

N g i n X

陶辉

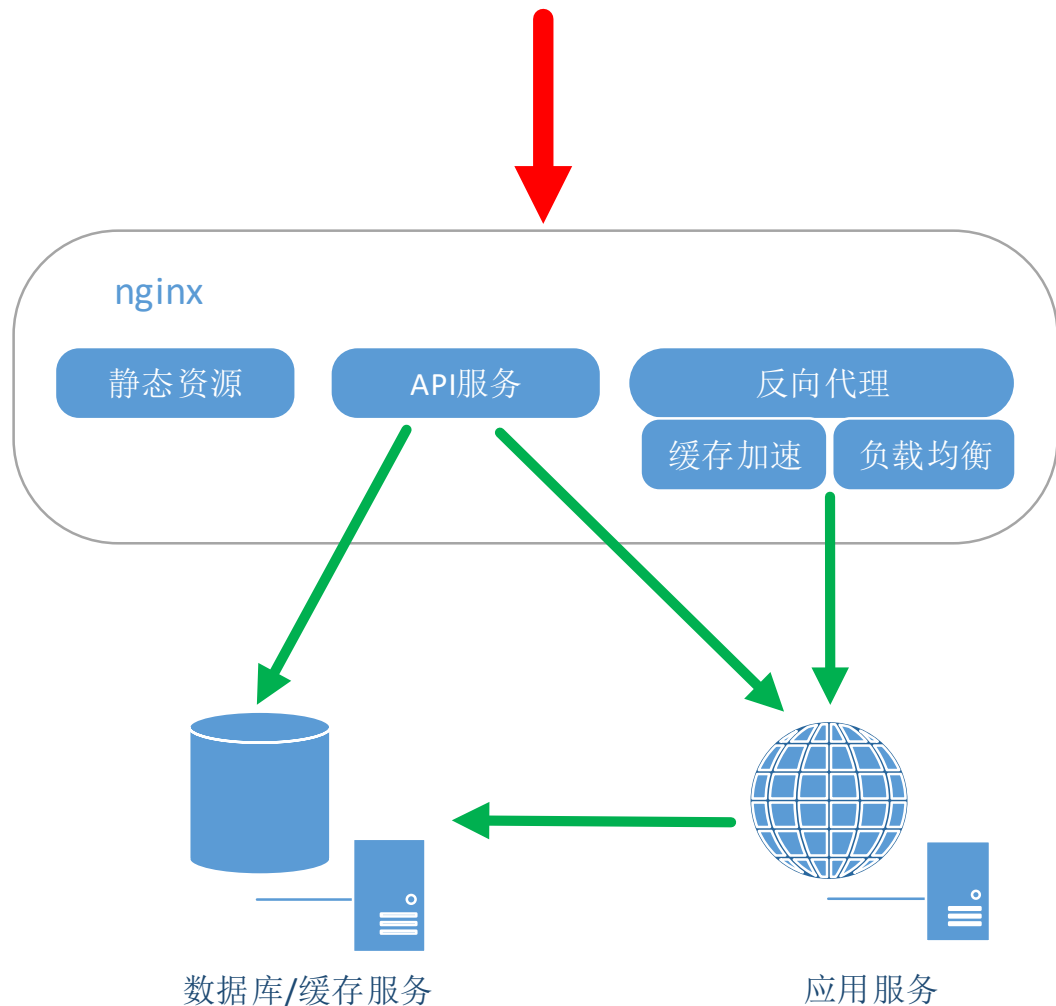


扫码试看/订阅  
《Nginx 核心知识100讲》



# 初识Nginx

# Nginx 的三个主要应用场景



## 静态资源服务

- 通过本地文件系统提供服务

## 反向代理服务

- Nginx的强大性能
- 缓存
- 负载均衡

## API 服务

- OpenResty

# Nginx 为什么会出现？

互联网的数据量快速增长

互联网的快速普及

全球化

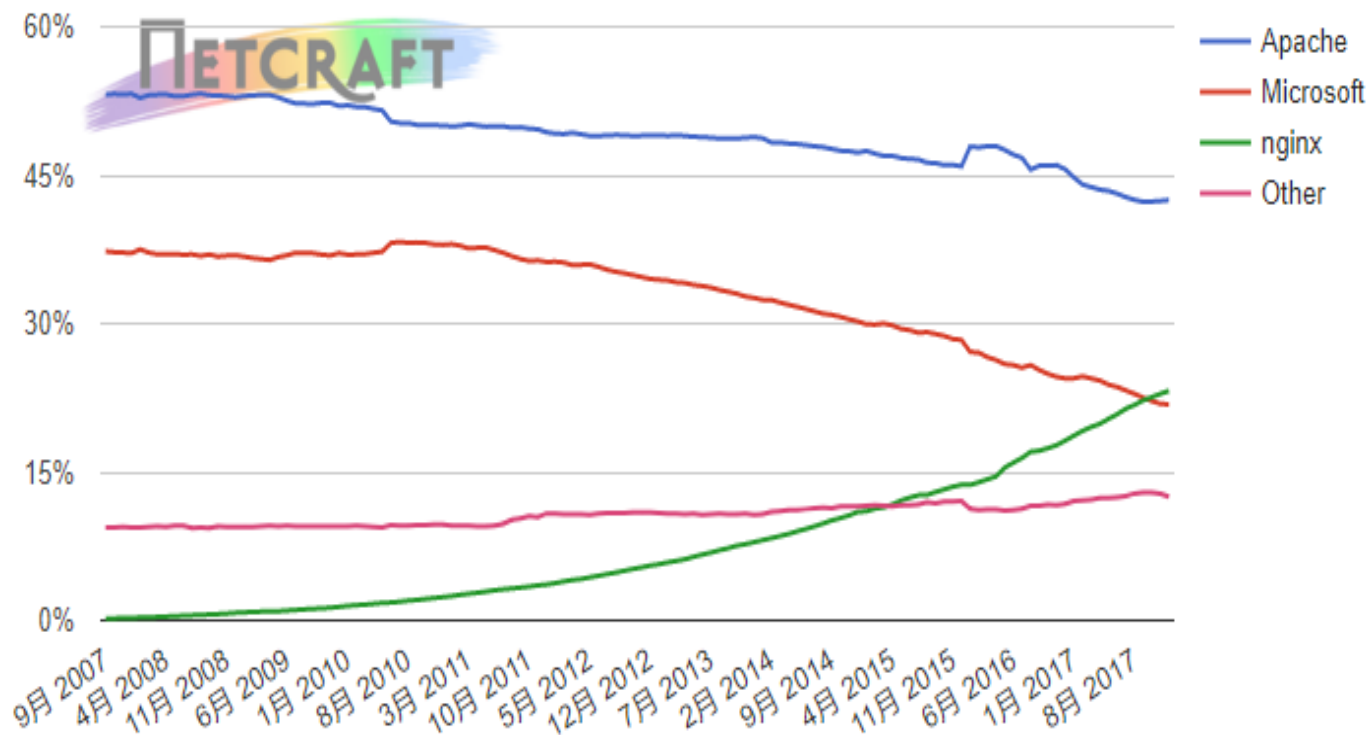
物联网

摩尔定律：性能提升

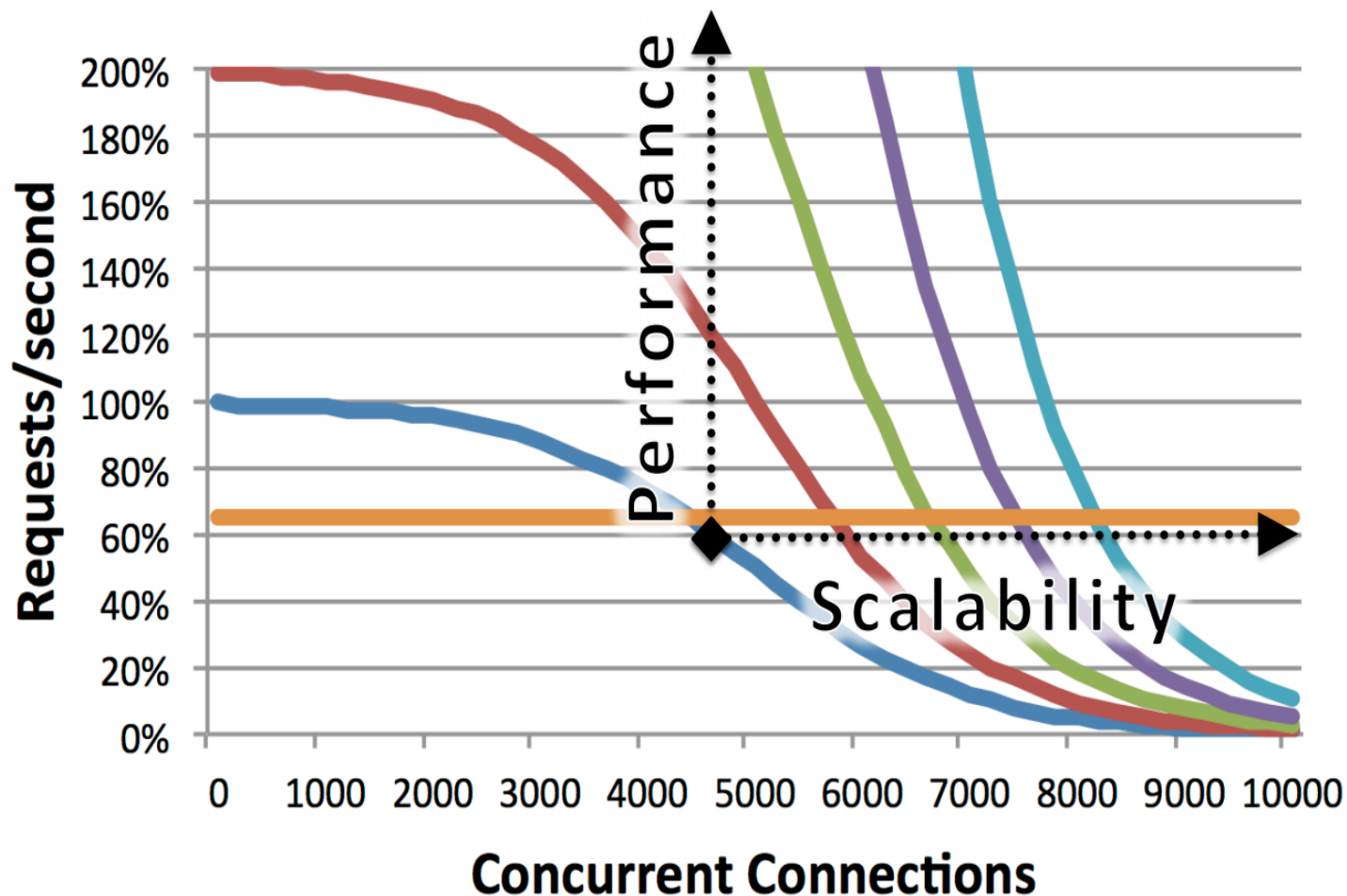
低效的 Apache

一个连接对应一个进程

Web server developers: Market share of computers

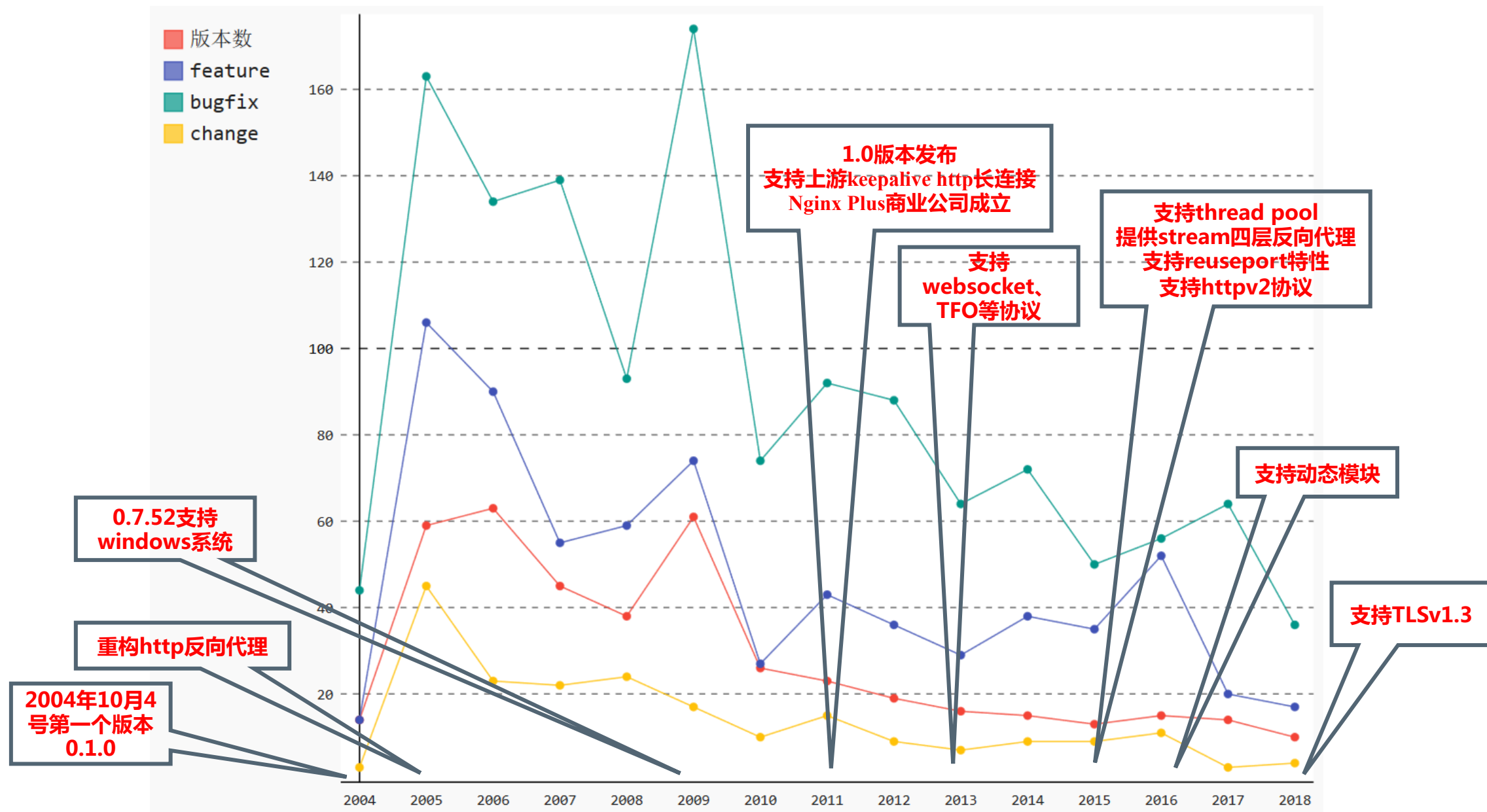


# Nginx 的优点



- ① 高并发，高性能
- ② 可扩展性好
- ③ 高可靠性
- ④ 热部署
- ⑤ BSD 许可证

# Nginx 版本发布情况 ( mainline )



# Nginx 的组成

## Nginx二进制可执行文件

- 由各模块源码编译出的一个文件

## Nginx.conf配置文件

- 控制nginx的行为

## access.log访问日志

- 记录每一条http请求信息

## error.log错误日志

- 定位问题





# 开源免费的 Nginx 与商业版 Nginx Plus

开源版：[nginx.org](http://nginx.org)

