



链滴

http1.1/http2/http3 说明以及 nginx 实现 http2

作者: [yuanhenglizhen](#)

原文链接: <https://ld246.com/article/1615442365996>

来源网站: 链滴

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

概述

摘自百度百科:

超文本传输协议 (HTTP, HyperText Transfer Protocol)是互联网上应用最为广泛的一种网络协议。有的WWW文件都必须遵守这个标准。设计HTTP最初的目的是为了提供一种发布和接收HTML页面方法。

HTTP属于OSI网络七层协议模型中的"最上层":应用层。由请求和响应构成,是一个标准的客户端服务模式。HTTP是一个无状态的协议。

HTTP默认端口号为80。它也可以承载在TLS和SSL之上,通过加密、认证的方式实现数据传输的安全称为HTTPS,HTTPS默认端口号为443。

早期HTTP用于传输网页HTML文件,发展到现在,应用变得广泛,客户端软件(PC, Android,iOS)大部分通过HTTP传输数据。

HTTP1.1

发布于1997年

持久连接 (persistent connection),即TCP连接默认不关闭,可以被多个请求复用,不用声明 **Connection: keep-alive**。解决了1.0版本的keepalive问题,1.1版本加入了持久连接,一个TCP连接可以许多个HTTP请求;

加入了管道机制,在同一个TCP连接里,允许多个请求同时发送,增加了并发性,进一步改善了HTTP协议的效率,虽然允许复用TCP连接,但是同一个TCP连接里面,所有的数据通信是按次序进行的。;

新增了请求方式PUT、PATCH、OPTIONS、DELETE等;

HTTP2.0

发布于2015年

为了解决1.1版本利用率不高的问题,提出了HTTP/2.0版本。增加双工模式,即不仅客户端能够同时送多个请求,服务端也能同时处理多个请求,解决了队头堵塞的问题(HTTP2.0使用了多路复用的技术,做到同一个连接并发处理多个请求,而且并发请求的数量比HTTP1.1大了好几个数量级);HTTP请求和响应中,状态行和请求/响应头都是些信息字段,并没有真正的数据,因此在2.0版本中将所有的信息字段建立一张表,为表中的每个字段建立索引,客户端和服务端共同使用这个表,他们之间就以索号来表示信息字段,这样就避免了1.0旧版本的重复繁琐的字段,并以压缩的方式传输,提高利用率。

