跟我做一个Java微服务项目

Week #5 安全加固 / 刘俊强

欢迎关注StuQ公众号

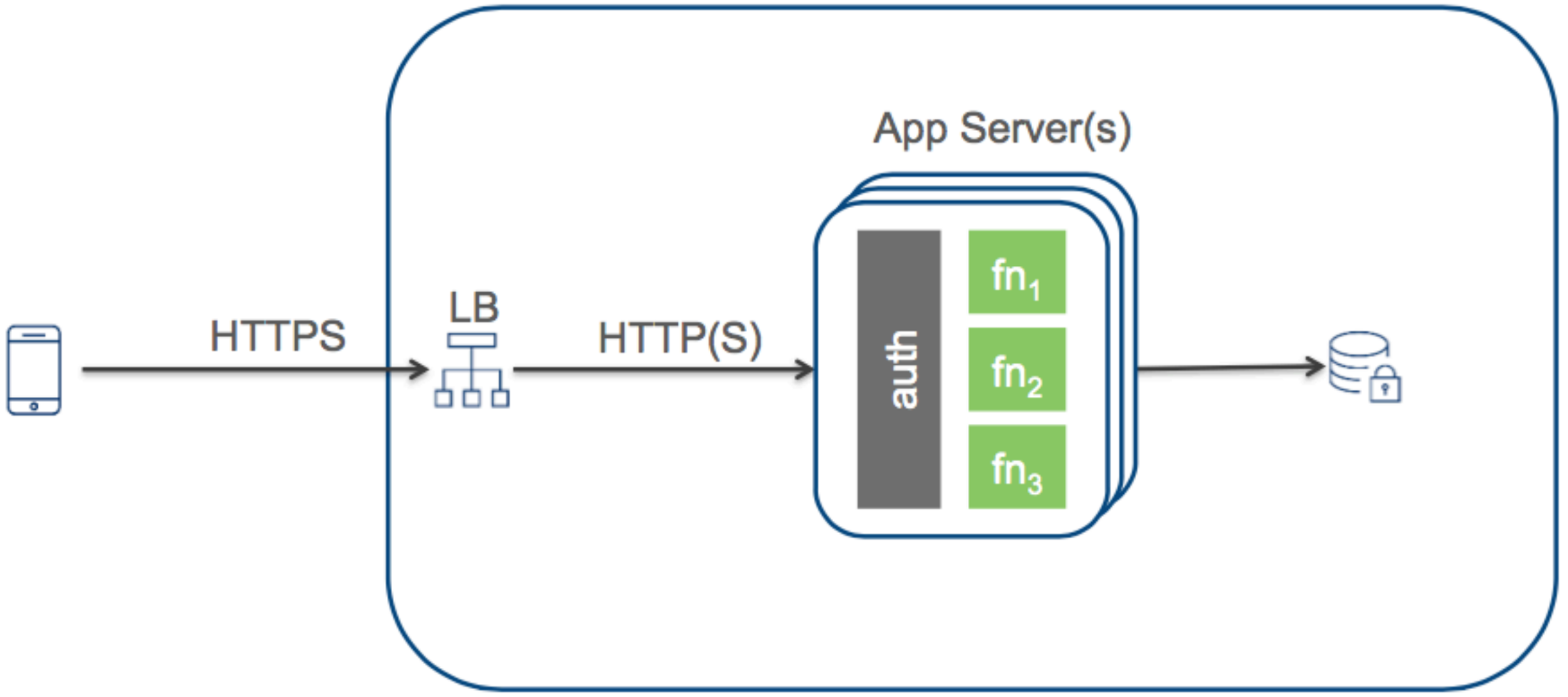
欢迎关注我的微信公众号

大纲

- 微服务的安全挑战
- 微服务的安全方案
- 安全加固贴士

微服务的安全挑战

单体应用安全加固



每个用户请求都进行身份验证

没有Session的情况