商业版：[nginx.com](http://nginx.com)

## Management Team



Gus Robertson  
Chief Executive Officer



Igor Sysoev  
CTO and Co-Founder



Maxim Konovalov  
VP, Engineering and Co-Founder



Mark Elchinoff  
Chief Financial Officer

## 简介

Tengine是由淘宝网发起的Web服务器项目。它在Nginx的基础上，针对大访问量网站的需求，添加了很多高级功能和特性。Tengine的性能和稳定性已经在大型的网站如[淘宝网](#)，[天猫商城](#)等得到了很好的检验。它的最终目标是打造一个高效、稳定、安全、易用的Web平台。

从2011年12月开始，Tengine成为一个开源项目，Tengine团队在积极地开发和维护着它。Tengine团队的核心成员来自于[淘宝](#)、[搜狗](#)等互联网企业。Tengine是社区合作的成果，我们欢迎大家[参与其中](#)，贡献自己的力量。

[下载](#)

# 免费 OpenResty 与商业版 OpenResty

开源 OpenResty : <http://openresty.org>

这个开源 Web 平台主要由章亦春 ([agentzh](#)) 维护。在 2011 年之前曾由[淘宝网](#)赞助，在后来的 2012 ~ 2016 年间主要由美国的 [CloudFlare 公司](#) 提供支持。目前，OpenResty® 主要由 OpenResty 软件基金会和 OpenResty Inc. 公司提供支持。

