China - Beijing

HTTPS性能优化

罗 成 / 腾讯资深研发工程师

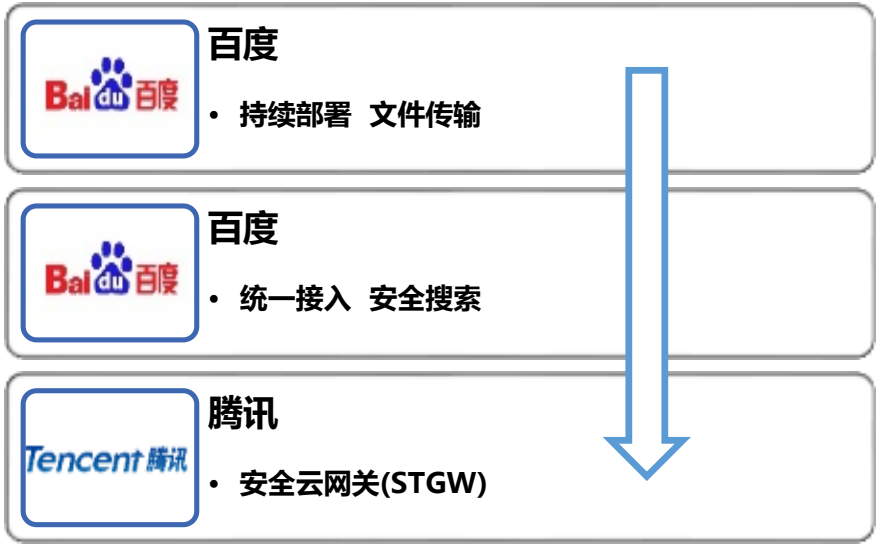
ArchSummit 全球架构师峰会 2016

[北京站]

Geekbang. InfoQ

个人简介

- 微博: 互联网罗成
- 知乎ID: helloworlds
- 知乎专栏:《HTTPS原理和实践》
  https://zhuanlan.zhihu.com/https

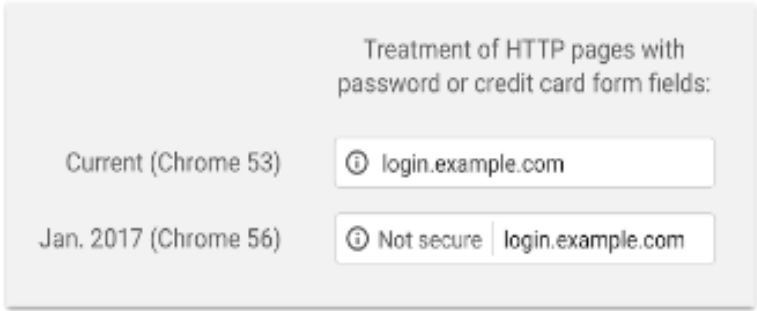


大纲

- 计算性能分析与优化
- 无秘钥加载
- 证书优化

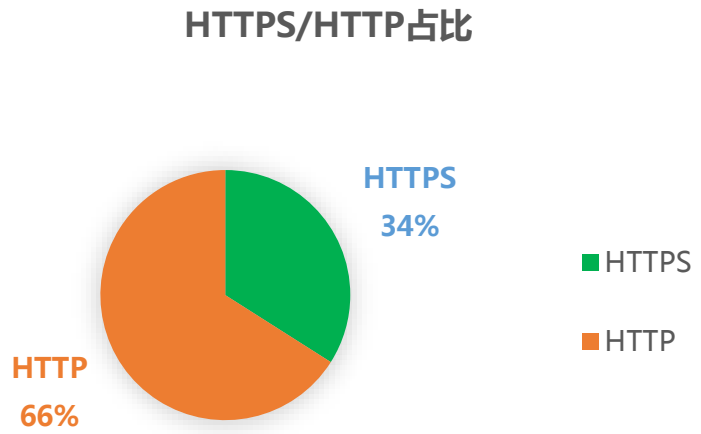
HTTPS
是互联网的趋势

- **HTTPS的优势**
  - 内容加密
  - 身份认证
  - 消息校验



为什么66%的网站不支持HTTPS?

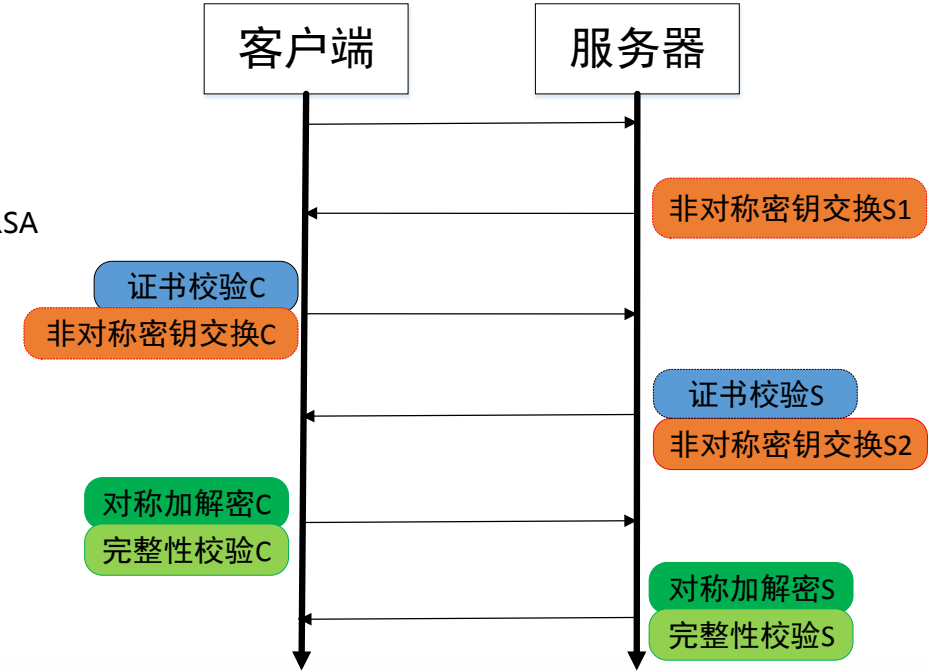
- **慢**
  - 移动端慢500ms以上
- **贵**
  - 增加服务器成本
    - HTTPS性能不到HTTP 1/10
  - 证书成本
    - 申请繁琐
    - 价格不一
    - 容易过期、失效