1. 检查用户凭证(登录)
2. 查询用户角色
3. 开始用户Session

有Session的情况

检查Session是否过期

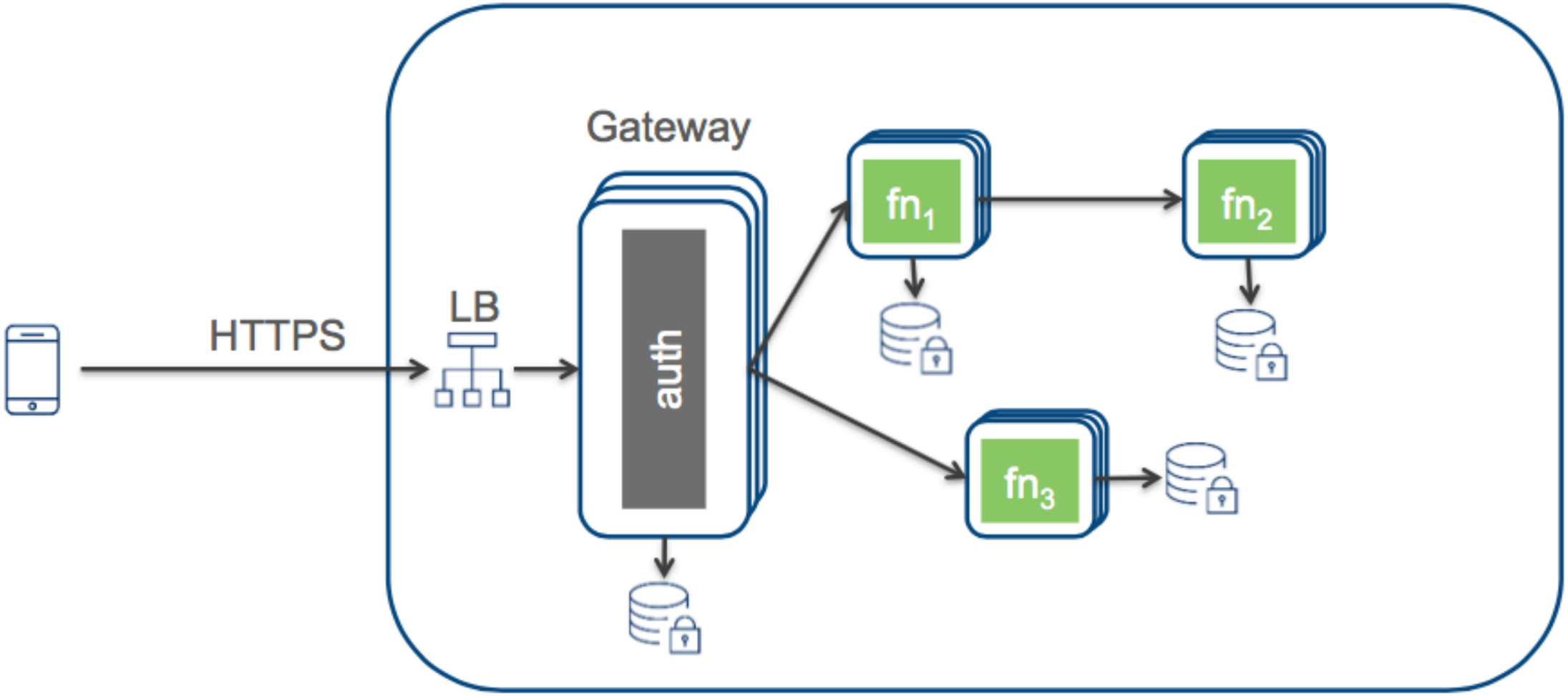
请求、响应都在一个服务进程内处理

方法间调用可被信任

单体应用的身份验证

- 身份验证都是有状态服务（stateful service）
- 移动端App兴起时成为问题