因为大部分 [Nginx](#) 模块都是由本软件包的维护者开发，所以可以确保所有这些模块及其他组件可以很好地一起工作。

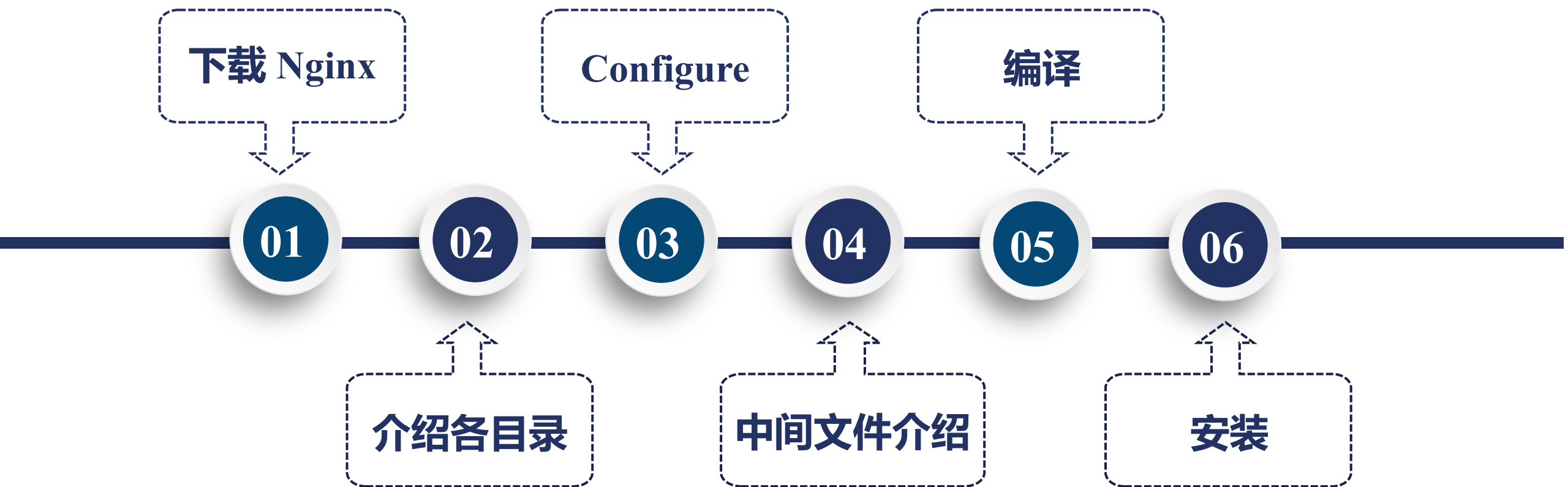
打包的各个软件组件版权属于各自的版权持有者。

商业版 OpenResty : <https://openresty.com>



章亦春  
创始人兼 CEO

OpenResty 开源平台创始人



# Nginx 配置语法

- 01 配置文件由指令与指令块构成
- 02 每条指令以 ; 分号结尾，指令与参数间以空格符号分隔
- 03 指令块以 { } 大括号将多条指令组织在一起
- 04 include 语句允许组合多个配置文件以提升可维护性
- 05 使用 # 符号添加注释，提高可读性
- 06 使用 \$ 符号使用变量
- 07 部分指令的参数支持正则表达式

# Nginx 语法示例

```
http {
    include      mime.types;
    upstream thwp {
        server 127.0.0.1:8000;
    }

    server {
        listen 443 http2;
        #Nginx配置语法
        limit_req_zone $binary_remote_addr zone=one:10m rate=1r/s;
        location ~* \.(gif|jpg|jpeg)$ {
            proxy_cache my_cache;
            expires 3m; proxy_cache_key $host$uri$is_args$args;
            proxy_cache_valid 200 304 302 1d;
            proxy_pass http://thwp;
        }
    }
}
```

# 配置参数：时间的单位

**ms**

**milliseconds**

**d**

**days**

**s**

**seconds**

**w**

**weeks**

**m**

**minutes**

**M**

**months, 30 days**

**h**

**hours**

**y**

**years, 365 days**

# Nginx 语法示例

```
http {
    include      mime.types;
    upstream thwp {
        server 127.0.0.1:8000;
    }

    server {
        listen 443 http2;
        #Nginx配置语法
        limit_req_zone $binary_remote_addr zone=one:10m rate=1r/s;
        location ~* \.(gif|jpg|jpeg)$ {
            proxy_cache my_cache;
            expires 3m;proxy_cache_key $host$uri$is_args$args;
            proxy_cache_valid 200 304 302 1d;
            proxy_pass http://thwp;
        }
    }
}
```



# 配置参数：空间的单位

bytes

k/K

kilobytes

m/M

megabytes

g/G

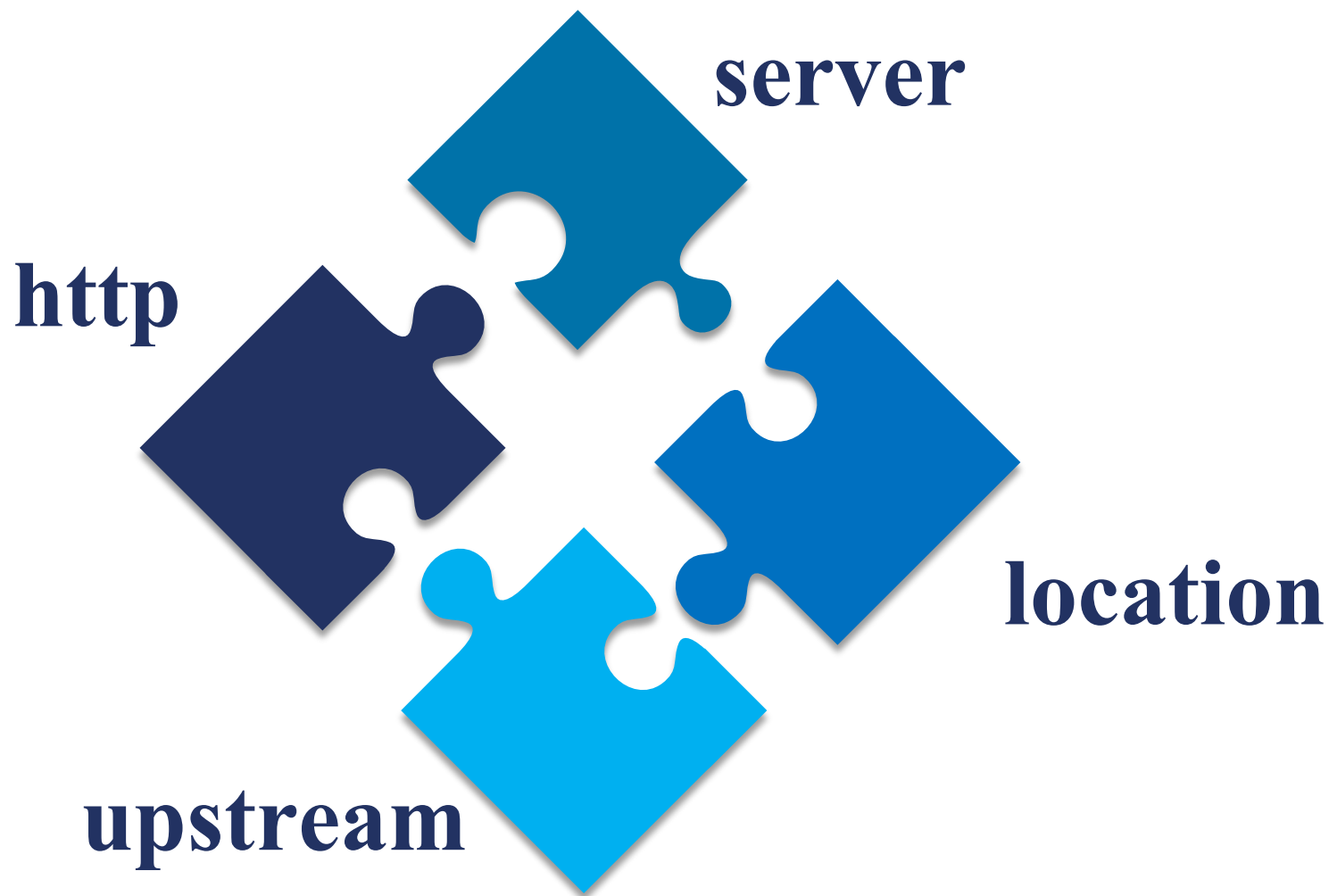
gigabytes

# Nginx 语法示例

```
http {
    include      mime.types;
    upstream thwp {
        server 127.0.0.1:8000;
    }

    server {
        listen 443 http2;
        #Nginx配置语法
        limit_req_zone $binary_remote_addr zone=one:10m rate=1r/s;
        location ~* \.(gif|jpg|jpeg)$ {
            proxy_cache my_cache;
            expires 3m;proxy_cache_key $host$uri$is_args$args;
            proxy_cache_valid 200 304 302 1d;
            proxy_pass http://thwp;
        }
    }
}
```

# http 配置的指令块



# Nginx 语法示例

```
http {
    include      mime.types;
    upstream thwp {
        server 127.0.0.1:8000;
    }

    server {
        listen 443 http2;
        #Nginx配置语法
        limit_req_zone $binary_remote_addr zone=one:10m rate=1r/s;
        location ~* \.(gif|jpg|jpeg)$ {
            proxy_cache my_cache;
            expires 3m;proxy_cache_key $host$uri$is_args$args;
            proxy_cache_valid 200 304 302 1d;
            proxy_pass http://thwp;
        }
    }
}
```