HTTPS为什么增加服务器成本？

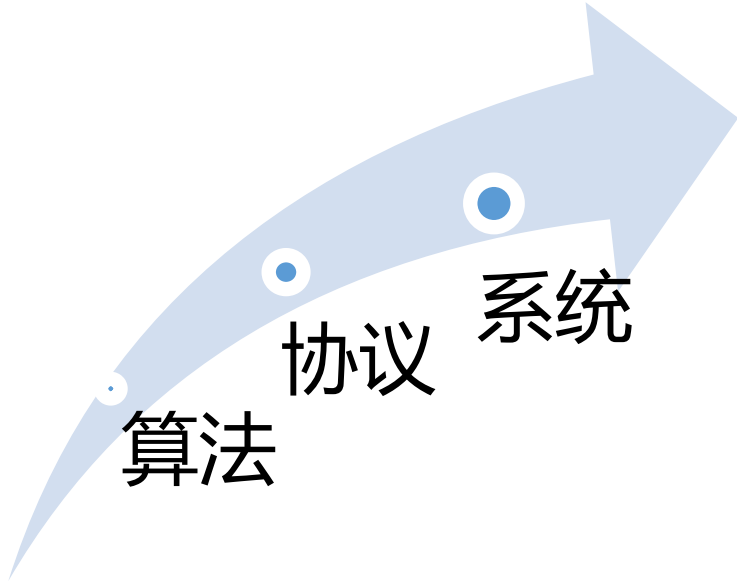
HTTPS主要的计算环节

- **非对称密钥交换**
  - RSA、ECDHE_RSA
- **对称加解密**
  - AES,RC4
- **一致性验证**
  - SHA2
- **证书校验**
  - RSA、ECDSA



计算性能的分析维度

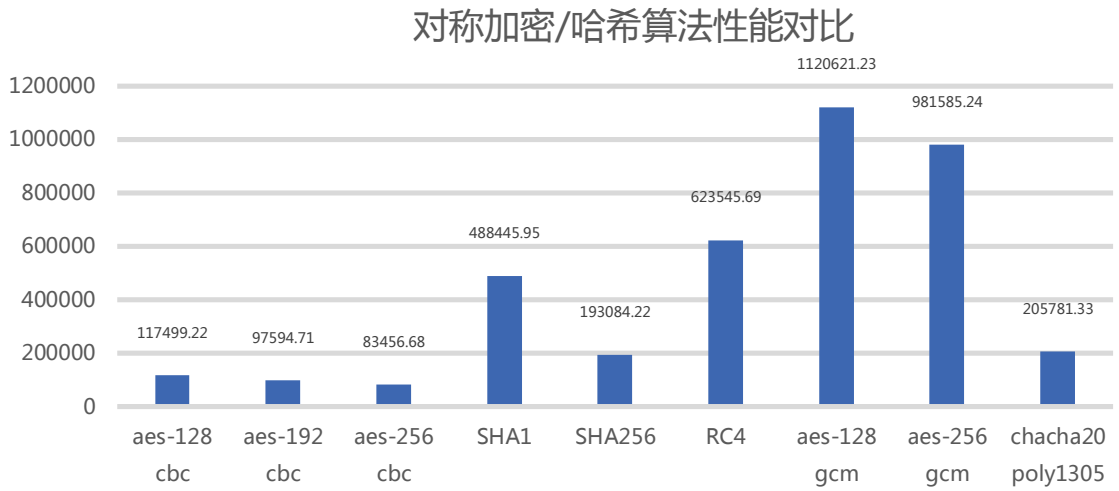
- **算法**
  - openssl speed
  - 对称加密，非对称密钥交换，签名算法，一致性校验算法
- **协议**
  - 完全握手
  - 函数级耗时
- **系统**
  - 热点事件
  - 工程实现



计算性能分析---对称加密、一致性校验算法测试

- openssl speed -elapsed evp

算法名	每秒处理的字节数（块大小为1K）	处理4K字节需要的时间
AES-128 CBC	117499.22k	0.00003s
AES-192 CBC	97594.71k	0.00004s
AES-256 CBC	83456.68k	0.000047s
SHA1	488445.95k	0.000008s
SHA256	193084.22k	0.00002s
RC4	623545.69k	0.0000064s
AES-128 GCM	1120621.23k	0.0000035s
AES-256 GCM	981585.24k	0.000004s
CHACHA20 POLY1305	205781.33k	0.000019s

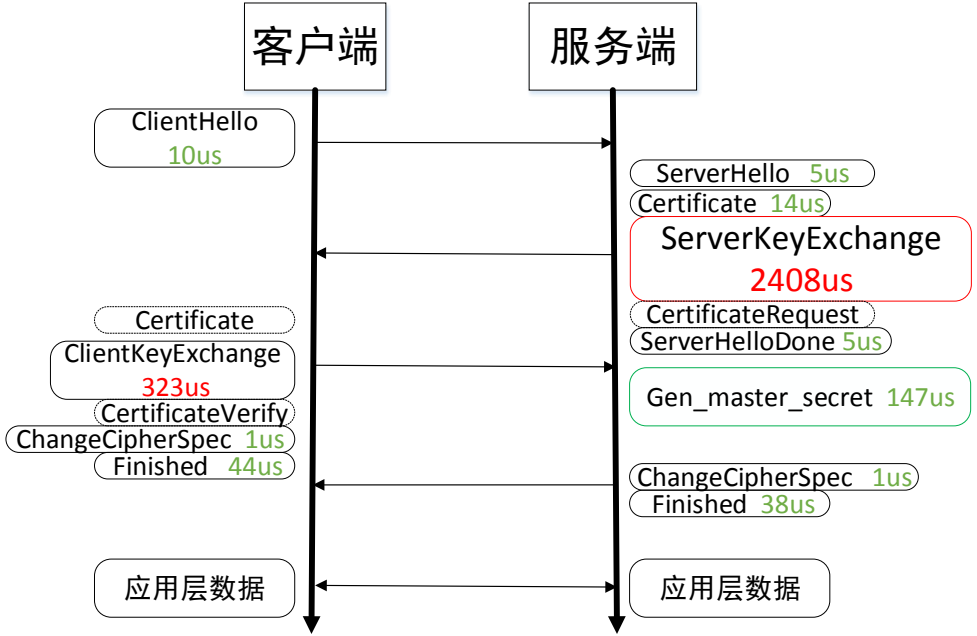


计算性能分析--- 密钥交换、签名算法测试

- openssl speed RSA
  - RSA签名计算一秒钟最多809次

算法名	Sign	Verify	Sign/s	Verify/s
RSA 2048	0.001235s	0.000037s	809.4	27339.7
DSA 2048	0.000435s	0.000463s	2297.0	2161.6
Ecdsa(nistp256)	0.0001s	0.0001s	16576.9	7012.4

