HTTPS和HTTP使用同一个端口

原理

NGINX 1.15.2版本中新增了一个关键功能， `stream_ssl_preread` 模块允许在协议握手阶段，从消息中提取协议类型或域名信息，根据不同的协议或域名进行转发。

在使用TCP(stream)代理转发流量时,可以使用 `ssl_preread_protocol` 变量区分SSL/TLS和其他协议。

`ssl_preread_protocol` 变量从消息字段中提取SSL/TLS 版本号。如果不是 SSL 或 TLS 连接，则变量将为空，表示连接使用的是 SSL/TLS 以外的协议。

`ssl_preread_protocol` 变量值：

- TLSv1
- TLSv1.1
- TLSv1.2
- TLSv1.3
- ""　非SSL/TLS 协议

配置示例

```
stream {
    upstream web {
        server 192.168.56.114:8080;
    }

    upstream https {
        server 192.168.56.114:8443;
    }

    log_format basic 'ssl_version: $ssl_preread_protocol | upstream: $upstream';
    access_log /var/log/nginx/nginx-access.log basic ;

    map $ssl_preread_protocol $upstream {
        "" web;
        "TLSv1.3" https;
        default https;
    }

     # HTTPS and HTTP on the same port
    server {
        listen 80;

        proxy_pass $upstream;
        ssl_preread on;
    }
}
```

```
server {
    listen       8080;
    listen       8443  ssl;
    server_name  localhost;

    ssl_certificate     /home/ssl/server.crt;
    ssl_certificate_key /home/ssl/server.key;
    ssl_protocols       TLSv1 TLSv1.1 TLSv1.2 TLSv1.3;
    ssl_ciphers         HIGH:!aNULL:!MD5;
    ssl_password_file   /home/ssl/cert.pass;

    location / {
        root   /usr/share/nginx/html;
        index  index.html index.htm;
    }

    error_page   500 502 503 504  /50x.html;
    location = /50x.html {
        root   /usr/share/nginx/html;
    }

}
```

如果要通过（例如）在同一端口上运行 SSL/TLS 和 其他TCP服务(例如SSH或数据库)来避免防火墙限制，这将非常有用。

除了 `ssl_preread_protocol` 变量，还支持以下变量：

- `ssl_preread_server_name` 获取请求的服务器名称
- `ssl_preread_alpn_protocols` 获取ALPN 协议列表,这些值用逗号分隔(例如 `h2,http/1.1`)

添加模块

nginx默认不包含 `stream_ssl_preread` 模块，我们需要自动从源码进行编译。

```
#查看nginx详细信息
nginx -V
```

```
nginx version: nginx/1.22.1
built by gcc 4.8.5 20150623 (Red Hat 4.8.5-44) (GCC)
built with OpenSSL 1.0.2k-fips  26 Jan 2017
TLS SNI support enabled
configure arguments: --prefix=/etc/nginx --sbin-path=/usr/sbin/nginx --modules-path=/usr/lib64/nginx/module
s --conf-path=/etc/nginx/nginx.conf --error-log-path=/var/log/nginx/error.log --http-log-path=/var/log/ngin
x/access.log --pid-path=/var/run/nginx.pid --lock-path=/var/run/nginx.lock --http-client-body-temp-path=/va
r/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/cach
e/nginx/scgi_temp --user=nginx --group=nginx --with-compat --with-file-aio --with-threads --with-http_addit
ion_module --with-http_auth_request_module --with-http_dav_module --with-http_flv_module --with-http_gunzip
_module --with-http_gzip_static_module --with-http_mp4_module --with-http_random_index_module --with-http_r
ealip_module --with-http_secure_link_module --with-http_slice_module --with-http_ssl_module --with-http_stu
b_status_module --with-http_sub_module --with-http_v2_module --with-mail --with-mail_ssl_module --with-stre
am --with-stream_realip_module --with-stream_ssl_module --with-stream_ssl_preread_module --with-cc-opt='-O2
 -g -pipe -Wall -Wp,-D_FORTIFY_SOURCE=2 -fexceptions -fstack-protector-strong --param=ssp-buffer-size=4 -gr
ecord-gcc-switches -m64 -mtune=generic -fPIC' --with-ld-opt='-Wl,-z,relro -Wl,-z,now -pie'
```

