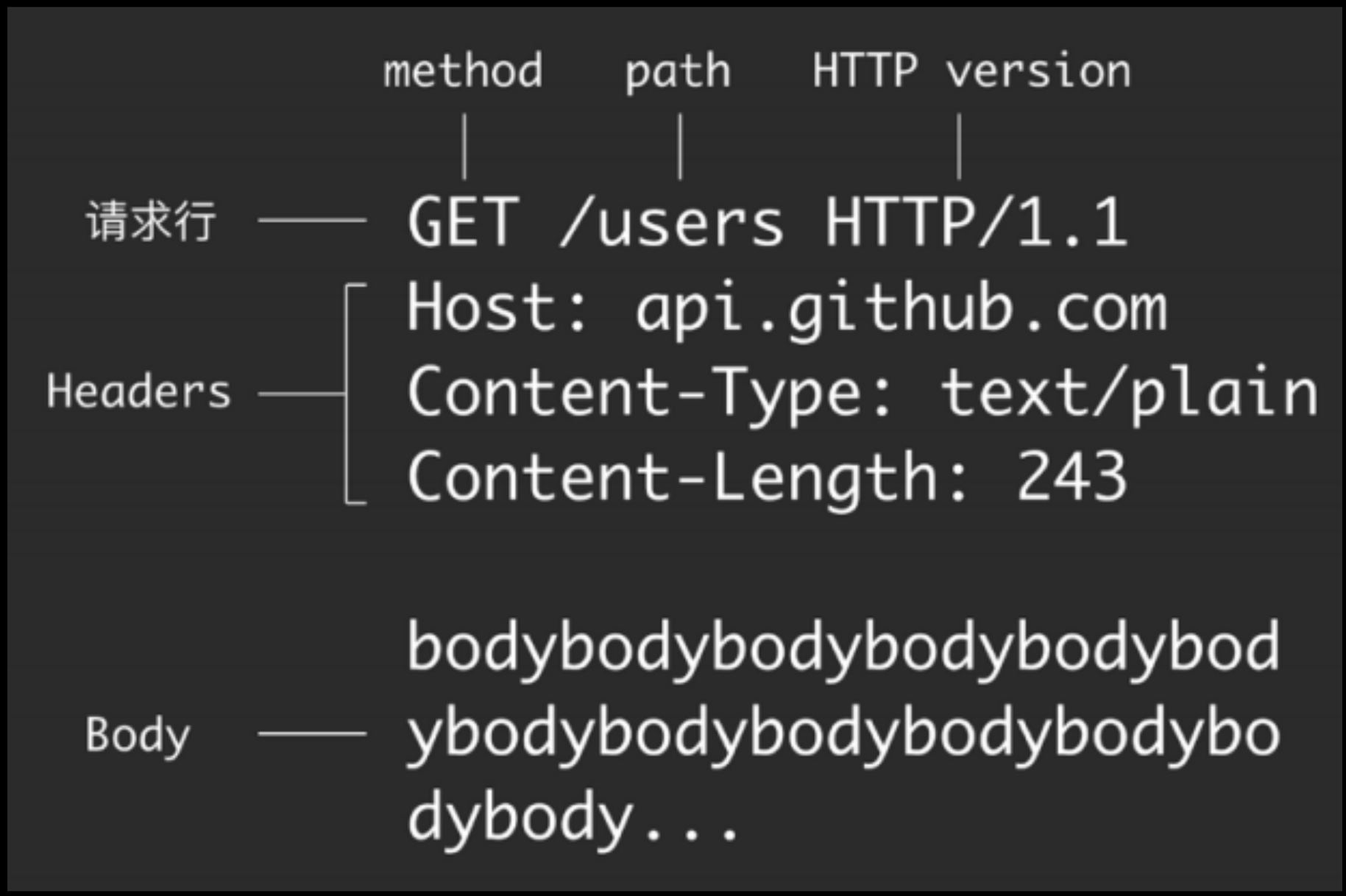
HTTP报文 & HTTPS握手

HTTP请求报文


method
path
HTTP version
请求行
GET /users HTTP/1.1
Host: api.github.com
Headers
Content-Type: text/plain
Content-Length: 243
Body
bodybodybodybodybodybodybod
ybodybodybodybodybodybodybo
dybody...