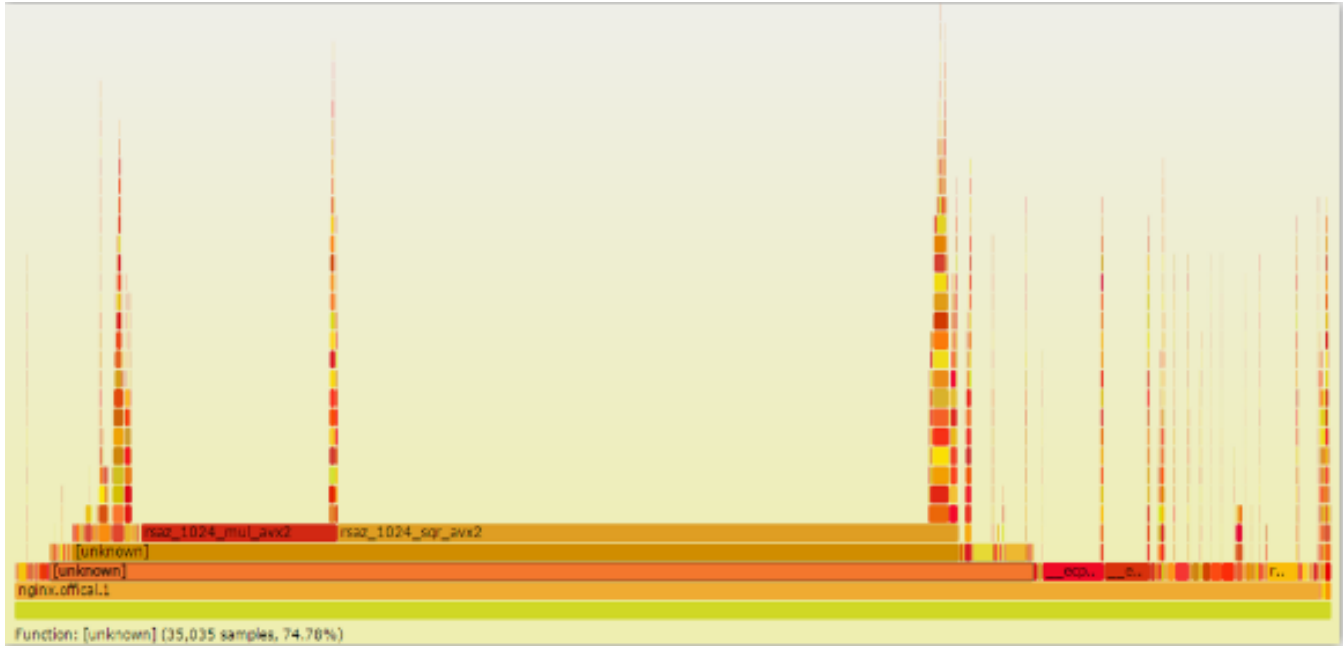
计算性能分析---握手协议的耗时



ECDHE_RSA握手耗时数据

计算性能分析--- 热点事件

- perf record / flame graph



计算性能分析结论

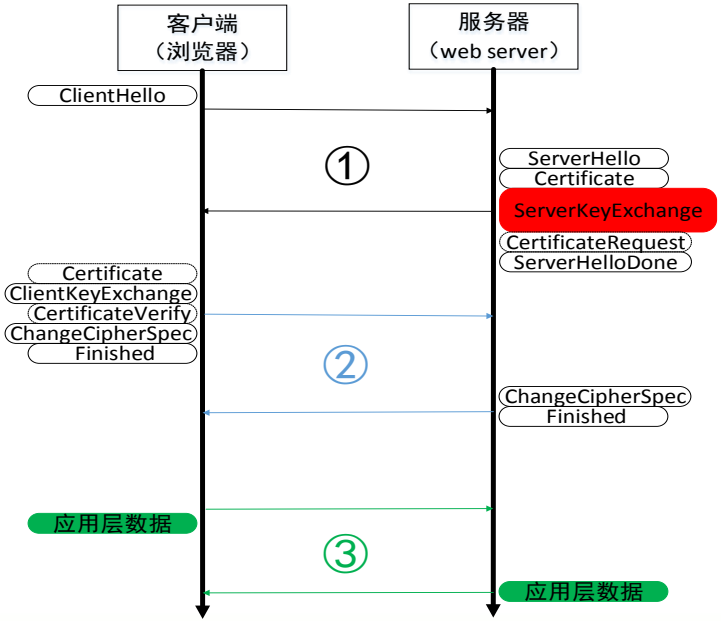
- 完全握手
  - 性能降低至普通HTTP性能的10%以下
- RSA算法对性能的影响
  - 消耗整体性能的75%左右
- ECC椭圆曲线
  - 约占整体计算量的7%
- 对称加解密及MAC计算
  - 对性能影响很小（微秒级别）

如何优化计算性能？

- 减少完全握手
  - 分布式session cache
  - 全局Session ticket
  - 自定义session ticket
- RSA异步代理计算

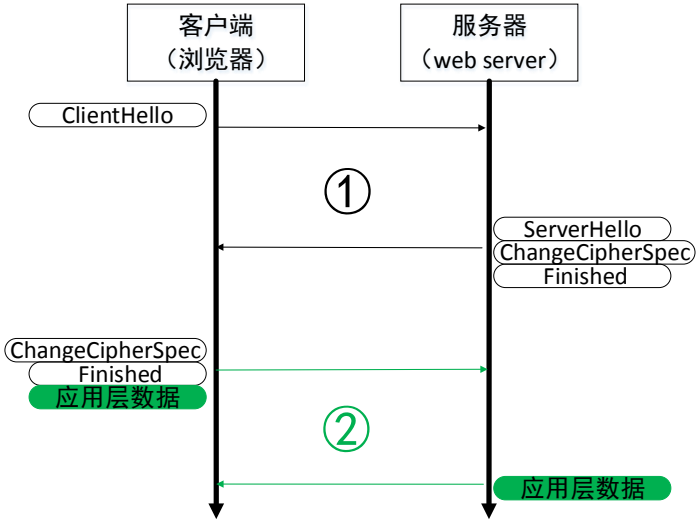
- 对称加密优化

完全握手

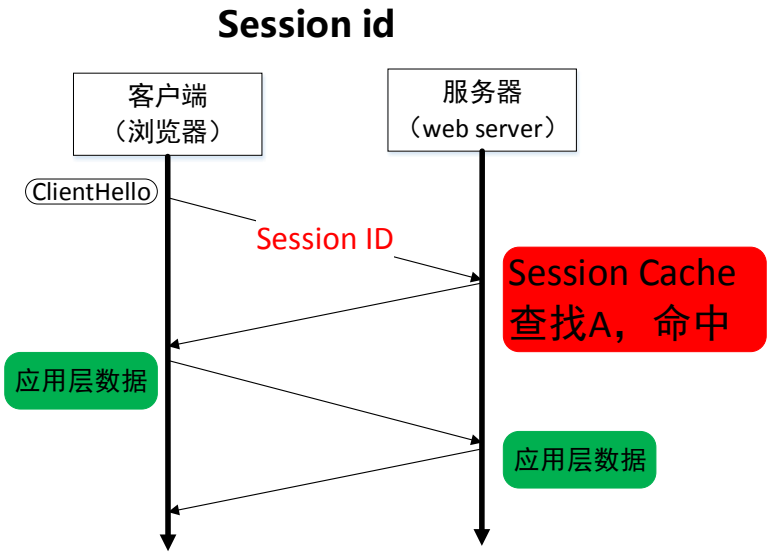

客户端
（浏览器）
服务器
（web server）
ClientHello
①
ServerHello
Certificate
ServerKeyExchange
CertificateRequest
ServerHelloDone
Certificate
ClientKeyExchange
CertificateVerify
ChangeCipherSpec
Finished
②
ChangeCipherSpec
Finished
应用层数据
③
应用层数据