版本

配置和模块信息

下载对应版本的源码：http://nginx.org/download/nginx-1.22.1.tar.gz

安装依赖

```
yum install -y pcre pcre-devel openssl openssl-devel \
               zlib zlib-devel gcc gcc-c++
```

```
tar -zxvf nginx-1.22.1.tar.gz
cd nginx-1.22.1

# 在原有的配置参数上加入 --with-stream_ssl_preread_module
./configure --with-stream_ssl_preread_module \
            --prefix=/etc/nginx --sbin-path=/usr/sbin/nginx --modules-path=/usr/lib64/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-compat --with-file-aio --with-threads --with-http_addition_module --with-http_auth_request_module --with-http_dav_module --with-http_flv_module --with-http_gunzip_module --with-http_gzip_static_module --with-http_mp4_module --with-http_random_index_module --with-http_realip_module --with-http_secure_link_module --with-http_slice_module --with-http_ssl_module --with-http_stub_status_module --with-http_sub_module --with-http_v2_module --with-mail --with-mail_ssl_module --with-stream --with-stream_realip_module --with-stream_ssl_module --with-stream_ssl_preread_module --with-cc-opt='-O2 -g -pipe -Wall -Wp,-D_FORTIFY_SOURCE=2 -fexceptions -fstack-protector-strong --param=ssp-buffer-size=4 -grecord-gcc-switches -m64 -mtune=generic -fPIC' --with-ld-opt='-Wl,-z,relro -Wl,-z,now -pie'
```

编译好的nginx在objs目录下，运行 `objs/nginx -V` ，查看是否包含新模块

```
nginx version: nginx/1.22.1
built by gcc 4.8.5 20150623 (Red Hat 4.8.5-44) (GCC)
built with OpenSSL 1.0.2k-fips  26 Jan 2017
TLS SNI support enabled
configure arguments: --with-stream_ssl_preread_module --prefix=/etc/nginx --sbin-path=/usr/sbin/nginx --mod
ules-path=/usr/lib64/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_te
mp --http-fastcgi-temp-path=/var/cache/nginx/fastcgi_temp --http-uwsgi-temp-path=/var/cache/nginx/uwsgi_tem
p --http-scgi-temp-path=/var/cache/nginx/scgi_temp --user=nginx --group=nginx --with-compat --with-file-aio
 --with-threads --with-http_addition_module --with-http_auth_request_module --with-http_dav_module --with-h
ttp_flv_module --with-http_gunzip_module --with-http_gzip_static_module --with-http_mp4_module --with-http_
random_index_module --with-http_realip_module --with-http_secure_link_module --with-http_slice_module --wit
h-http_ssl_module --with-http_stub_status_module --with-http_sub_module --with-http_v2_module --with-mail -
-with-mail_ssl_module --with-stream --with-stream_realip_module --with-stream_ssl_module --with-stream_ssl_
preread_module --with-cc-opt='-O2 -g -pipe -Wall -Wp,-D_FORTIFY_SOURCE=2 -fexceptions -fstack-protector-str
ong --param=ssp-buffer-size=4 -grecord-gcc-switches -m64 -mtune=generic -fPIC' --with-ld-opt='-Wl,-z,relro
-Wl,-z,now -pie'
```

安装nginx (会覆盖原有的nginx，请提前做好备份)

```
make && make install
```

重启nginx

```
systemctl restart nginx
```

测试

开启防火墙，只放行80端口

```
systemctl start firewalld
# 放行80端口
firewall-cmd --zone=public --permanent --add-service=http
firewall-cmd --reload
```

参考文档：

https://v2ex.com/t/894781

http://nginx.org/en/docs/stream/ngx_stream_ssl_preread_module.html

https://www.nginx.com/blog/running-non-ssl-protocols-over-ssl-port-nginx-1-15-2/