# Nginx 命令行

01 > 格式：nginx -s reload

02 帮助：-? -h

03 > 使用指定的配置文件：-c

04 指定配置指令：-g

05 > 指定运行目录：-p

06 发送信号：-s

07 > 测试配置文件是否有语法错误：-t -T

08 打印nginx的版本信息、编译信息等：-v -V

立刻停止服务：stop  
优雅的停止服务：quit  
重载配置文件：reload  
重新开始记录日志文件：  
reopen

重载配置  
文件

热部署

切割日  
志文件

# 静态资源服务演示

01

http

server

02

03

location

alias

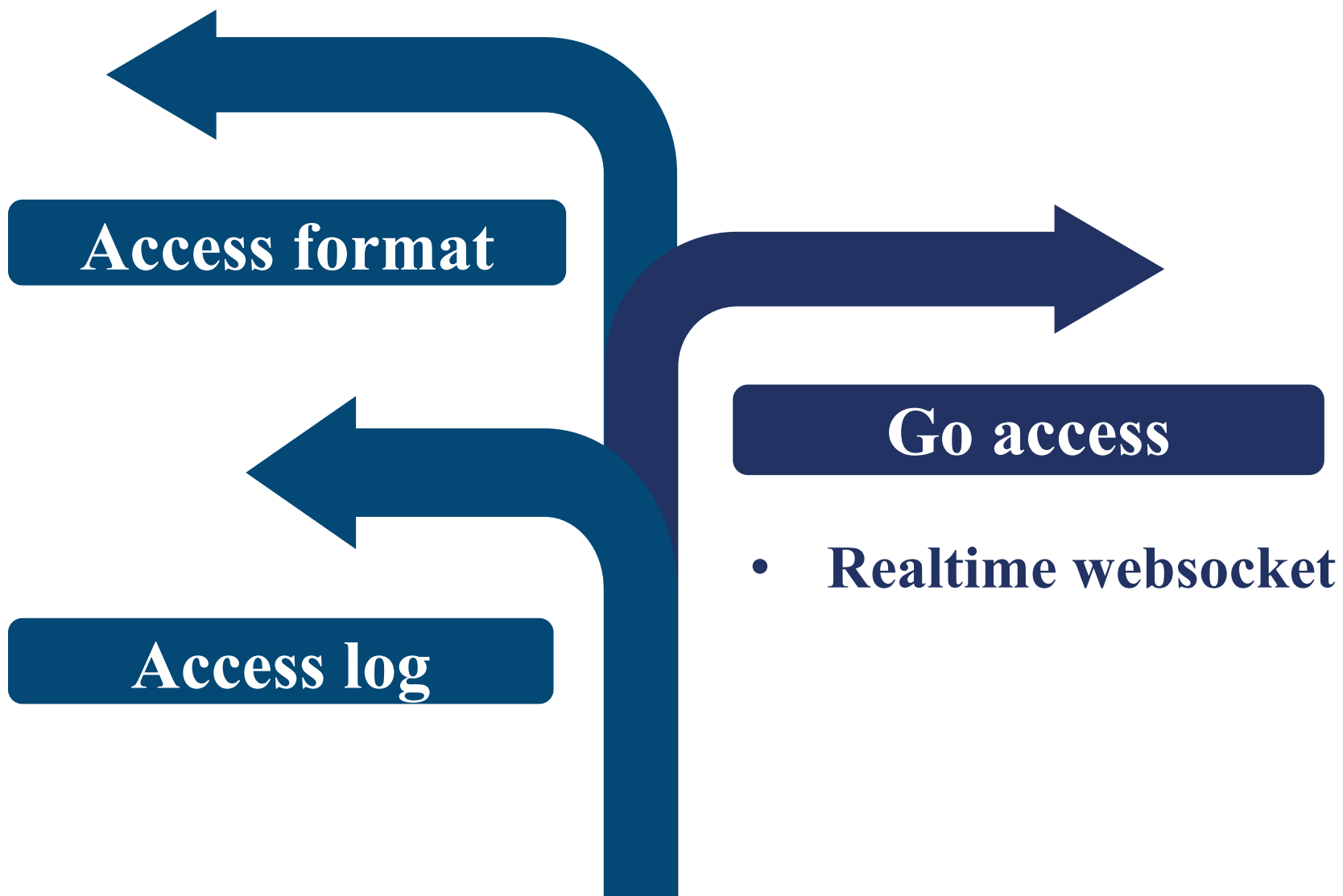
04

05

gzip

thread pool

06

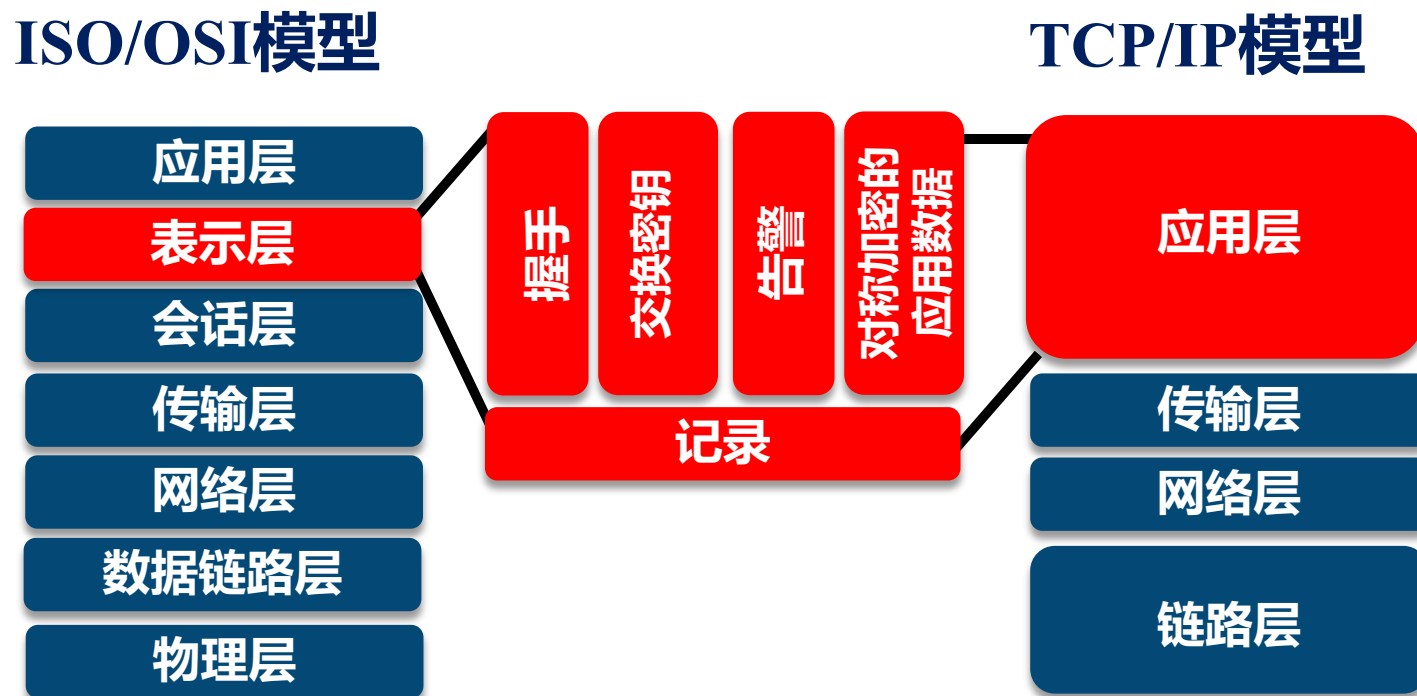




# TLS/SSL 发展

- 1995 SSL3.0
- 1999 TLS1.0
- 2006 TLS1.1
- 2008 TLS1.2
- 2018 TLS1.3

## SSL/TLS通用模型



SSL(Secure Sockets Layer)  
TLS(Transport Layer Security)