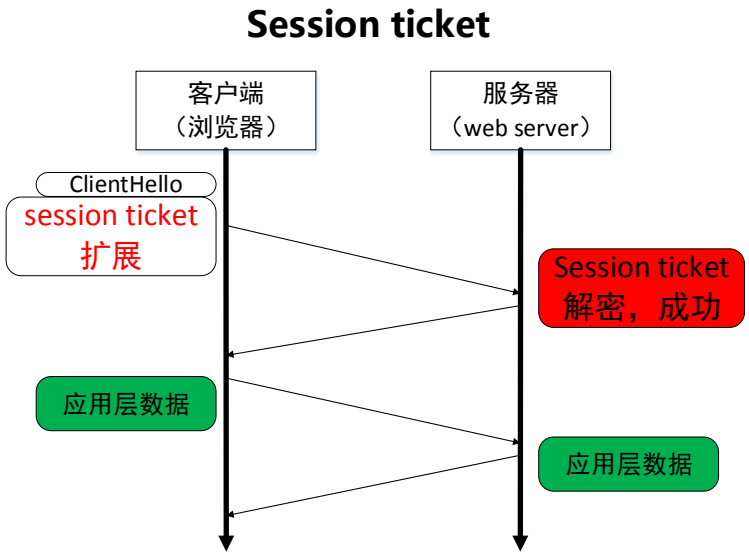
VS

简化握手


客户端
（浏览器）
服务器
（web server）
ClientHello
①
ServerHello
ChangeCipherSpec
Finished
ChangeCipherSpec
Finished
应用层数据
②
应用层数据

协议层面实现简化握手

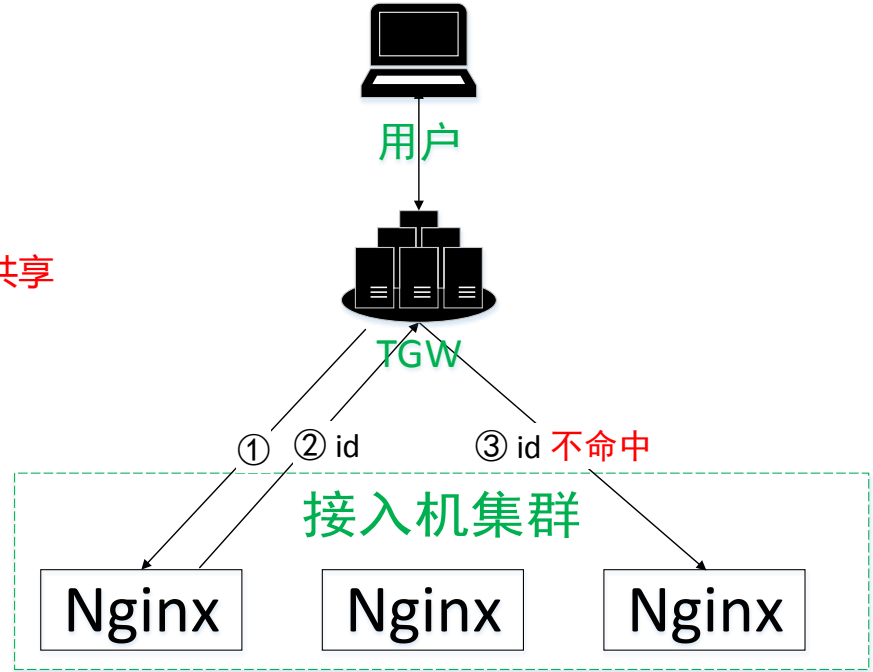

Session id
客户端
（浏览器）
服务器
（web server）
ClientHello
Session ID
Session Cache
查找A，命中
应用层数据
应用层数据


Session ticket
客户端
（浏览器）
服务器
（web server）
ClientHello
session ticket
扩展
Session ticket
解密，成功
应用层数据
应用层数据

Session Resumption

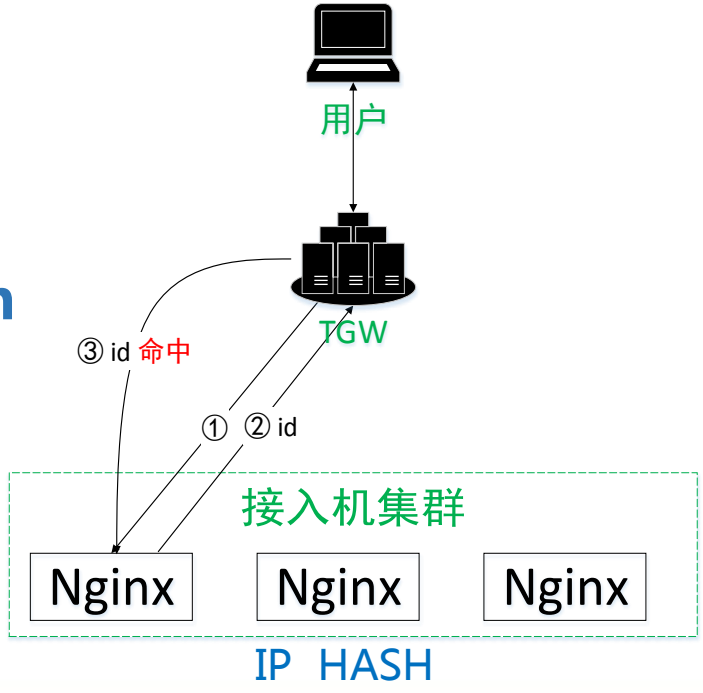
---工程实现的局限

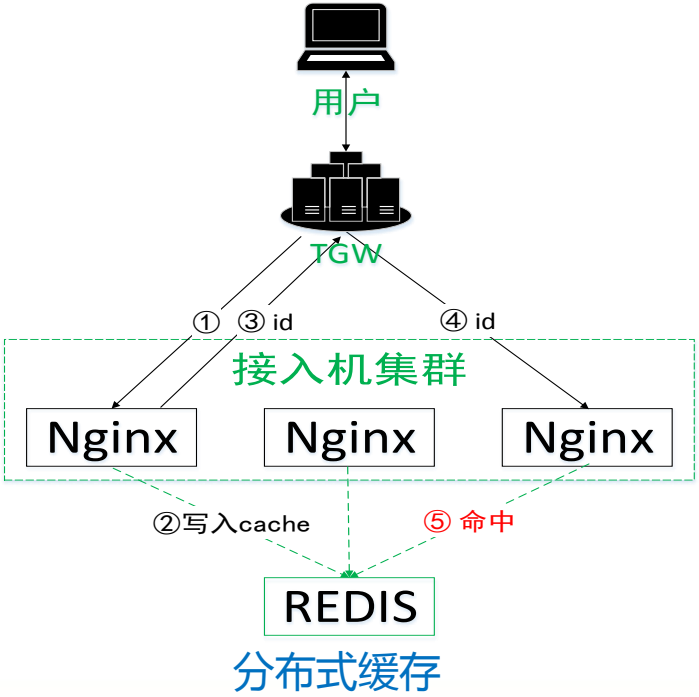
- nginx单机多进程间共享
- Openssl同步
- 多接入机环境
  - 命中率低