METHOD

- GET
  - 只获取资源，不对数据进行添加、修改
  - 不发送BODY
- POST
  - 增加或者修改数据
  - 给服务器发送的数据都在body里面
- PUT
  - 用于修改数据
  - 给服务器发送的数据都在body里面
- DELETE
  - 删除数据
  - 没有Body

HEADER

- HOST

  指明服务器域名以及端口号。并不是用于寻址的，而在虚拟主机的情况下找到具体服务器。

- Content-type
  - text/html : 表示内容是HTML
  - multipart/form-data：复杂表单，
  - application/json, image/jpeg, application/zip……

- Content-length

  指明BODY的长度（字节长度），用于表示数据是否结束。

- Accept-Range / Range /Content-Range
  - Accept-Range:bytes，响应报文中出现，用于表示服务器端支持按字节获取范围数据。
  - Range：bytes=<start>-<end> ，请求报文中出现，用户表示按照start到end范围获取数据。
  - Content-Range：bytes=<start>-<end>， 响应报文中出现，表示出现的是那段报文。

HTTP响应报文

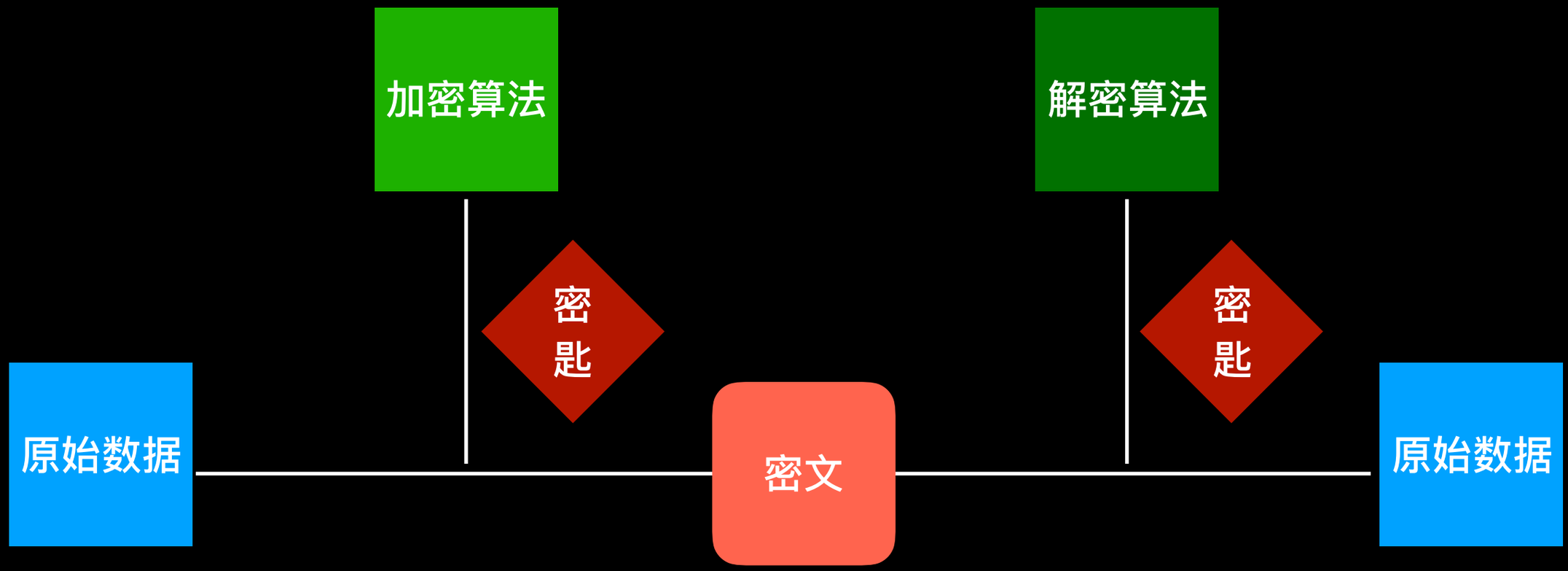

HTTP version
status code
status message
状态行
HTTP/1.1 200 OK
Headers
content-type: application/json; charset=utf-8
cache-control: public, max-age=60, s-maxage=60
vary: Accept,Accept-Encoding
etag: W/"02eec5b334b0e4c05253d3f4138daa46"
content-encoding: gzip
Body
[{"login":"mojombo","id":1,"node_id":"MDQ6VXNlcjE
=","avatar_url":"https://avatars0.githubuserconte
nt.com/u/1?v=4","gravat.....