# TLS 安全密码套件解读



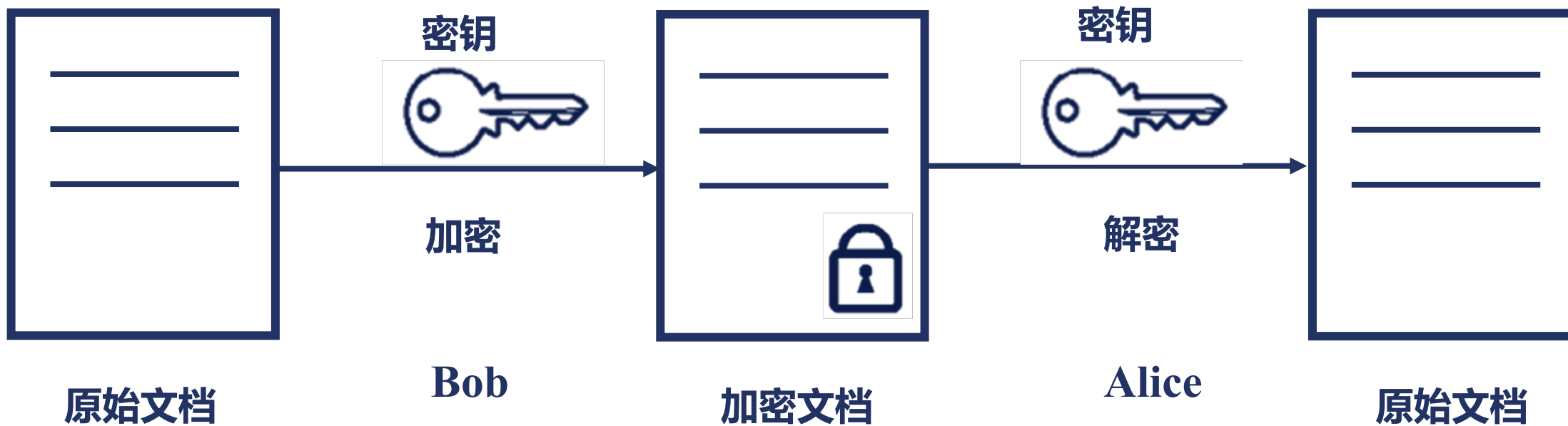
密钥交换算法

身份验证算法

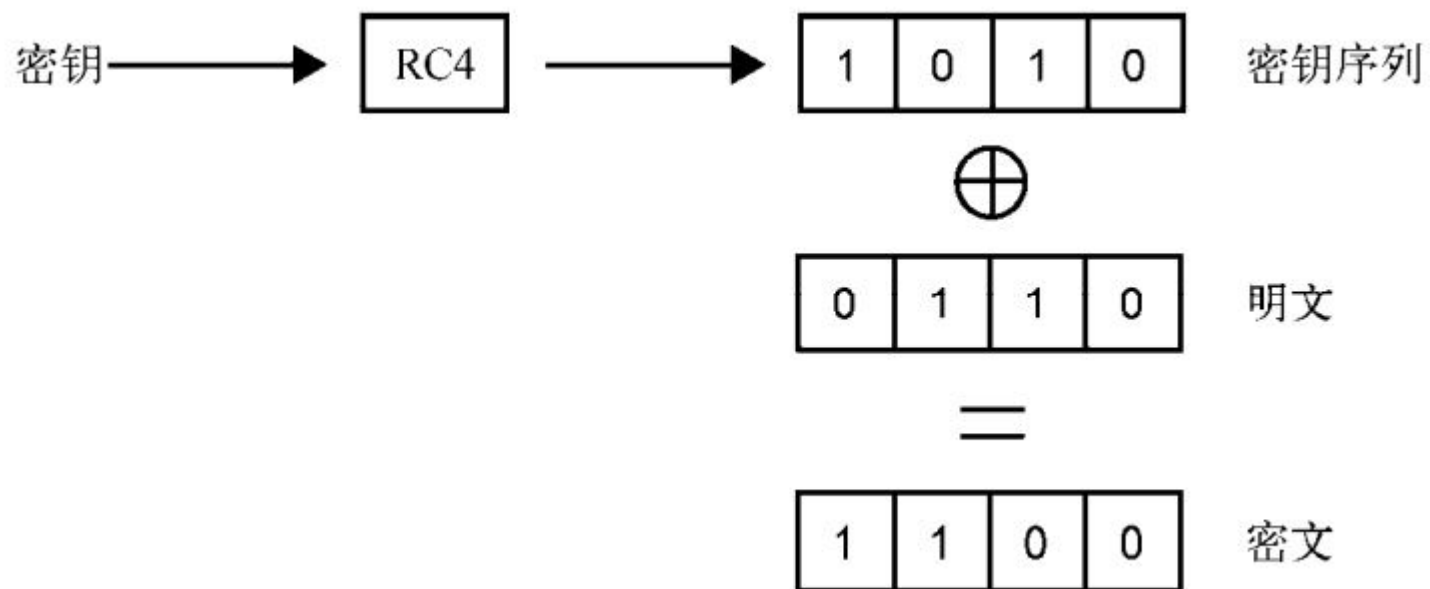
对称加密算法、强度、分组模式

签名hash算法

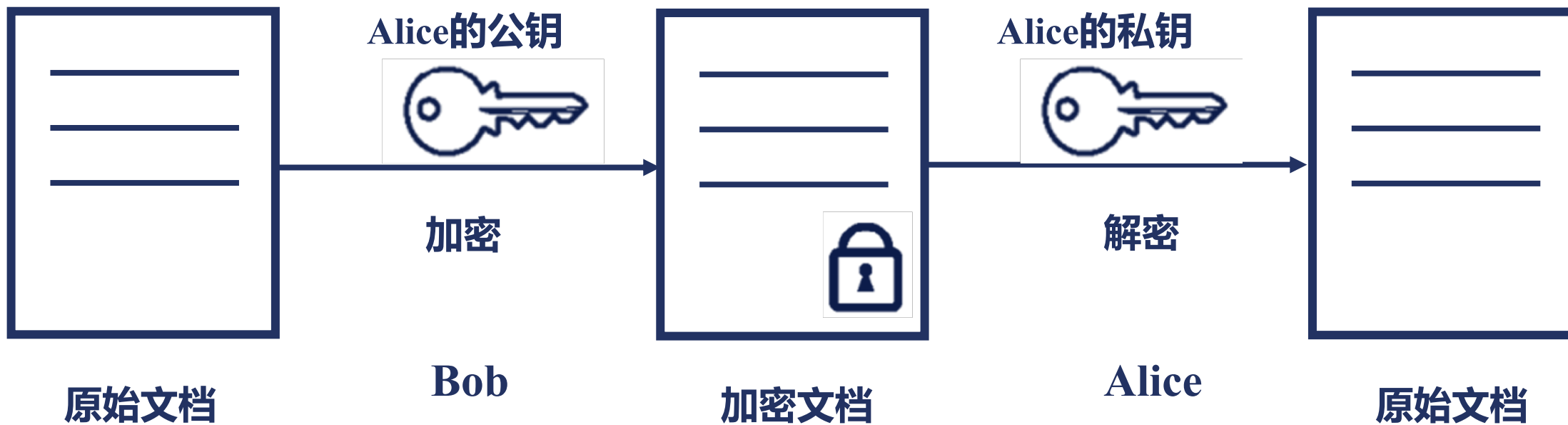
# 对称加密



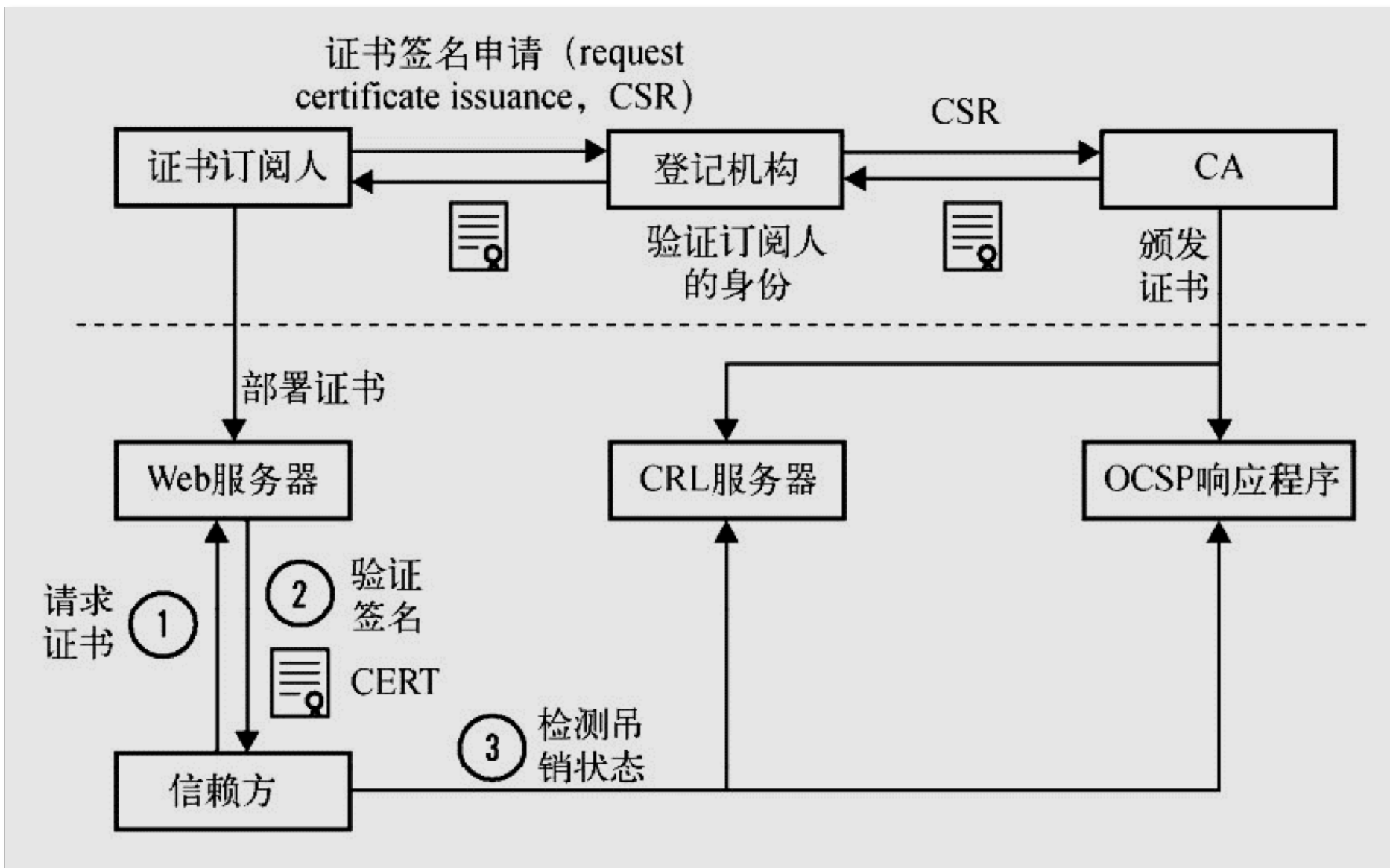
# 对称加密



# 非对称加密



# PKI 公钥基础设施



## 域名验证 ( domain validated , DV ) 证书

  <https://www.taohui.pub>

## 组织验证 ( organization validated , OV ) 证书

  <https://www.jd.com>

## 扩展验证 ( extended validation , EV ) 证书

  浙江名友金融信息服务有限公司 (CN) | <https://www.mingyou.com>

# 证书链

证书查看器: "www.taohui.pub"

基本信息(G) 详细信息(D)

## 证书层次结构(H)

√ DigiCert Global Root CA  
    √ Encryption Everywhere DV TLS CA - G1  
        www.taohui.pub

## 证书字段(F)

√ www.taohui.pub  
    √ 证书  
        版本  
        序列号  
        证书签名算法  
        颁发者  
    √ 有效性  
        不早于

## 字段值(V)

CN = Encryption Everywhere DV TLS CA - G1  
OU = www.digicert.com  
O = DigiCert Inc  
C = US

```
Transmission Control Protocol, Src Port: 443, Dst Port: 10007, Seq: 1453, Ack: 518, Len: 1452
[2 Reassembled TCP Segments (2646 bytes): #6(1377), #7(1269)]
Secure Sockets Layer
  TLSv1.2 Record Layer: Handshake Protocol: Certificate
    Content Type: Handshake (22)
    Version: TLS 1.2 (0x0303)
    Length: 2641
  Handshake Protocol: Certificate
    Handshake Type: Certificate (11)
    Length: 2637
    Certificates Length: 2634
  Certificates (2634 bytes)
    Certificate Length: 1430
    Certificate: 308205923082047aa00302010202100ad501af4f3ba9fc98... (id-at-commonName=www.taohui.pub)
      signedCertificate
      algorithmIdentifier (sha256WithRSAEncryption)
      Padding: 0
      encrypted: 12bcbca78e544cca647fdc8c68231a5fc14cef5a799356f7...
      Certificate Length: 1198
    Certificate: 308204aa30820392a00302010202100279ac458bc1b245ab... (id-at-commonName=Encryption Everywhere DV TLS CA - G1)
      signedCertificate
      algorithmIdentifier (sha256WithRSAEncryption)
      Padding: 0
      encrypted: 2b71a9ebf686abb68166cc5ffe843e4c3fc14b05b7014e04...
```



