CentOS 7 安装Nginx

作者：justdoit
原文链接：https://ld246.com/article/1470211300416
来源网站：链滴

<h2 id="toc_h2_0">一、准备工作：</h2>
<h3 id="toc_h3_1">1、安装必备工具：</h3>

<pre class="brush:shell; toolbar: true; auto-links: false; hljs elixir"><code class="hljs elixir"><
pan class="hljs-variable">$ yum -y install gcc gcc-c++ autoconf automake
</code> <code class="hljs elixir">$ yum -y install zl
b zlib-devel openssl openssl-devel pcre-devel</code> </pre>
说明：
pcre: 用来作地址重写的功能。
zlib：nginx 的gzip模块,传输数据打包，省流量（但消耗资源）。
openssl：提供ssl加密协议。
<h3 id="toc_h3_2">2、新建一个系统级用户组和匿名用户，以及下面编译时使用</h3>
<pre class="brush:shell; toolbar: true; auto-links: false; hljs elixir"><code class="hljs elixir"><
pan class="hljs-variable">$ sudo groupadd -r nginx
</code> <code class="hljs elixir">$ sudo useradd -s
/sbin/nologin -g nginx -r nginx-user</code> </pre>
<h2 id="toc_h2_3">二、Nginx编译安装：</h2>
<h3 id="toc_h3_4">1、下载Nginx：<a href="https://ld246.com/forward?goto=http%3A%2F
2Fnginx.org%2Fen%2Fdownload.html" rel="nofollow ugc" target="_blank">http://nginx.org/
n/download.html</h3>
<h3 id="toc_h3_5">2、解压编译：</h3>
<pre class="brush:shell; toolbar: true; auto-links: false; hljs haml"><code class="hljs haml">$
ar -zxvf nginx-1.11.3.tar.gz
$ cd nginx-1.11.3/
</code> <code class="hljs haml">$ ./configure
--prefix=<span class="hljs-regex
">/etc/nginx--sbin-path=<span cl
ss="hljs-regexp">/usr/sbin<sp
n class="hljs-regexp">/nginx
-<span class="
ljs-regexp">-conf-path=/etc<s
an class="hljs-regexp">/nginx/nginx.conf
--error-log-path=<spa
>/var/log<
pan>/nginx/error.log
--http-log-path=<span
/var/log<sp
n>/nginx/access.log
--pid-path=<sp
n class="hljs-regexp">/var/run
span class="hljs-regexp">/nginx.pid
-<span class="
ljs-regexp">-lock-path=/var<s
an class="hljs-regexp">/run/nginx.lock
--http-client-body-temp-path=<span class="hljs-r
gexp">/var/cache<span class=
hljs-regexp">/nginx/client<spa
>_temp
--http-proxy-temp-path=<span class="hljs-regexp
>/var/cache<span class="hljs-r
gexp">/nginx/proxy_te
p
--http-fastcgi-temp-path=<span class="hljs-regexp
>/var/cache<span class="hljs-r

gexp">/nginx/fastcgi_tmp
--http-uwsgi-temp-path=/var/cache/nginx/uwsgi_tmp
--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-mail
--with-mail_ssl_module
--with-file-aio
--with-ipv6
--with-http_spdy_module
--with-cc-opt='-O2 -g -pipe -Wp,-D_FORTIFY_SOURCE=2 -fexceptions -fstack-protector --param=ssp-buffer-size=4 -m64 -mtune=generic'</code></></pre>
上面的参数的作用可以通过--help来查看下文附加1中有提供
<pre class="brush:shell; toolbar: true; auto-links: false; hljs lua"><code class="hljs lua">$ ./configure --help</code></pre>
编译日志：
<pre class="brush:shell; toolbar: true; auto-links: false; hljs"><code class="hljs r">checking for OS
 + Linux 3.10.0-229.4.2.el7.x86_64 x86_64
checking for C compiler ... found
 + using GNU C compiler
 + gcc version: 4.8.3 20140911 (Red Hat 4.8.3-9) (GCC)
checking for gcc -pipe switch ... found
checking for gcc builtin atomic operations ... found

checking for C99 variadic macros ... found
checking for gcc variadic macros ... found
checking for unistd.h ... found
checking for inttypes.h ... found
checking for limits.h ... found
checking for sys/filio.h ... not found
checking for sys/param.h ... found
checking for sys/mount.h ... found
checking for sys/statvfs.h ... found
checking for crypt.h ... found
checking for Linux specific features
checking for epoll ... found
checking for EPOLLRDHUP ... found
checking for O_PATH ... found
checking for sendfile() ... found
checking for sendfile64() ... found
checking for sys/prctl.h ... found
checking for prctl(PR_SET_DUMPABLE) ... found
checking for sched_setaffinity() ... found
checking for crypt_r() ... found
checking for sys/vfs.h ... found
checking for poll() ... found
checking for /dev/poll ... not found
checking for kqueue ... not found
checking for crypt() ... not found
checking for crypt() in libcrypt ... found
checking for F_READAHEAD ... not found
checking for posix_fadvise()