- REST API为无状态

微服务架构安全加固



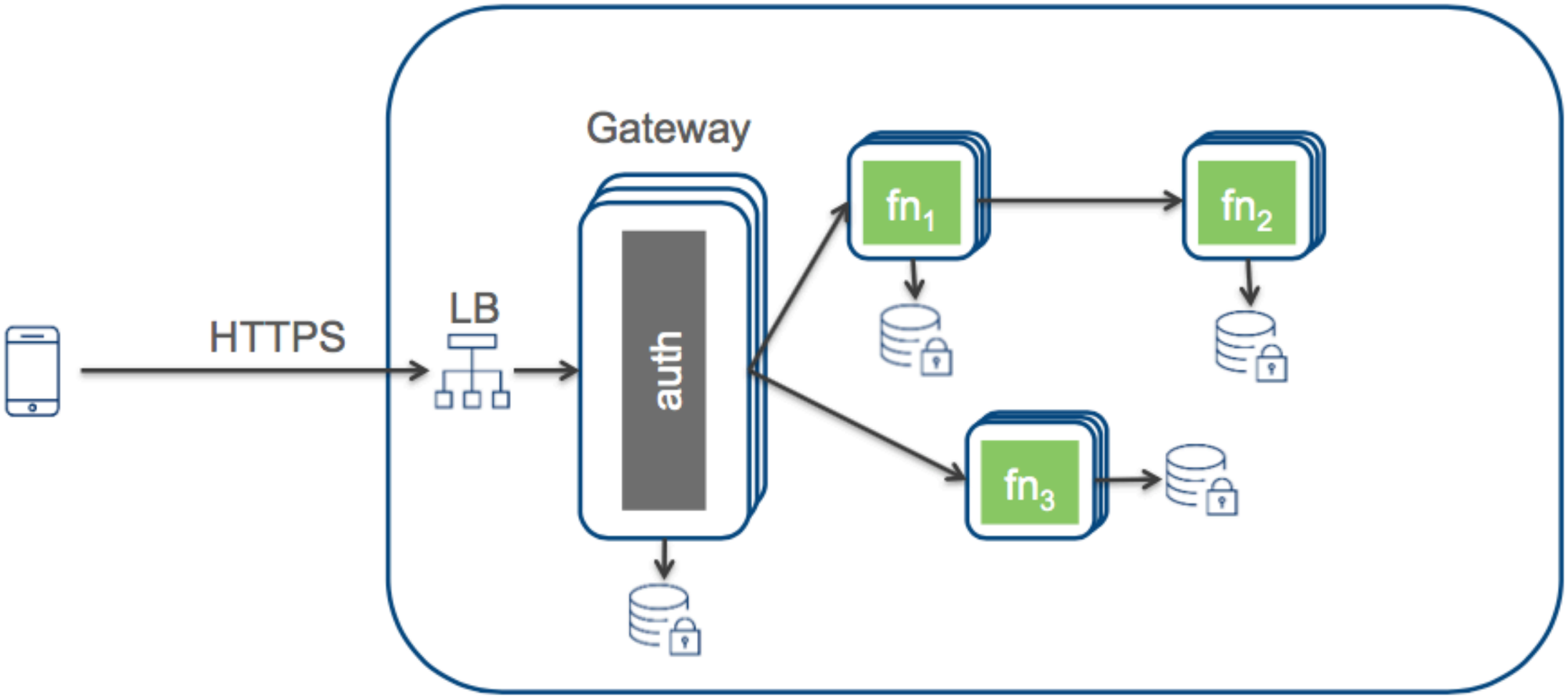
微服务架构安全加固难点

- 更为大的攻击面
- 微服务间如何知道谁在访问?
- 微服务间该如何互相信任?
- 认证服务瓶颈
- 单点登录
- 无状态服务
- SSO
- 无状态
- 对浏览器和客户端友好
- 授权的颗粒度

