.
- ❑ **Perl:** стандартная версия годится.

На кого рассчитана эта книга

Книга рассчитана на опытных системных администраторов и системных инженеров, знающих, как производится установка и настройка серверов под конкретные нужды. Предварительное знакомство с NGINX не требуется.

Графические выделения

В этой книге используются различные шрифты для обозначения типа информации. Ниже приведено несколько примеров с пояснениями.

Фрагменты кода внутри абзаца выделяются следующим образом: «NGINX попытается собрать библиотеку статически, если при запуске скрипта `configure` указан параметр `--with-<library>=<path>`».

Кусок кода выглядит так:

```
$ export BUILD_DIR='pwd'
$ export NGINX_INSTALLDIR=/opt/nginx
$ export VAR_DIR=/home/www/tmp
$ export LUAJIT_LIB=/opt/luajit/lib
$ export LUAJIT_INC=/opt/luajit/include/luajit-2.0
```

Чтобы привлечь внимание к участку внутри куска кода, он выделяется полужирным шрифтом:

```
$ export BUILD_DIR='pwd'
$ export NGINX_INSTALLDIR=/opt/nginx
$ export VAR_DIR=/home/www/tmp
$ export LUAJIT_LIB=/opt/luajit/lib
$ export LUAJIT_INC=/opt/luajit/include/luajit-2.0
```

Входная и выходная информация командных утилит выглядит так:

```
$ mkdir $HOME/build
$ cd $HOME/build && tar xzf nginx-<version-number>.tar.gz
```

Новые термины и важные фрагменты выделяются полужирным шрифтом. Например, элементы графического интерфейса в меню

или диалоговых окнах выглядят в книге так: «Нажатие кнопки **Next** приводит к переходу на следующий экран».



Предупреждения и важные примечания выглядят так.



Советы и рекомендации выглядят так.

Отзывы

Мы всегда рады отзывам читателей. Расскажите нам, что вы думаете об этой книге – что вам понравилось или, быть может, не понравилось. Читательские отзывы важны для нас, так как помогают выпускать книги, из которых вы черпаете максимум полезного для себя.

Чтобы отправить обычный отзыв, просто пошлите письмо на адрес feedback@packtpub.com, указав название книги в качестве темы.

Если вы являетесь специалистом в некоторой области и хотели бы стать автором или соавтором книги, познакомьтесь с инструкциями для авторов по адресу www.packtpub.com/authors.

Поддержка клиентов

Счастливым обладателям книг Packt мы можем предложить ряд услуг, которые позволят извлечь из своего приобретения максимум пользы.

Загрузка кода примеров

Вы можете скачать код примеров ко всем книгам издательства Packt, купленным на сайте <http://www.PacktPub.com>. Если книга была куплена в другом месте, зайдите на страницу <http://www.PacktPub.com/support>, зарегистрируйтесь, и мы отправим файлы по электронной почте.

Опечатки

Мы проверяли содержимое книги со всем тщанием, но какие-то ошибки все же могли проскользнуть. Если вы найдете в нашей книге ошибку, в тексте или в коде, пожалуйста, сообщите нам о ней. Так

вы избавите других читателей от разочарования и поможете нам сделать следующие издания книги лучше. При обнаружении опечатки просьба зайти на страницу <http://www.packtpub.com/support>, выбрать книгу, щелкнуть по ссылке **errata submission form** и ввести информацию об опечатке. Проверив ваше сообщение, мы поместим информацию об опечатке на нашем сайте или добавим ее в список замеченных опечаток в разделе Errata для данной книги. Список подтвержденных опечаток можно просмотреть, выбрав название книги на странице <http://www.packtpub.com/support>.

Нарушение авторских прав

Незаконное размещение защищенного авторским правом материала в Интернете – проблема для всех носителей информации. В издательстве Packt мы относимся к защите прав интеллектуальной собственности и лицензированию очень серьезно. Если вы обнаружите незаконные копии наших изданий в любой форме в Интернете, пожалуйста, незамедлительно сообщите нам адрес или название веб-сайта, чтобы мы могли предпринять соответствующие меры.

Просим отправить ссылку на вызывающий подозрение в пиратстве материал по адресу copyright@packtpub.com.

Мы будем признательны за помощь в защите прав наших авторов и содействие в наших стараниях предоставлять читателям полезные сведения.