STATUS CODE

- 1XX： 暂时性消息： 101 切换协议）
- 2XX：成功：200 ok、204 创建成功
- 3XX： 重定向：307
- 4XX： 客户端错误：400
- 5XX： 服务器端错误 500

加密算法

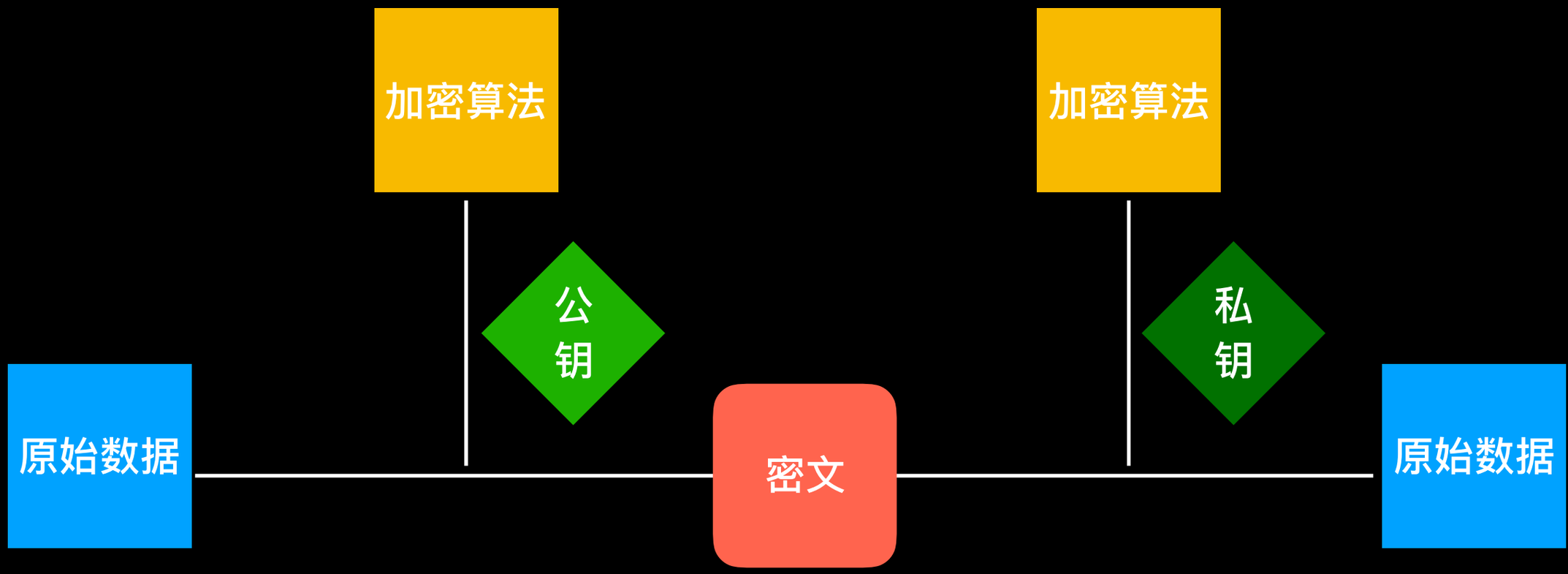