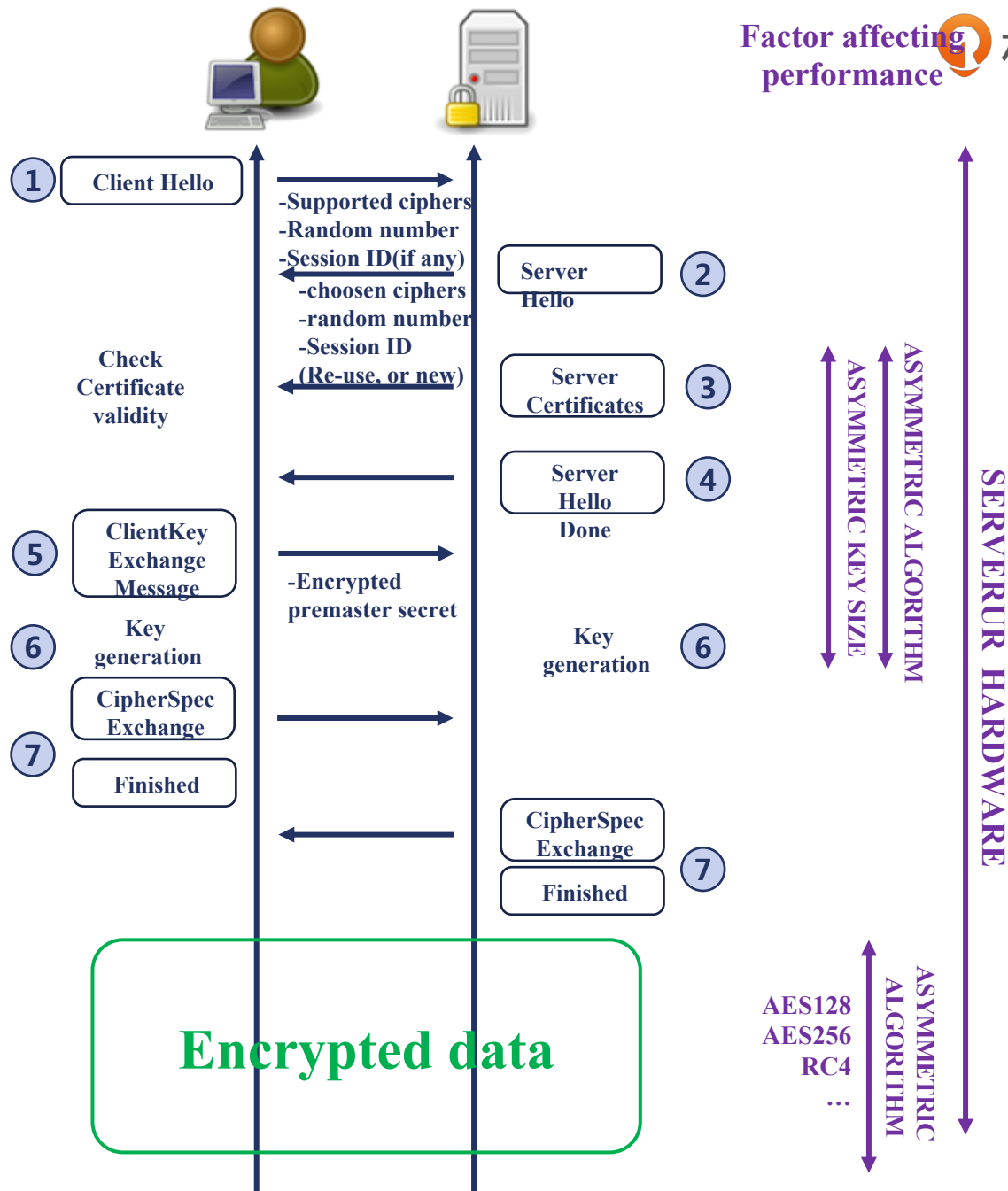
# TLS 通讯过程

验证身份

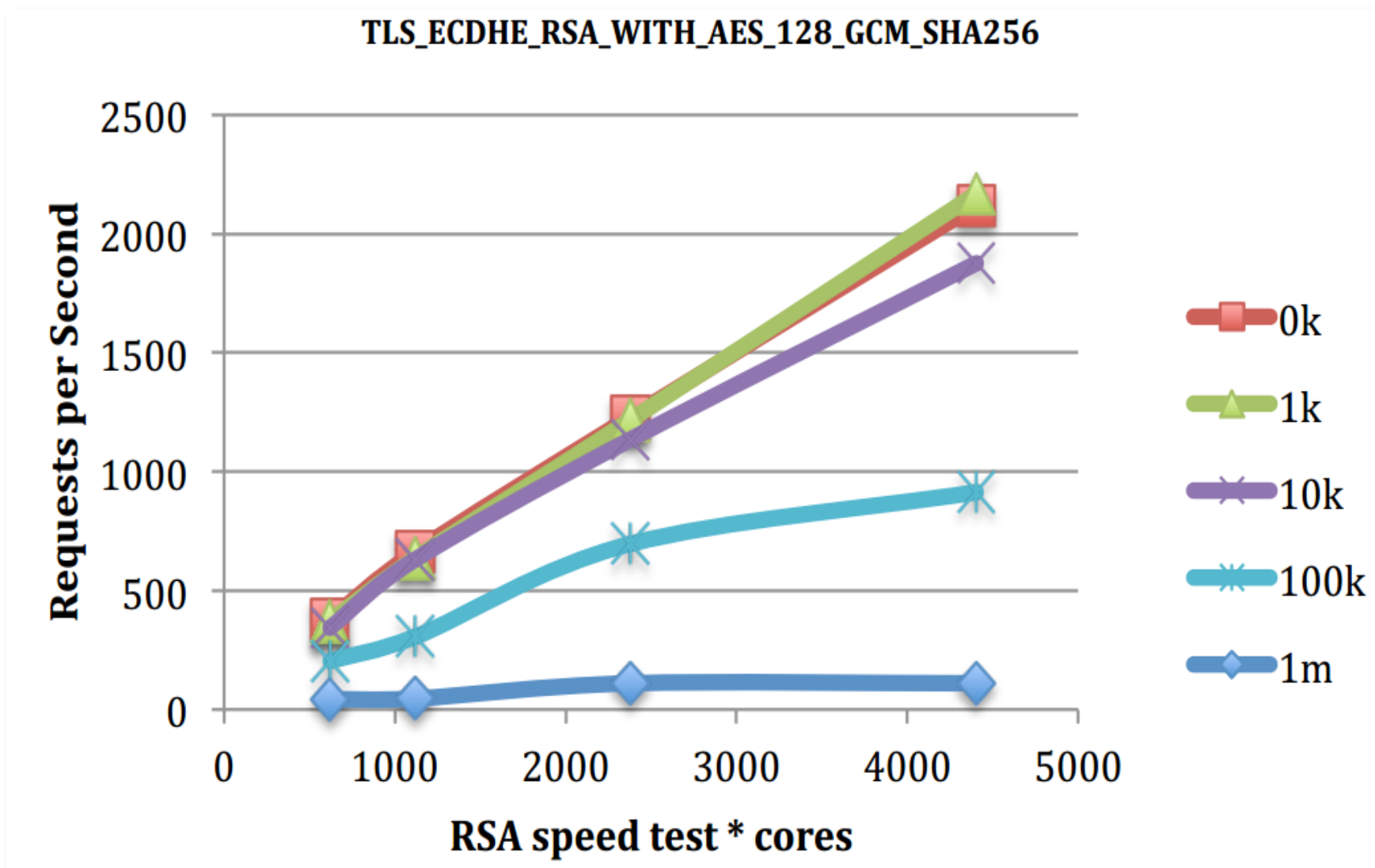
达成安全套件共识

传递密钥

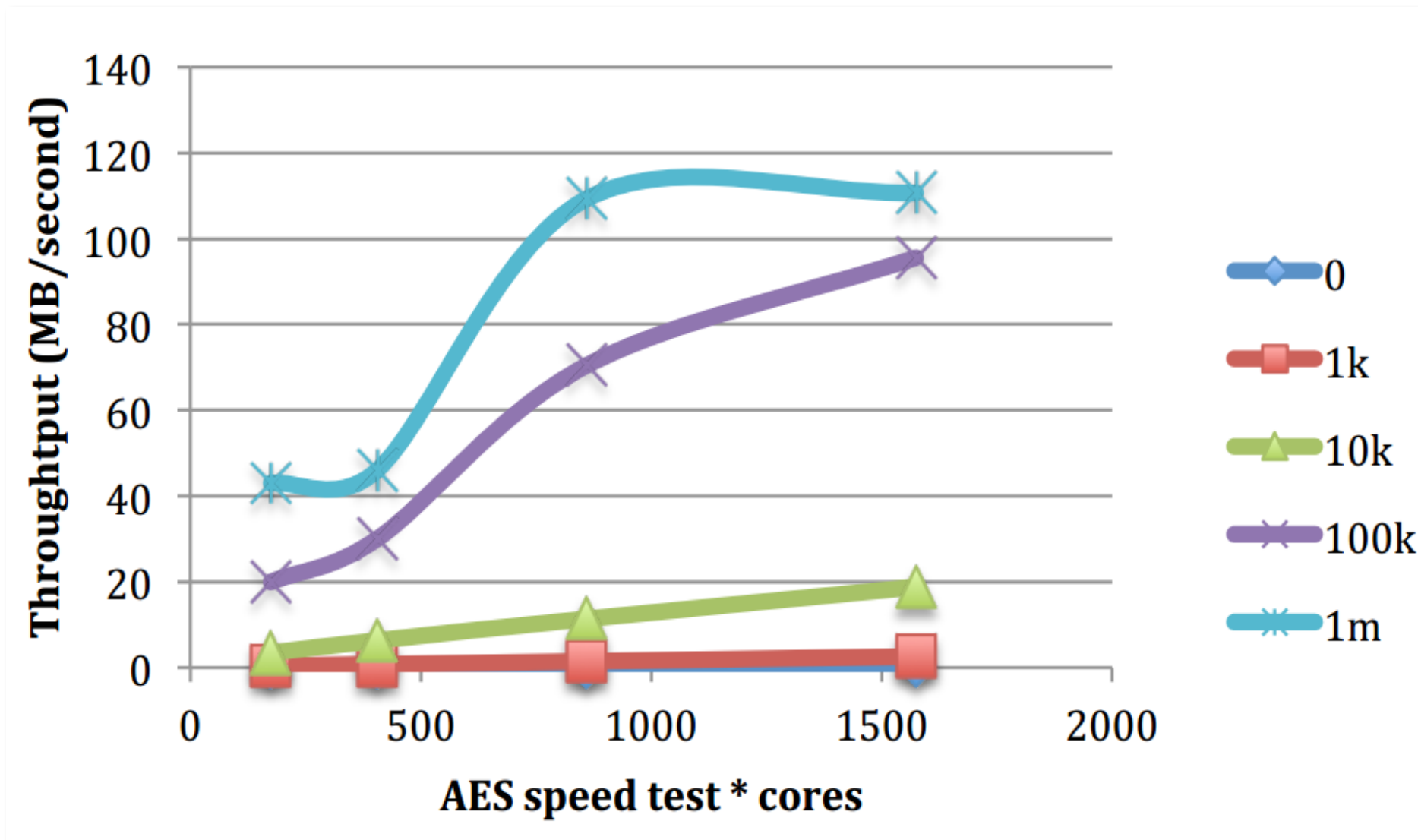
加密通讯



