

HTTP/2 in Action

sinsy 整理

2020-11-20

HTTP 历史

- HTTP 的诞生
  - 打开网页会经过什么节点
- HTTP 语法
  - HTTP/0.9
  - HTTP/1.0
  - HTTP/1.1
- HTTP 工具
  - 浏览器开发者工具
  - Advacned REST Client
  - 命令程序
    - curl
    - wget
    - httpie

什么是 HTTP/2

- HTTP1.1 的缺点
  - 传输消息阻塞
  - TCP 头阻塞问题
  - 缺少良好的浏览器支持管道
  - 文本传输
- 解决 HTTP1.1问题方案
  - 使用多个 HTTP 连接
  - 合并多个 HTTP 请求
  - 文
- HTTP/2 的前身
  - SPDY
    - 流多路利用
    - 请求优先级
    - HTTP 头部压缩
- HTTP/2 对性能影响
  - 更短时间加载图片
  - 创造更多连接

如何升级 HTTP/2

- nginx 配置
- 使用工具来检查协议

HTTP/2 如何实现

- HTTP/2 优化
  - 使用二进制格式替代文本格式
  - 多路复用代替同步请求
  - 流优先以及流控制
  - 首部压缩
  - 服务端推送
- HTTP/2 验证

HTTP/2 的服务推送

- 什么是 HTTP2 服务推送
- 如何推送
- 在浏览器如何运作
- 如何实现推送原理
- 要推送什么
- 推送对性能的影响

HTTP/2 优化

- HTTP1.1 优化手段
  - 压缩资源
  - 带宽增加
  - 域名分片
  - 内联资源
- HTTP/2 依然有效的优化手段
  - 减少传输数据量
  - 使用缓存防止重复发
  - Service Work 减少网络加载（就是离线资源）
  - 不发送不需要内容
  - HTTP 资源暗示
  - 优化 HTTPS
- 针对 HTTP1.1 HTTP/2做优化策略
  - 分别计算使用 HTTP/1 以及 HTTP/2 的数量，分别优化

HTTP/2 高级进阶

- 流状态
- 流优先
- 流控制
- HPACK 首部压缩

HTTP 未来的发展

- QUIC 以及 HTTP/3
- 什么是 QUIC
  - 使用 UDP 协议做底层

HTTP/2 争议

- HTTP/2 争议
  - 加密问题
  - cookie 和第三方跟踪
  - 不再是无状态
- 实际应用
- 未来的版本规划
  - 进一步改进二进制协议
  - 传输层之上进一步优化 HTTP
  - 新的 HTTP 方法
  - 新的 HTTP 首部
  - 新的内容编码
  - 新的状态
  - 摆脱TCP（待定）