Вопросы

Если вас смущает что-то в этой книге, вы можете связаться с нами по адресу questions@packtpub.com, и мы сделаем все возможное для решения проблемы.



Глава 1. Установка NGINX и сторонних модулей

Первоначально NGINX задумывалась как HTTP-сервер. Она создавалась для решения проблемы C10K, описанной Дэниэлом Кегелем (Daniel Kege1) на странице <http://www.kege1.com/c10k.html>, – проектирование веб-сервера, способного обрабатывать одновременно 10 000 соединений. NGINX может это делать за счет основанного на событиях механизма обработки соединений и для достижения цели использует зависящий от ОС механизм событий.

Прежде чем переходить к настройке NGINX, ее необходимо установить. В этой главе описывается, как установить саму NGINX и где взять дополнительные модули. NGINX по природе своей является модульной программой, и существует обширное сообщество разработчиков сторонних модулей, расширяющих функциональность основного сервера. Эти модули прикомпилируются к серверу и устанавливаются вместе с ним.

В этой главе:

- Установка NGINX с помощью менеджера пакетов.
- Сборка NGINX из исходного кода.
- Настройка для работы в качестве веб-сервера или почтового сервера.
- Включение модулей.
- Поиск и установка сторонних модулей.
- Полный пример.

Установка NGINX с помощью менеджера пакетов

Для установки достаточно простой команды менеджера пакетов:

- Linux (дистрибутивы на основе Debian)Linux
`sudo apt-get install nginx`

- Linux (дистрибутивы на основе rpm)

```
sudo yum install nginx
```

- FreeBSD

```
sudo pkg_install -r nginx
```



Команда `sudo` применяется для получения привилегий суперпользователя ('root'). Если ваша операционная система поддерживает управление доступом на основе ролей (**RBAC – Role-based access control**), то для достижения той же цели следует использовать другую команду, например `pfexec`.

Описанные команды устанавливают NGINX в стандартные места, зависящие от операционной системы. Это предпочтительный метод установки в случае, когда использование пакетов ОС – непреложное требование.

Команда, разрабатывающая ядро NGINX, предлагает также двоичные файлы стабильной версии на странице <http://nginx.org/en/download.html>. Если в дистрибутиве отсутствует пакет `nginx` (как, например, в CentOS), то можно установить заранее откомпилированные и протестированные двоичные файлы, как описано ниже.

CentOS

Добавьте репозиторий NGINX в конфигурацию `yum`, создав следующий файл:

```
sudo vi /etc/yum.repos.d/nginx.repo
```

```
[nginx]
name=nginx repo
baseurl=http://nginx.org/packages/centos/6/$basearch/
gpgcheck=0
enabled=1
```

Затем выполните следующую команду для установки `nginx`:

```
sudo yum install nginx
```

На странице с указанным выше URL имеются также инструкции по установке пакета `nginx-release`.

Debian

Скачайте ключ подписания NGINX – http://nginx.org/keys/nginx_signing.key – и добавьте его в связку ключей apt:

```
sudo apt-key add nginx_signing.key
```

Добавьте репозиторий `nginx.org` в конец файла `/etc/apt/sources.list`:

```
vi /etc/apt/sources.list
deb http://nginx.org/packages/debian/ squeeze nginx
deb-src http://nginx.org/packages/debian/ squeeze nginx
```

Установите `nginx`, выполнив такие команды:

```
sudo apt-get update
sudo apt-get install nginx
```

Если среди пакетов, прилагаемых к операционной системе, нет `nginx` или версия, включенная в пакет, устарела и не умеет делать того, что вам нужно, или пакеты, размещенные на сайте `nginx.org`, не отвечают вашим потребностям или вы хотите использовать версию NGINX для разработчиков, то остается единственная возможность – собрать NGINX из исходного кода.

Сборка NGINX из исходного кода

Существуют две ветви кода NGINX – стабильная и разрабатываемая. Именно в последней находятся все новые функции, которые интегрируются и тестируются перед включением в стабильную версию. Выпуск разрабатываемой версии сопровождается таким же контролем качества и прогоном функциональных тестов, как и выпуск стабильной версии, поэтому ту и другую можно использовать в производственных системах. Основное различие между ними связано с поддержкой сторонних модулей. В разрабатываемой версии внутренний API может измениться, тогда как в стабильной он фиксирован, поэтому обратная совместимость сторонних модулей гарантируется только для стабильных версий.