Session Resumption

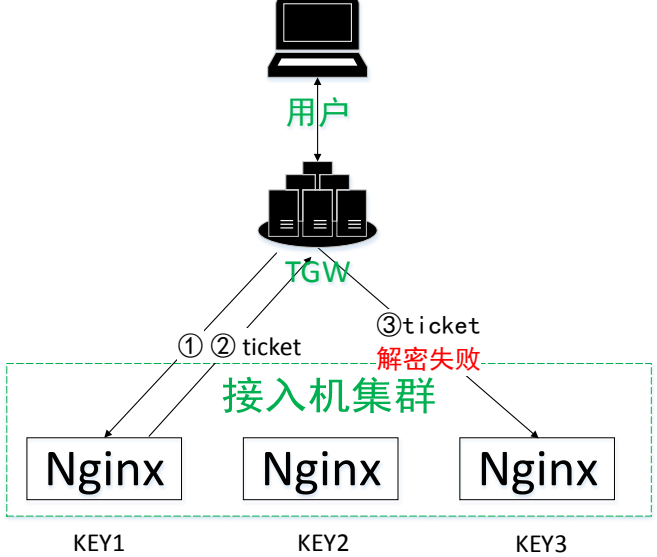
---分布式session cache



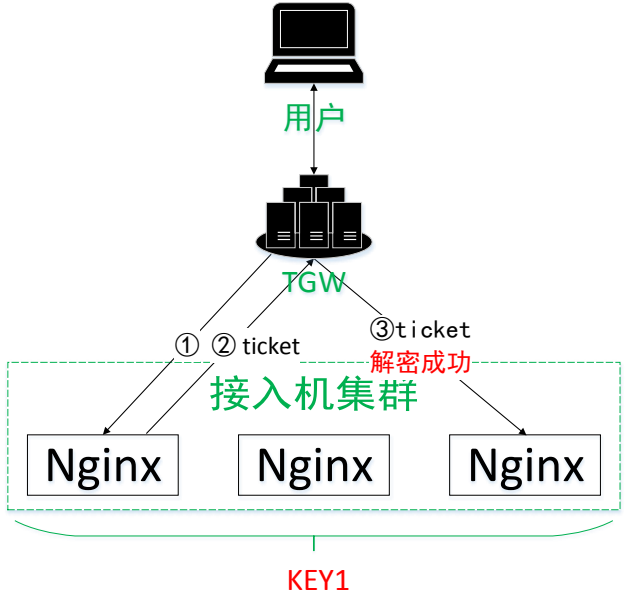


Session Resumption

---全局session ticket



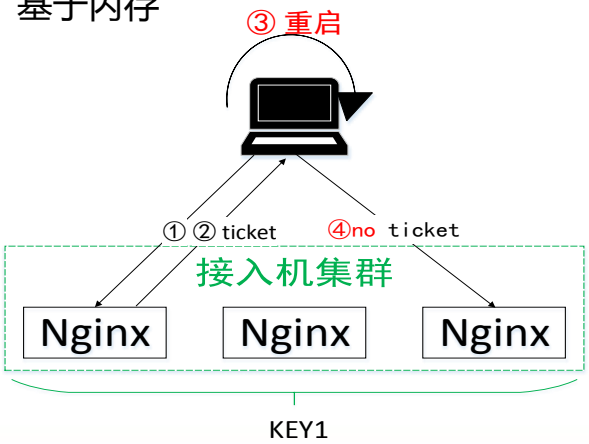
- openssl生成key
  openssl rand 48 >key1



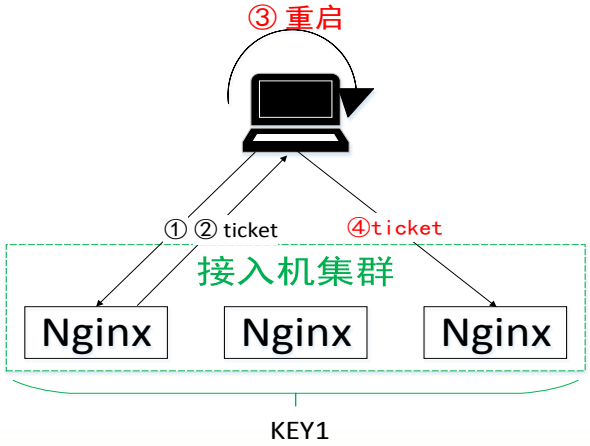
- Nginx配置：
  ssl_session_ticket_key key1;
  ssl_session_ticket_key pre.key;

Session Resumption---self session ticket

- 完全握手的场景
  - App，浏览器，OS重启
  - 基于内存



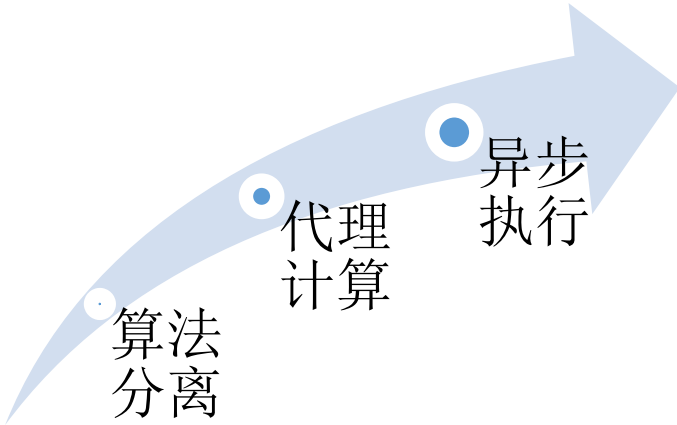
- 安全性分析
  - 私有路径



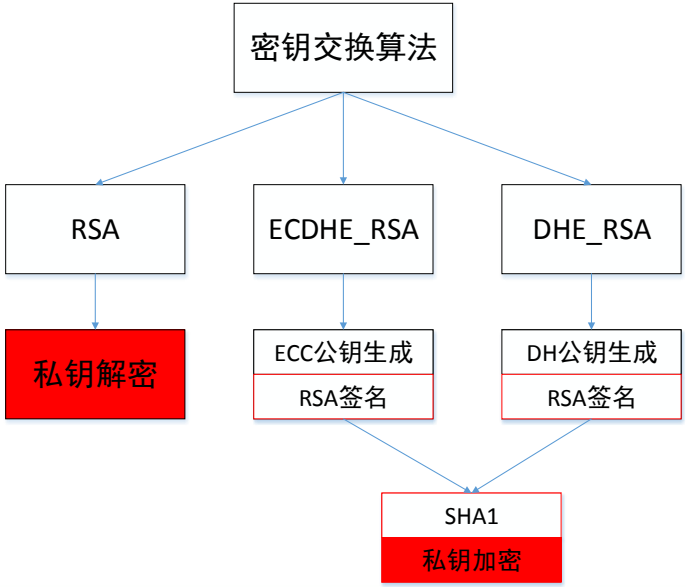
完全握手性能优化

---RSA异步代理计算

- 算法分离
  - RSA，ECDHE_RSA，DHE_RSA
- 代理计算
  - 硬件加速卡，GPU，空闲CPU
- 异步执行
  - Openssl状态机