>... found
checking for O_DIRECT ...
/span> found
checking for F_NOCACHE .
. not found
checking for directio() ...<
span> not found
checking for statfs() ...</s
an> found
checking for statvfs() ...</
pan> found
checking for dlopen() ...</
pan> not found
checking for dlopen() in</
pan> libdl ... found
checking for sched_yield()
.. found
checking for SO_SETFIB ...
/span> not found
checking for SO_REUSEPORT <span class="hljs-keywor
">... found
checking for SO_ACCEPTFILTER <span class="hljs-keyw
rd">... not found
checking for TCP_DEFER_ACCEPT <span class="hljs-ke
word">... found
checking for TCP_KEEPIDLE <span class="hljs-keyword
>... found
checking for TCP_FASTOPEN <span class="hljs-keywor
">... not found
checking for TCP_INFO ...<
span> found
checking for accept4() ...<
span> found
checking for eventfd() ...<
span> found
checking for int size ...</s
an> 4 bytes
checking for long size ...</
pan> 8 bytes
checking for long long size <span class="hljs-keyword
>... 8 bytes
checking for void * size ...
/span> 8 bytes
checking for uint64_t ...</
pan> found
checking for sig_atomic_t ..
 found
checking for sig_atomic_t size <span class="hljs-keywo
d">... 4 bytes
checking for socklen_t ...<
span> found
checking for in_addr_t ...<
span> found
checking for in_port_t ...</

pan> found
checking for rlim_t ...</spn> found
checking for uintptr_t ...</pan> uintptr_t found
checking for system byte ordering ... little endian
checking for size_t size ... 8 bytes
checking for off_t size ...</pan> 8 bytes
checking for time_t size .../span> 8 bytes
checking for setproctitle() .. not found
checking for pread() ...</san> found
checking for pwrite() ...</pan> found
checking for sys_nerr ...</pan> found
checking for localtime_r() .. found
checking for posix_memalign() ... found
checking for memalign() .. found
checking for mmap(MAP_ANON|MAP_SHARED) ... found
checking for mmap("/dev/zeo", MAP_SHARED) ... found
checking for System V shared memory ... found
checking for POSIX semaphores ... not found
checking for POSIX semaphores in libpthread ... found
checking for struct msghdr.msg_control ... found
checking for ioctl(FIONBIO) <span class="hljs-keyword>... found
checking for struct tm.tm_gmtoff ... found
checking for struct dirent.d_namlen ... not found
checking for struct dirent.d_type ... found
checking for sysconf(_SC_NPROCESSORS_ONLN) ... found
checking for openat(), fstatat() ... found
checking for getaddrinfo() .. found
checking for PCRE library

/span> ... found
checking for PCRE JIT support <span class="hljs-keywo
d">... found
checking for OpenSSL libr
ry ... found
checking for zlib library</
pan> ... found
creating objs/Makefile
Configuration summary

using system PCRE library
using system OpenSSL library
md5: using OpenSSL library
sha1: using OpenSSL library
using system zlib library

</code> <code class="hljs r">nginx path prefix: "/usr"</span

nginx binary file: "/usr/sbin/nginx"
nginx configuration prefix: "/etc/nginx"
nginx configuration file: "/etc/nginx/nginx.conf"
nginx pid file: "/var/run/nginx/nginx.pid"
nginx error log file: "/var/log/nginx/error.log"
nginx http access log file: "/var/log/nginx/http.log"
nginx http client request body temporary files: "/var/tmp/nginx/cli
nt"
nginx http proxy temporary files: "/var/tmp/nginx/proxy"
br>
nginx http fastcgi temporary files: "/var/tmp/nginx/fcgi"<
r>
nginx http uwsgi temporary files: "uwsgi_temp"
nginx http scgi temporary files: "scgi_temp"</code>
/pre>
好像很成功。
<h3 id="toc_h3_6">3、安装:</h3>

<pre class="brush:shell; toolbar: true; auto-links: false; hljs go"><code class="hljs go">$ <sp
n class="hljs-built_in">make && make</span
 install</code></pre>
<h3>4、启动：</h3>

<pre class="brush:shell; toolbar: true; auto-links: false; hljs elixir"><code class="hljs elixir"><
pan class="hljs-variable">$ nginx -c /etc/nginx/nginx.conf</code></pre>

如果提示缺什么包，直接用yum安装
<pre class="brush:shell; toolbar: true; auto-links: false; hljs elixir"><code class="hljs elixir"><
pan class="hljs-variable">$ yum install xxxxx</code></pre>
如果提示:
<blockquote>nginx: [emerg] mkdir() "/var/tmp/nginx/client_temp" failed (2: No such file or di
ectory)</blockquote>
则用管理员去mkdir 创建文件夹后再启动