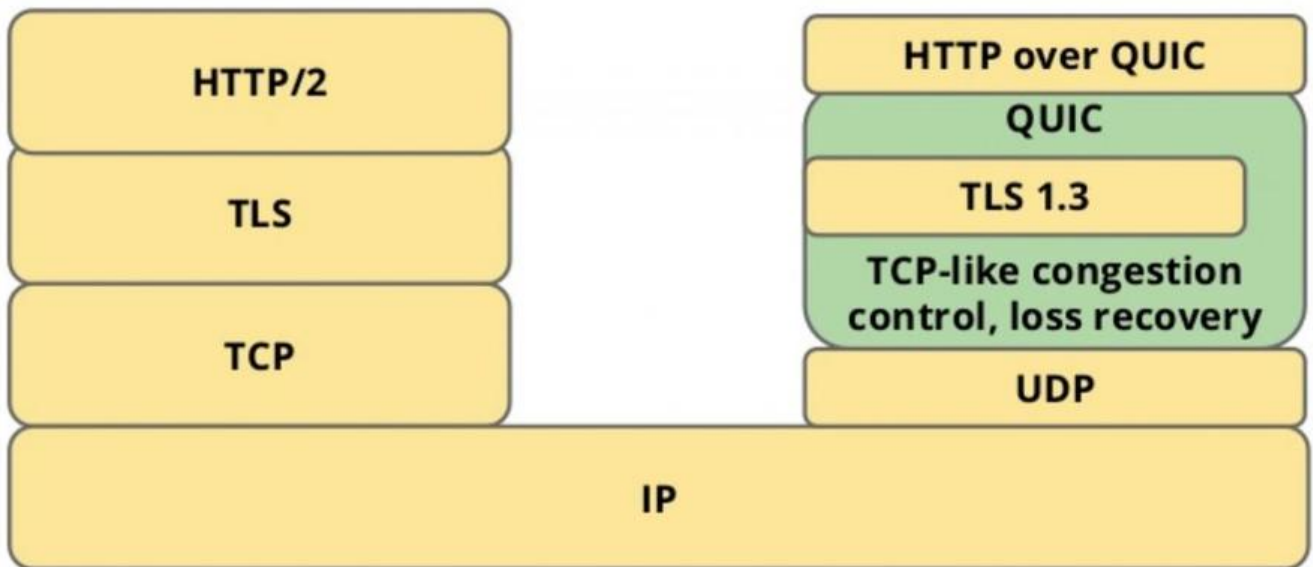
另外也增加服务器推送的功能,即不经请求服务端主动向客户端发送数据。

HTTP3.0

HTTP3的主要改进在传输层上。传输层不会再有那些繁重的TCP连接了。现在,一切都会走UDP。QUIC的意思是“快速UDP Internet连接”。协议的这种更改将显著加快连接建立和数据传输的速度。而,虽说UDP肯定更快、更简单,但它不具备TCP的可靠性和错误处理能力。

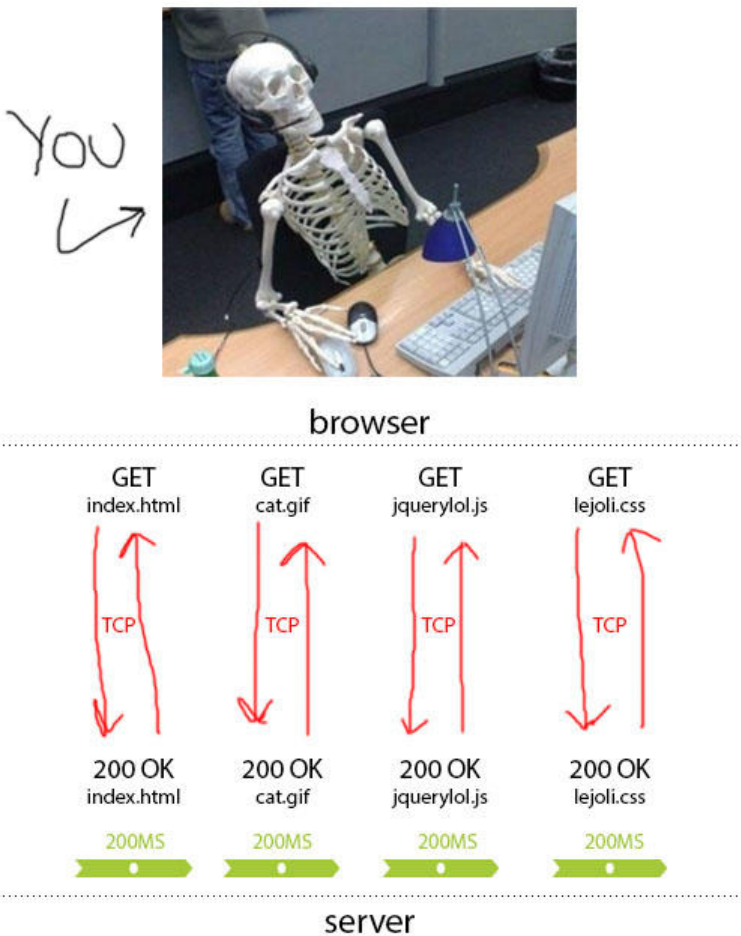
HTTP3是HTTP2的复用和压缩,协议从TCP更改为UDP。然后,谷歌的那些人在协议中添加了他们做的层,以确保稳定性、数据包接收顺序及安全性。