对称加密



常见对称加密

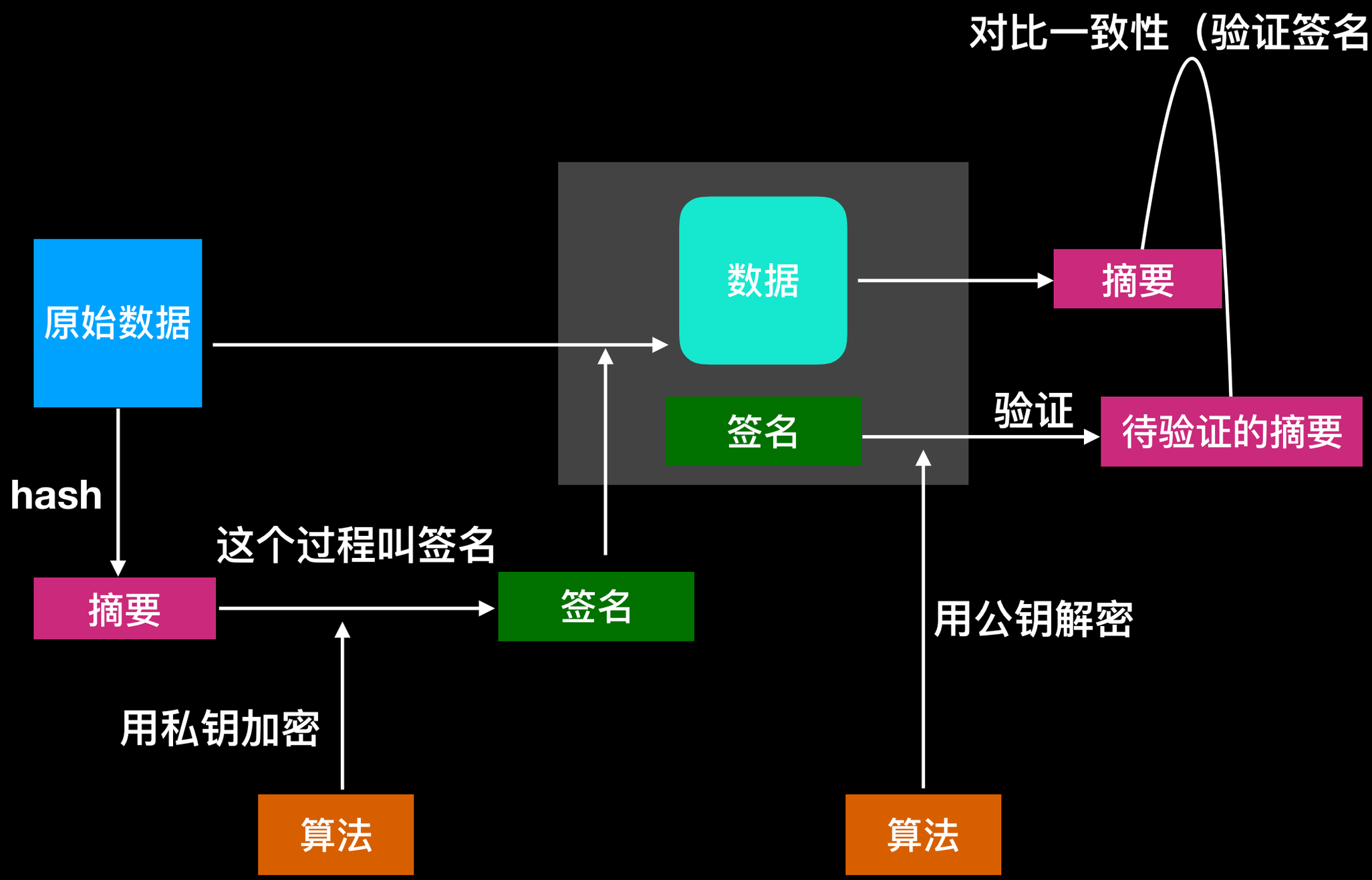
DES、**3DES**、**DESX**、**Blowfish**、**IDEA**、**RC4**、**RC5**、**RC6**和**AES**