Подготовка среды для сборки

Для сборки NGINX из исходного кода система должна отвечать определенным требованиям. Кроме компилятора, понадобятся библиотеки OpenSSL и **PCRE (Perl Compatible Regular Expressions)** и файлы-заголовки для них – если вы хотите иметь поддержку для SSL и модуля переписывания URL соответственно. В некоторых дистрибутивах эти требования удовлетворены изначально. Если же это не так, то вам предстоит найти и установить подходящий пакет либо скачать исходный код, распаковать его в какой-то каталог и сообщить об этом каталоге скрипту конфигурирования NGINX.

NGINX попытается собрать библиотеку статически, если при запуске скрипта `configure` указан параметр `--with-<library>=<path>`. Это бывает полезно, если вы хотите, чтобы NGINX не зависела от того, что установлено в системе, или требуется добиться от двоичного файла `nginx` максимальной производительности. Если используются функции внешних библиотек, доступные лишь начиная с некоторой версии (например, расширение TLS Next Protocol Negotiation – согласование следующего протокола, – появившееся в версии OpenSSL 1.0.1), то указывайте путь к распакованному исходному коду именно этой версии.

Существуют и другие необязательные пакеты для поддержки той или иной функциональности, в том числе алгоритмов хеширования MD5 и SHA-1, библиотеки сжатия `zlib` и библиотеки `libatomic`. Алгоритмы хеширования применяются во многих местах NGINX, в частности, для вычисления хеш-кода URI, играющего роль ключа кэша. Библиотека `zlib` используется для сжатия отправляемого клиенту содержимого. Если доступна библиотека `atomic_ops`, то NGINX будет использовать атомарные операции обновления памяти для реализации быстрого алгоритма блокировки памяти.

Компиляция исходного кода

NGINX можно скачать со страницы <http://nginx.org/en/download.html>. Там вы найдете исходный код обеих ветвей в форматах `.tar.gz` и `.zip`. Распакуйте архив во временный каталог:

```
$ mkdir $HOME/build
$ cd $HOME/build && tar xzf nginx-<version-number>.tar.gz
```

Произведите конфигурирование, выполнив команду:

```
$ cd $HOME/build/nginx-<version-number> && ./configure
```

После чего соберите программу:

```
$ make && sudo make install
```

При самостоятельной сборке двоичного файла `nginx` вы можете включить только то, что вам нужно, например, указать, от имени какого пользователя должна работать NGINX, или задать подразумеваемые по умолчанию места расположения журналов, чтобы их не нужно было явно прописывать в конфигурационном файле. В таблице ниже приведены параметры `configure`, которые позволяют изменить способ сборки двоичного файла. Эти параметры не зависят от того, какие модули NGINX подключаются.

Общие параметры `configure`

| Параметр | Описание |
|--|--|
| <code>--prefix=<path></code> | Корень дерева установки. Все остальные пути указываются относительно этого корня |
| <code>--sbin-path=<path></code> | Путь к двоичному файлу <code>nginx</code> . Если не задан, то файл создается в каталоге <code>prefix</code> |
| <code>--conf-path=<path></code> | Путь к каталогу, в котором <code>nginx</code> ищет свой конфигурационный файл, если тот не указан в командной строке |
| <code>--error-log-path=<path></code> | Путь к каталогу, в который <code>nginx</code> записывает журнал ошибок, если не задан иным способом |
| <code>--pid-path=<path></code> | Путь к каталогу, в котором создается <code>pid</code> -файл главного процесса, обычно <code>/var/run</code> |
| <code>--lock-path=<path></code> | Путь к файлу, управляющему взаимоблокировкой разделяемой памяти |
| <code>--user=<user></code> | Пользователь, от имени которого запускаются рабочие процессы |
| <code>--group=<group></code> | Пользователь, от имени которой запускаются рабочие процессы |
| <code>--with-file-aio</code> | Включает асинхронный ввод-вывод для FreeBSD 4.3+ и Linux 2.6.22+ |
| <code>--with-debug</code> | Включает отладочное протоколирование. Не рекомендуется для производственных систем |

Можно также задать оптимизации, отсутствующие в готовом пакете. Ниже перечислены некоторые особенно полезные параметры.