因此，HTTP3 在保持 QUIC 稳定性的同时使用 UDP 来实现高速度，同时又不会牺牲 TLS 的安全性。是的，在 QUIC 中就有 TLS1.3，你可以用它发起优雅的 SSL。这些层的底层机制是下面这样：

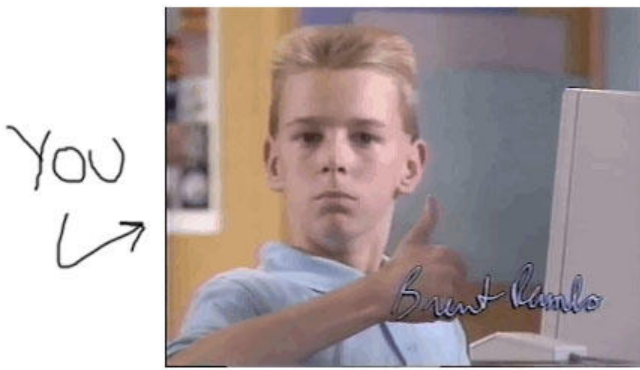


通信过程

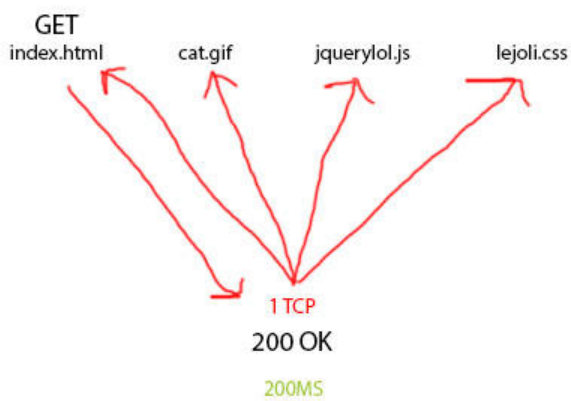
HTTP



HTTP2

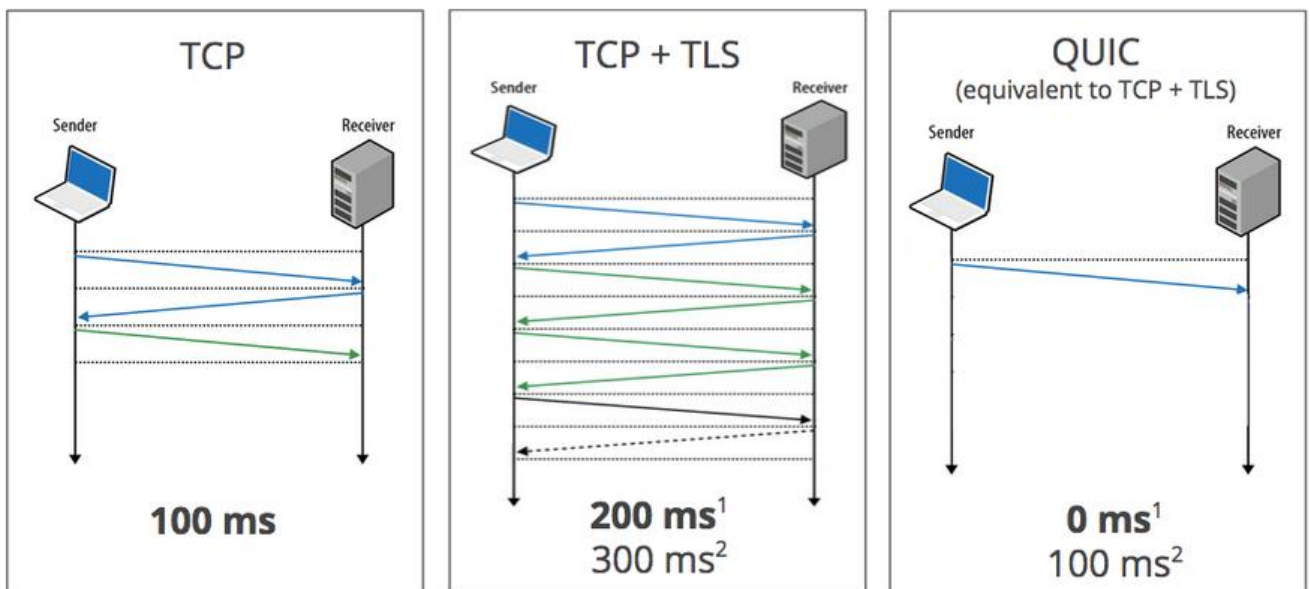


browser



server

Zero RTT Connection Establishment



1. Repeat connection
2. Never talked to server before

Nginx实现http2.0

检查nginx支持的模块

```
/usr/sbin/nginx -V
```

如果包含 `--with-http_v2_module` 这个模块，恭喜你可以直接使用

```
server {  
    listen 443 ssl http2;  
    ...  
}
```

使用方式很简单在配置文件server模块里面的listen那边加个http2即可

可以通过chrome的devtools来检查修改是否生效

http1.1:

Name	Status	Protocol	Type	Initiator	Size	Time	Waterfall
index.css	200	http/1.1	styles...	(index)	341 B	195 ms	
swiper.min.css	200	http/1.1	styles...	(index)	3.9 kB	190 ms	
wap.css	200	http/1.1	styles...	(index)	3.4 kB	364 ms	

http2.0:

font.css	200	h2	styles...	index.html	1.0 kB	160 ms	
com.css	200	h2	styles...	index.html	1.2 kB	162 ms	
main.css	200	h2	styles...	index.html	4.8 kB	163 ms	
index.css	200	h2	styles...	index.html	191 B	159 ms	