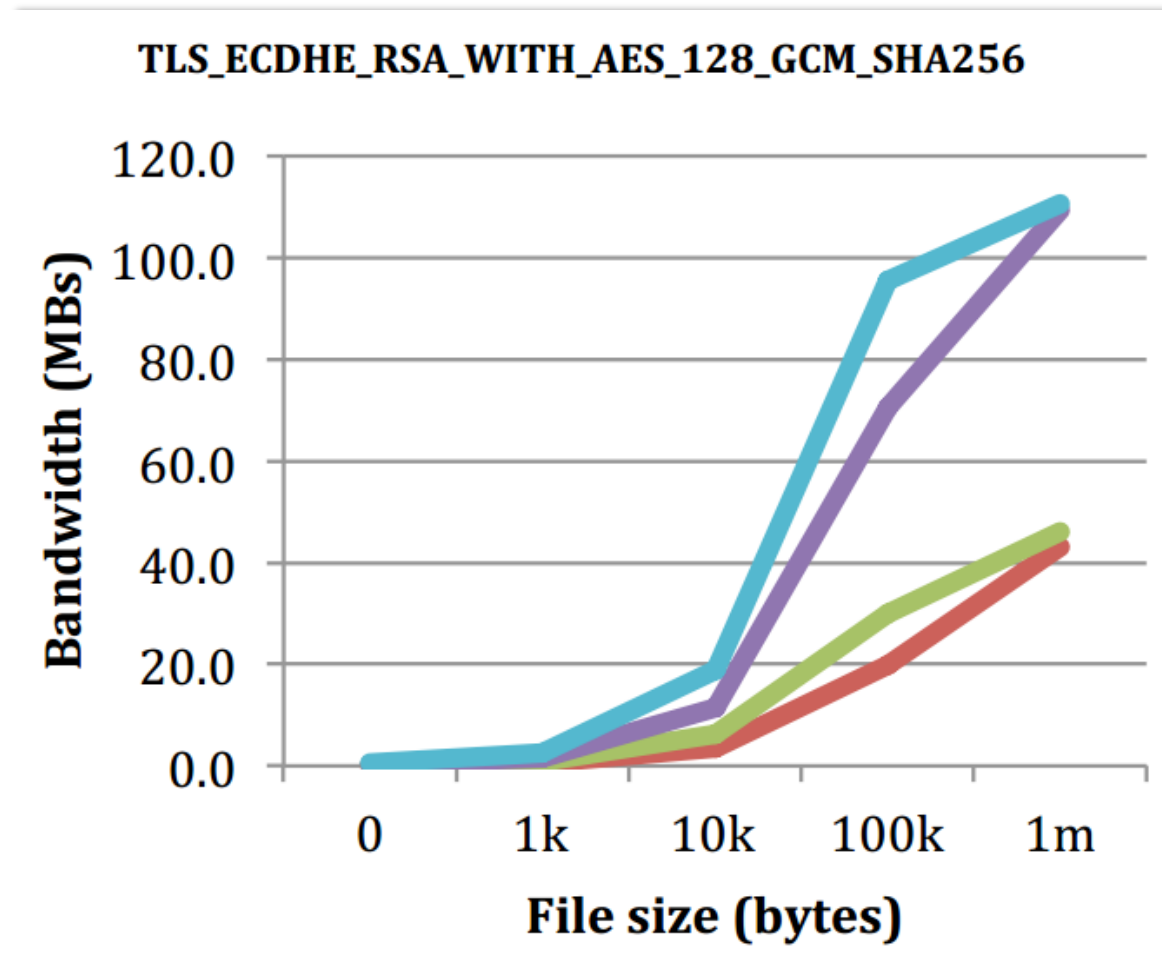
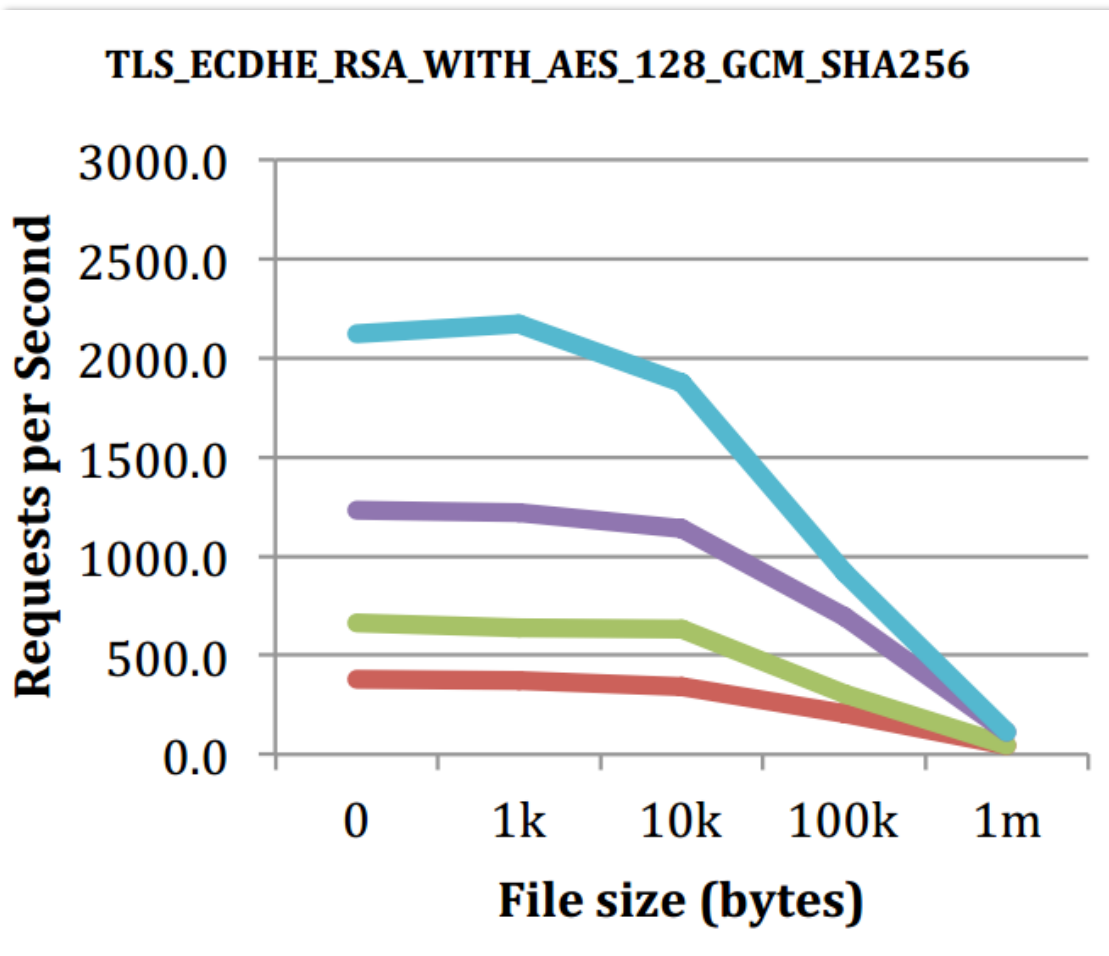
# Nginx 握手性能

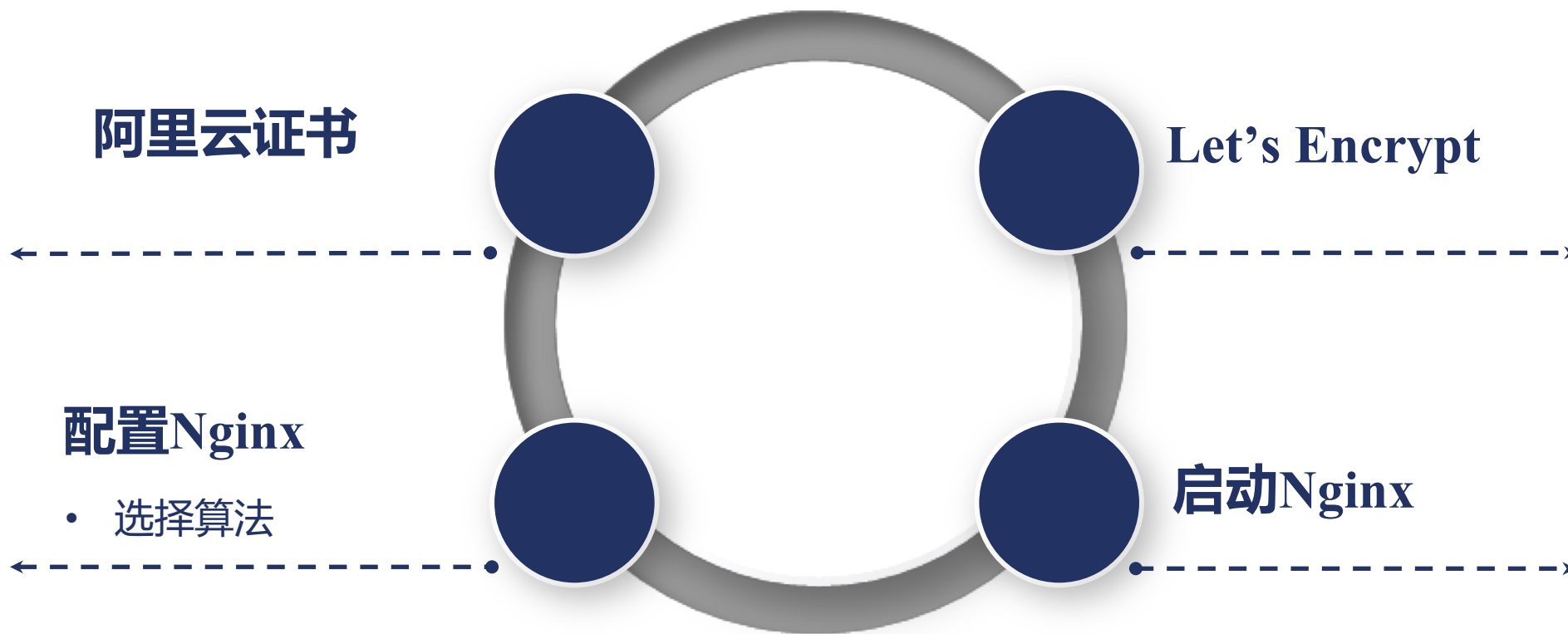


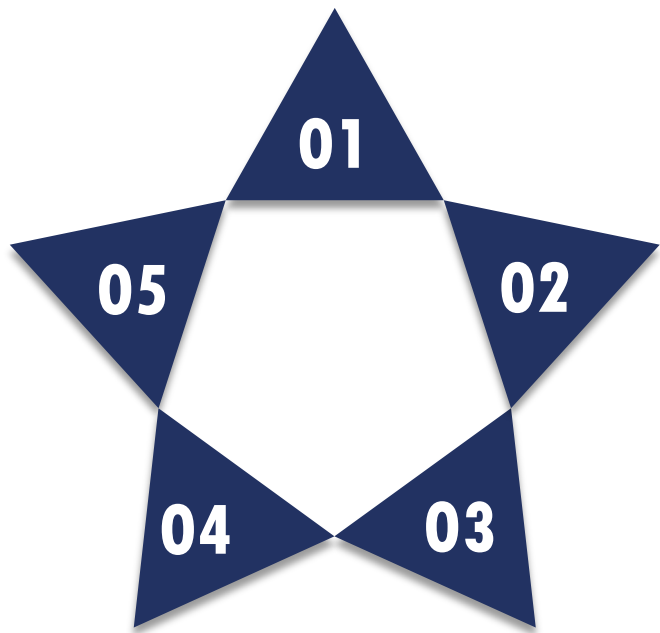
# Nginx 数据加密性能



# Nginx 综合性能







**01 下载 OpenResty**

---

**02 分析目录结构**

---

**03 编译**

---

**04 添加 Lua 代码**

---

**05 运行**

---



扫码试看/订阅  
《Nginx 核心知识100讲》