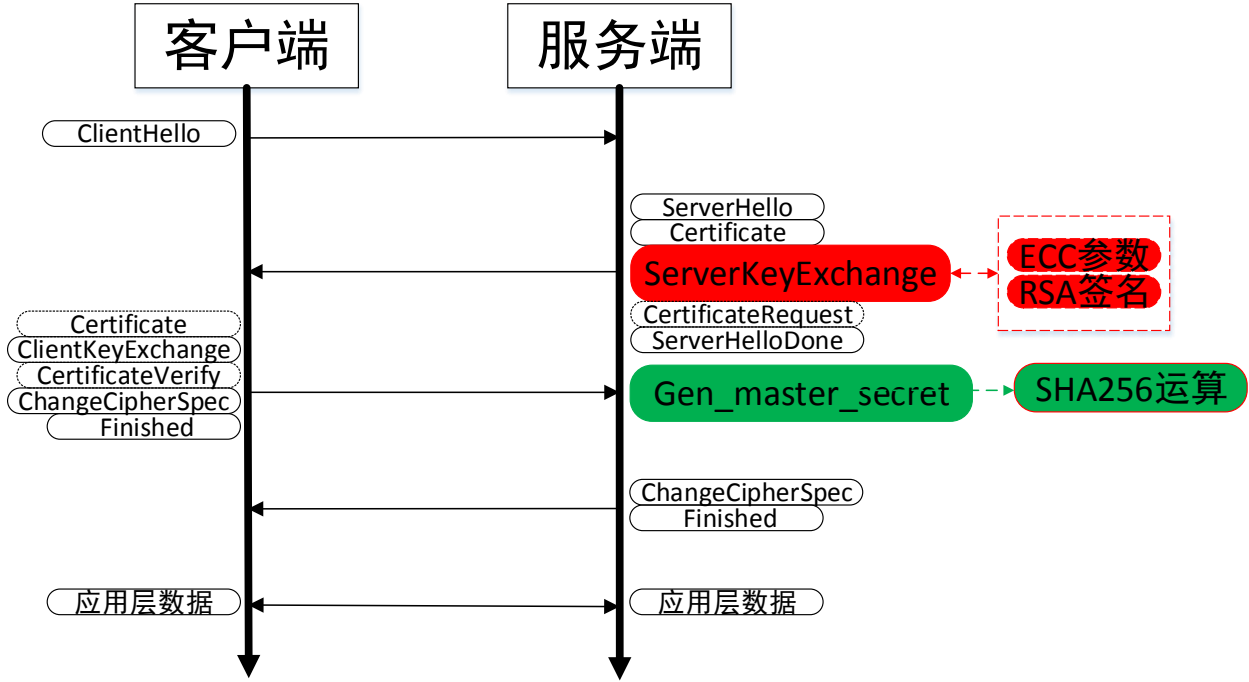
RSA异步代理计算---算法分离



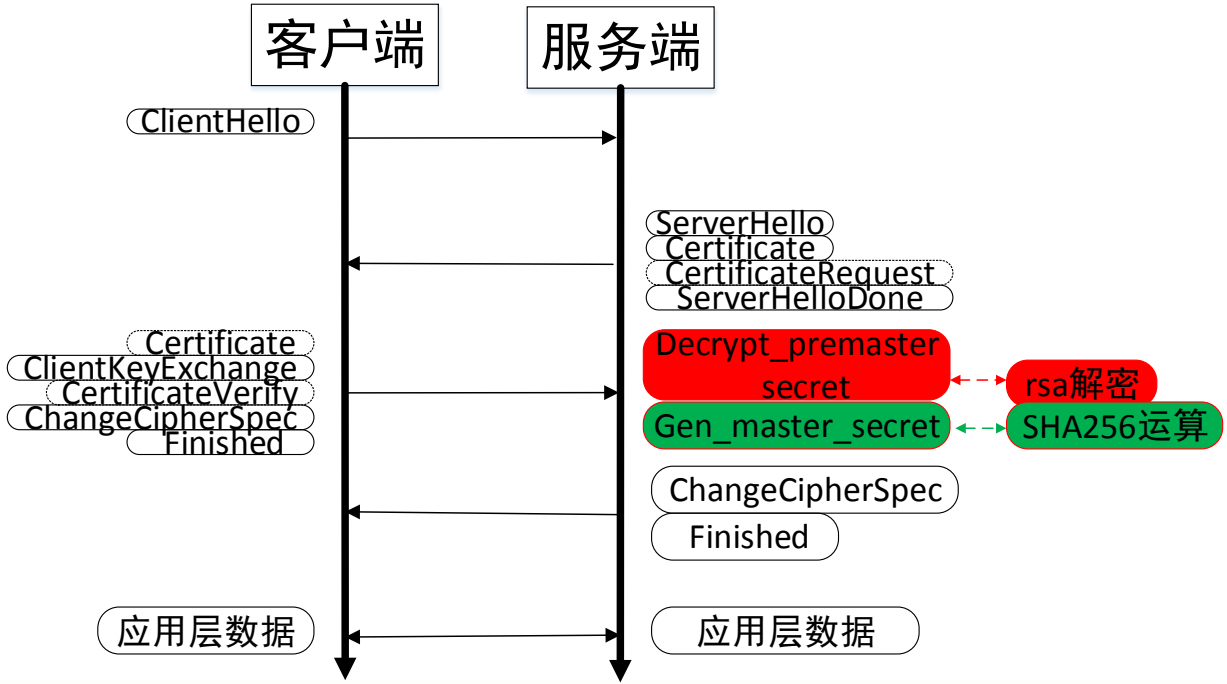
$$c \equiv m^e \bmod(n)$$

$$m \equiv c^d \ mod\ (n)$$

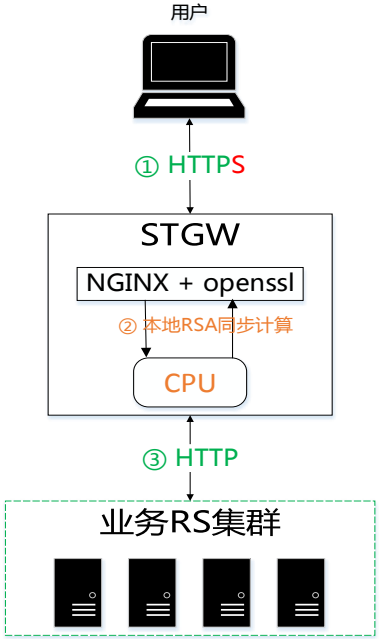
算法分离---ECDHE_RSA



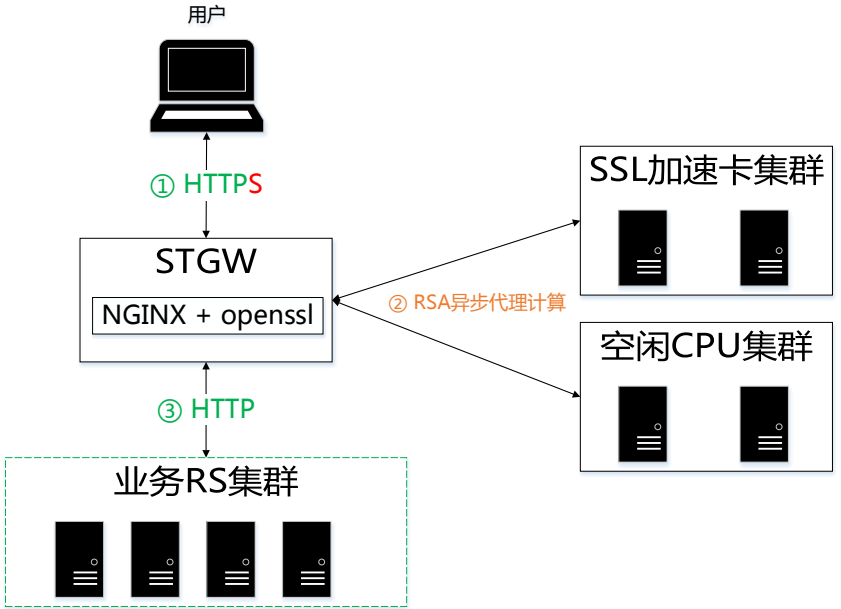
算法分离---RSA


客户端
服务端
ClientHello
ServerHello
Certificate
CertificateRequest
ServerHelloDone
Certificate
ClientKeyExchange
CertificateVerify
ChangeCipherSpec
Finished
Decrypt_premaster secret
rsa解密
Gen_master_secret
SHA256运算
ChangeCipherSpec
Finished
应用层数据
应用层数据

异步代理计算---架构



本地CPU同步计算模型



硬件加速卡异步代理计算模型

异步代理计算---工程实现

- **Nginx**
  - Event/ngx_event_openssl_engine.c
  - 模块无法实现
- **Openssl**