非对称加密



常见非对称加密算法

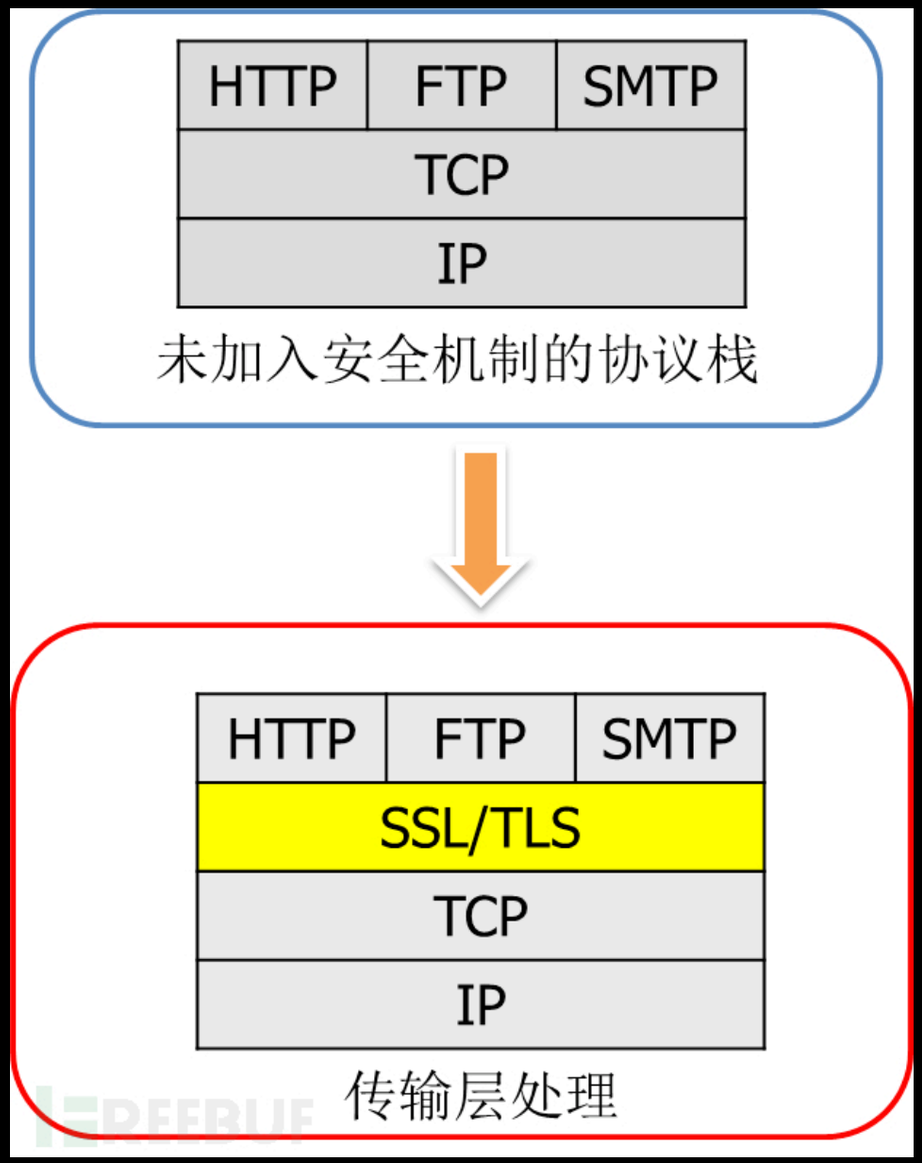
RSA、**ECC**（移动设备用）、**Diffie-Hellman**（**DH**）、
El Gamal、**DSA**（数字签名用）


对比一致性（验证签名）
原始数据
数据
摘要
签名
验证
待验证的摘要
hash
这个过程叫签名
摘要
签名
用私钥加密
用公钥解密
算法
算法

常见的Hash算法

MD2、MD4、MD5、HAVAL、SHA、SHA-1、HMAC、HMAC-MD5、HMAC-SHA1

HTTPS结构


HTTP
FTP
SMTP
TCP
IP
未加入安全机制的协议栈
HTTP
FTP
SMTP
SSL/TLS
TCP
IP
传输层处理


应用层
SSL/TLS
传输层
网络层
数据链路层
物理层

HTTPS握手过程

Client

Server

ClientHello

1. TLS Version
2. Radom
3. Cipher suites

ServerHello

1. TLS Version
2. Radom
3. Cipher suites

Certificate

1. 服务器的证书。
   公钥、名称、证书的签名。
2. 签发[服务器证书]机构的证书
   公钥(用于验证服务器证书签名的)、名称、签名

ServerKeyExchage

对于Certificate的补充（可选步骤）

- 协商采用RSA加密，但是服务器没有提供RSA证书
- 协商采用了DH加密，没有提供DH参数
- 协商采用fortezza_kea加密，但是服务器端证书没有提供参数