swiper.min.css	200	h2	styles...	index.html	3.7 kB	163 ms	
wap.css	200	h2	styles...	index.html	3.3 kB	162 ms	
layui.css	200	h2	styles...	index.html	16.8 kB	240 ms	
jquery-3.4.1.min.js	200	h2	script	index.html	34.8 kB	348 ms	

Nginx实现http3.0

编译安装支持quic的nginx，可以参看《[Nginx 配置启用 QUIC 和 HTTP/3.0](#)》

```
# Enable QUIC and HTTP/3.  
listen 443 quic reuseport;
```

```
# Enable HTTP/2 (optional).  
listen 443 ssl http2;
```

```
# 只支持tls1.3
```

```
ssl_protocols TLSv1.2 TLSv1.3;
```

```
ssl_ciphers [TLS13+AESGCM+AES128|TLS13+CHACHA20]:TLS13+AESGCM+AES256:[EECDH+  
CDSA+AESGCM+AES128|EECDH+ECDSA+CHACHA20]:EECDH+ECDSA+AESGCM+AES256:EE  
DH+ECDSA+AES128+SHA:EECDH+ECDSA+AES256+SHA:[EECDH+aRSA+AESGCM+AES128|E  
CDH+aRSA+CHACHA20]:EECDH+aRSA+AESGCM+AES256:EECDH+aRSA+AES128+SHA:EE  
H+aRSA+AES256+SHA;
```

```
ssl_prefer_server_ciphers on;
```

```
ssl_early_data on;
```

```
ssl_ecdh_curve X25519:P-256:P-384;
```

```
ssl_session_cache shared:SSL:10m;
```

```
ssl_session_timeout 10m;
```

```
error_page 497 https://$host$request_uri;
```