  - Ssl/s3_srvr.c
  - 1.1.0支持异步
- **性能65000 cps，提升了3.5倍**
  - ecdhe_rsa

ECC椭圆曲线优化

- 优先使用NIST p256
  - P224以上安全
- Openssl版本
  - 1.0.1l

- Openssl1.1.0b

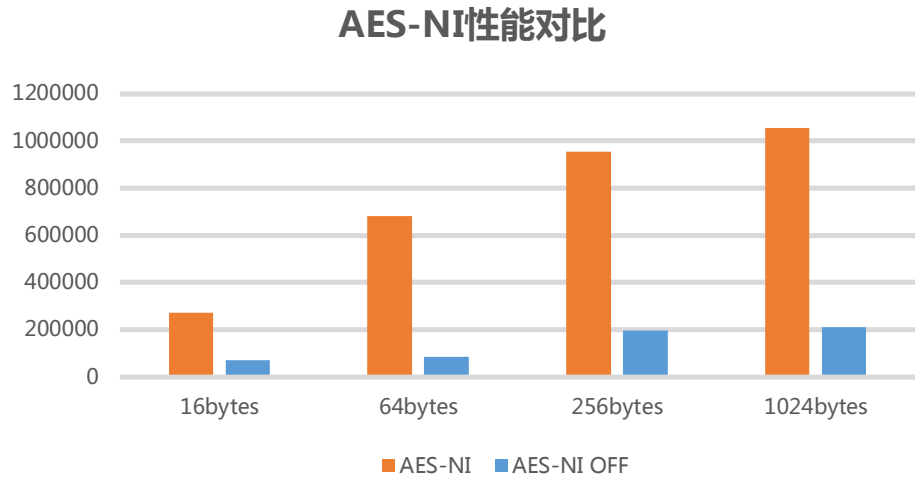
算法名	OP/s	OP
ecdh (nistp192)	0.0003s	3805.3
ecdh (nistp224)	0.0004s	2808.8
ecdh (nistp256)	0.0001s	10271.9
ecdh (nistp384)	0.0009s	1176.0

- Openssl1.0.1e

算法名	OP/s	OP
ecdh (nistp256)	0.0004s	2548.8
ecdh (nistp384)	0.0008s	1192.8

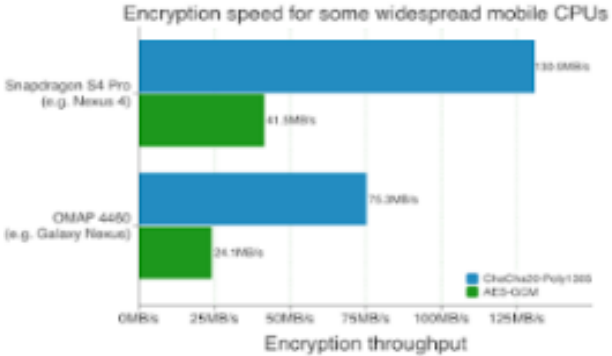
块式对称加密算法的优化

- **AES-GCM**
  - 性能最高
- **AES-NI**
  - 性能提升5倍左右
  - EVP_EncryptInit_ex vs AES_encrypt
  - OPENSSL_ia32cap="~0x200000200000000" openssl speed -elapsed -evp aes-128-gcm
- **高性能CPU**
  - TCO