ServerHelloDone

类似于TCP的ACK

Client

Client Key Exchange

Pre-master secret

Server

Change Cipher Suites

客户端通知服务器，后续的报文将被加密

Encrypted Handshake message

握手后的第一条加密消息

服务器通知客户端，后续的报文将被加密

Change Cipher Suites

握手后的第一条加密消息

Encrypted Handshake message

GlobalSign Root CA

↳ GlobalSign Organization Validation CA - SHA256 - G2

↳ baidu.com

baidu.com

签发者：GlobalSign Organization Validation CA - SHA256 - G2

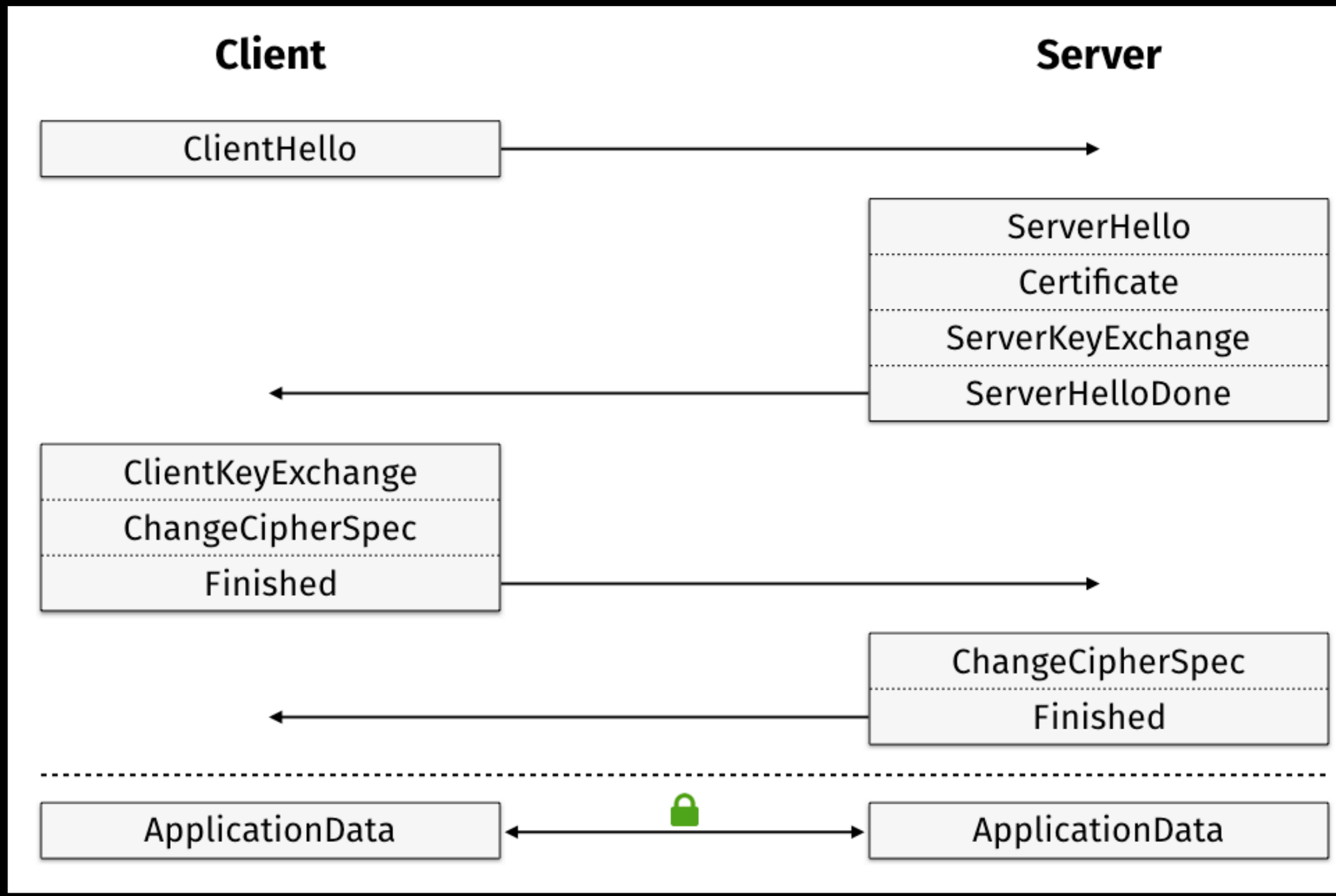
过期时间：2019年5月26日 星期日 中国标准时间 13:31:02

此证书有效

▼ 细节

主题名称	
国家/地区	CN
省/市/自治区	beijing
所在地	beijing
组织单位	service operation department
组织	Beijing Baidu Netcom Science Technology Co., Ltd
常用名称	baidu.com
签发者名称	
国家/地区	BE
组织	GlobalSign nv-sa
常用名称	GlobalSign Organization Validation CA - SHA256 - G2

好


Client
Server
ClientHello
ServerHello
Certificate
ServerKeyExchange
ServerHelloDone
ClientKeyExchange
ChangeCipherSpec
Finished
ChangeCipherSpec
Finished
ApplicationData
ApplicationData

Client Hello

- TSL Version
  - 表示Client最高支持的TSL版本，最终的使用版本由Server决定，如果Server不支持Client的版本，Client需要对TSL降级。
- Random
  - 一个32位随机数，用于生成最后加密的密钥。
- Cipher Suites 加密套件
  - 客户端可以支持加密算法列表。每一个套件都代表一个密匙规格，都有“TSL”开头，接着是密匙交换算法_WITH_数据传输加密算法_认证算法