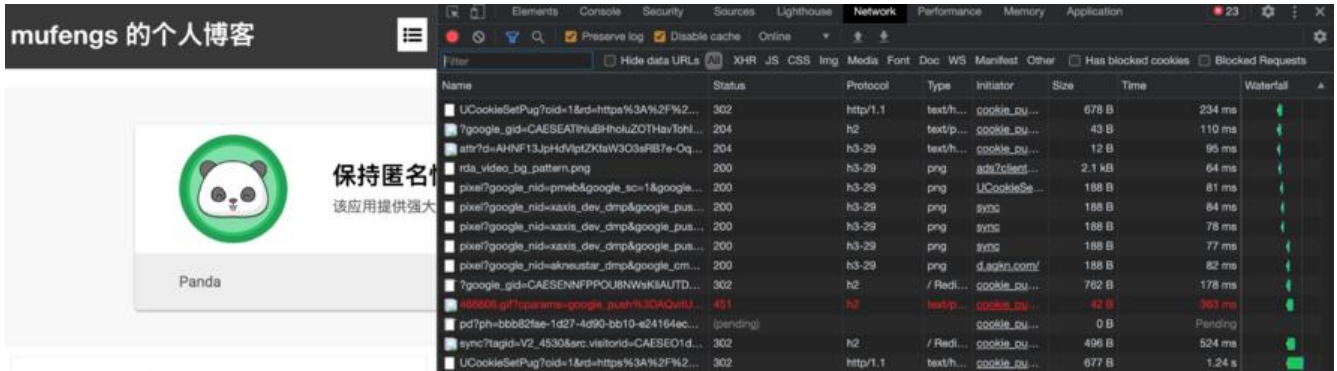
```
# OCSP Stapling 启用
```

```
ssl_stapling on;
ssl_stapling_verify on;
ssl_stapling_file /home/others/ocsp/nange.cn.ocsp.resp;
```

```
# Add Alt-Svc header to negotiate HTTP/3.
add_header alt-svc 'quic=":443"; ma=2592000;';
```

http3.0

我的博客是挂了cloudflare的cdn， 可以看出来cloudflare那边已经在使用http3了



Name	Status	Protocol	Type	Initiator	Size	Time	Waterfall
UCookieSetPug?oid=1&rd=https%3A%2F%2...	302	http/1.1	text/h...	cookie_sj...	678 B	234 ms	
?google_gid=CAESEATInu6HhcluZOT1avToH...	204	h2	text/p...	cookie_sj...	43 B	110 ms	
attr?d=AHNF13jphdVpt2KtaW3O3sRB7e-Oq...	204	h3-29	text/h...	cookie_sj...	12 B	95 ms	
rda_video_bg_pattern.png	200	h3-29	png	ads?client...	2.1 kB	64 ms	
pixel?google_nid=pmeb&google_sce=1&google...	200	h3-29	png	UCookieSe...	188 B	81 ms	
pixel?google_nid=xaxia_dev_dmp&google_pus...	200	h3-29	png	svcs	188 B	84 ms	
pixel?google_nid=xaxia_dev_dmp&google_pus...	200	h3-29	png	svcs	188 B	78 ms	
pixel?google_nid=xaxia_dev_dmp&google_pus...	200	h3-29	png	svcs	188 B	77 ms	
pixel?google_nid=akneustar_dmp&google_cm...	200	h3-29	png	d.agkn.com/	188 B	82 ms	
?google_gid=CAESENFPPOURNWwKIAUTD...	302	h2	/ Redi...	cookie_sj...	762 B	178 ms	
451	451	h2	text/p...	cookie_sj...	43 B	393 ms	
pd7ph-bbb82fae-1d27-4d90-bb10-e24164ec...	(pending)				0 B	Pending	
sync?tagid=V2_4530&rc.visitorid=CAESED1d...	302	h2	/ Redi...	cookie_sj...	496 B	524 ms	
UCookieSetPug?oid=1&rd=https%3A%2F%2...	302	http/1.1	text/h...	cookie_sj...	677 B	1.24 s	

如果本文对你有帮助，麻烦双击·666·加关注·

原文引用：

<https://www.infoq.cn/article/WhCOxbfbgtphY7ijv1kp>

<https://www.nange.cn/quic-and-http3-for-nginx.html>

https://blog.csdn.net/qq_22238021/article/details/81197157