流式对称加密算法的优化

- Chacha20-Poly1305
  - 3倍性能提升



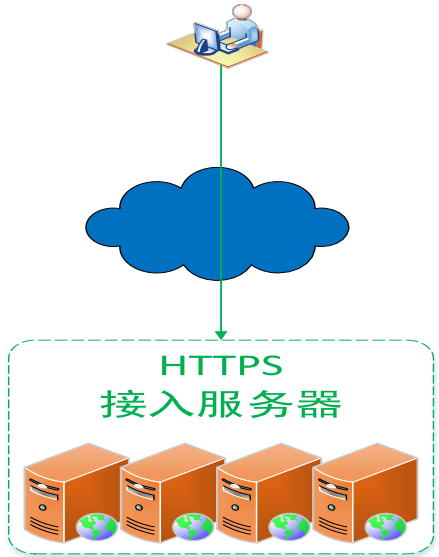
- RC4
  - SSLv3，安全性强于AES-CBC

大纲

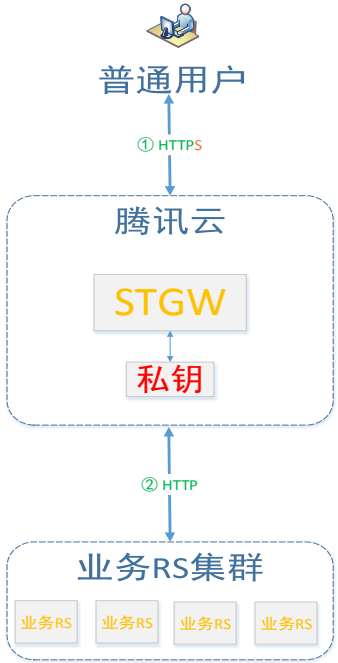
- 计算性能分析与优化
- 无秘钥加载
- 证书优化

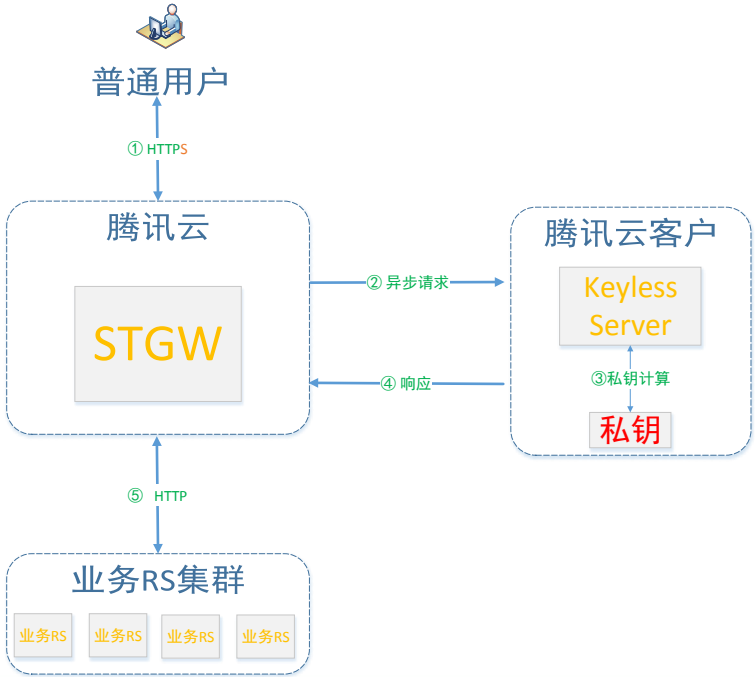
Keyless无秘钥加载---同机部署的风险

- 私钥是安全的根本
- 同机部署
  - 接入服务器
- 泄露风险大
  - CDN
  - 金融客户



无秘钥加载---流程


普通用户
① HTTPS
腾讯云
STGW
私钥
② HTTP
业务RS集群
业务RS
业务RS
业务RS
业务RS

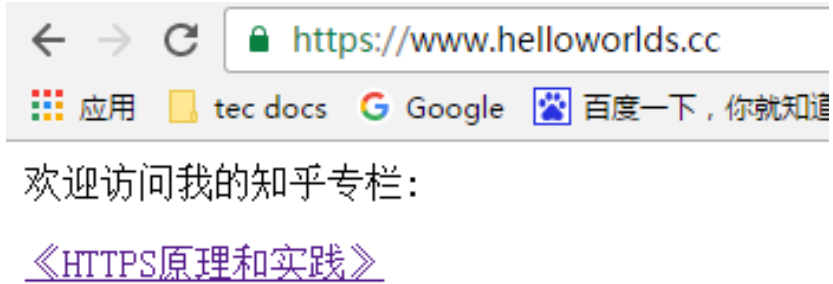

普通用户
① HTTPS
腾讯云
STGW
② 异步请求
④ 响应
腾讯云客户
Keyless Server
③私钥计算
私钥
⑤ HTTP
业务RS集群
业务RS
业务RS
业务RS
业务RS

大纲

- 计算性能分析与优化
- 无秘钥加载
- 证书优化

个人用户的选择---Let’ s Encrypt

- **优点**
  - 免费，开源
  - 自动部署
- **缺点**
  - DV
  - 风险高
  - 兼容性低
- **建议**
  - 推荐个人用户



企业用户的证书选择

- **EV && OV**
  - DV不安全
- **云**
  - 腾讯云、阿里云、AWS等
- **优势**
  - 申请简单、成本低
  - 自主证书品牌

腾讯云

证书签名的选择--RSA or ECDSA?

- RSA
  - 兼容性好
  - 服务端性能差
- ECDSA
  - 兼容性差
    - XP不支持
    - 支持ECDHE，但系统缺少root ca
  - 服务端性能好，客户端性能差
- 同时支持
  - 成本增加

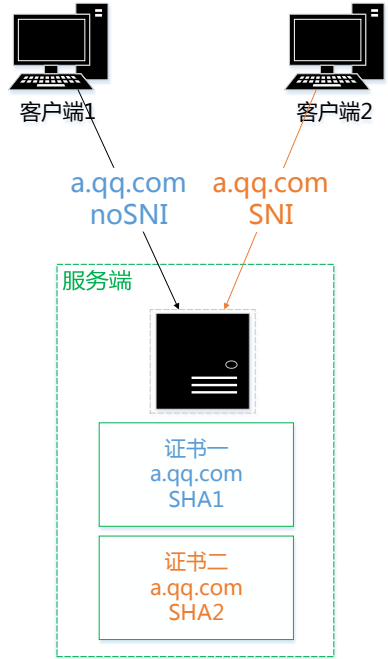
浏览器	最低版本
Apple Safari	4 (On ECC Compatible OS)
Google Chrome	1.0 (On ECC Compatible OS)
Microsoft Internet Explorer	7 (On ECC Compatible OS)
Mozilla Firefox*	2.0

操作系统	最低版本
Apple OS X	OS X 10.6
Google Android	4.0
Microsoft Windows	Windows Vista
Red Hat Enterprise Linux	6.5

证书的问题

---兼容SHA1、SHA256

- **SHA1 or SHA256**
  - SHA1不安全
  - SHA2兼容性差
- **不支持SNI = 不支持SHA2？**
- **Nginx配置**
  - 证书一 server_name空



HTTPS的发展趋势

●更广

- http2主流实现强制使用https
- ATS 强制使用HTTPS
- Chrome mark http unsecure

●更强

- RSA 2048 -> 4096
- RSA -> ECC

●更快

- Tls1.3
- QUIC

●更开放

- Let’ s encrypt

THANKS