微服务的身份验证与授权

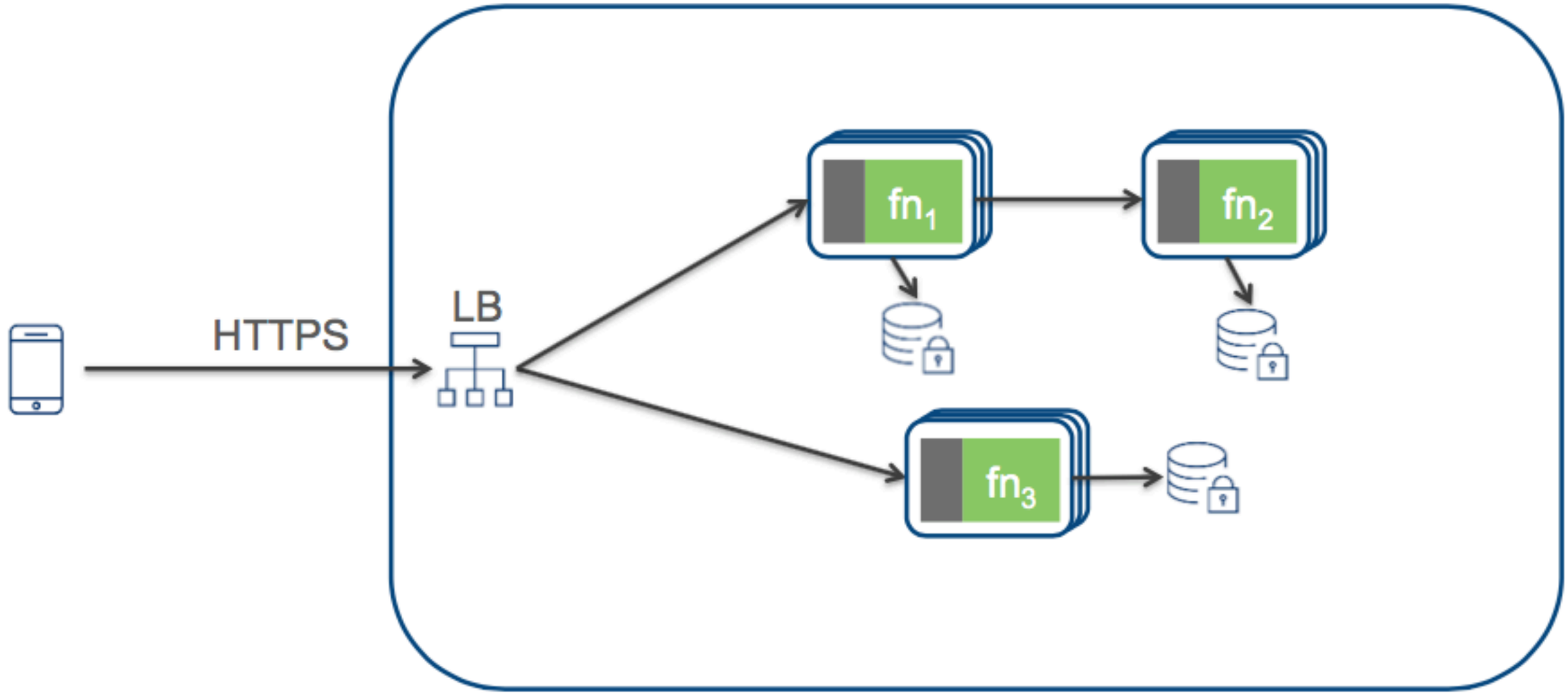
- Authentication 身份验证：对接收到的用户身份凭证进行验证。
- Authorization 角色授权：决定用户是否有访问某些特定资源的权限。
- 在微服务架构里：
  - 身份验证可以是一个单独的微服务
  - 角色授权可以是在微服务里面的功能

微服务的安全方案


Gateway
HTTPS
LB
auth
fn1
fn2
fn3

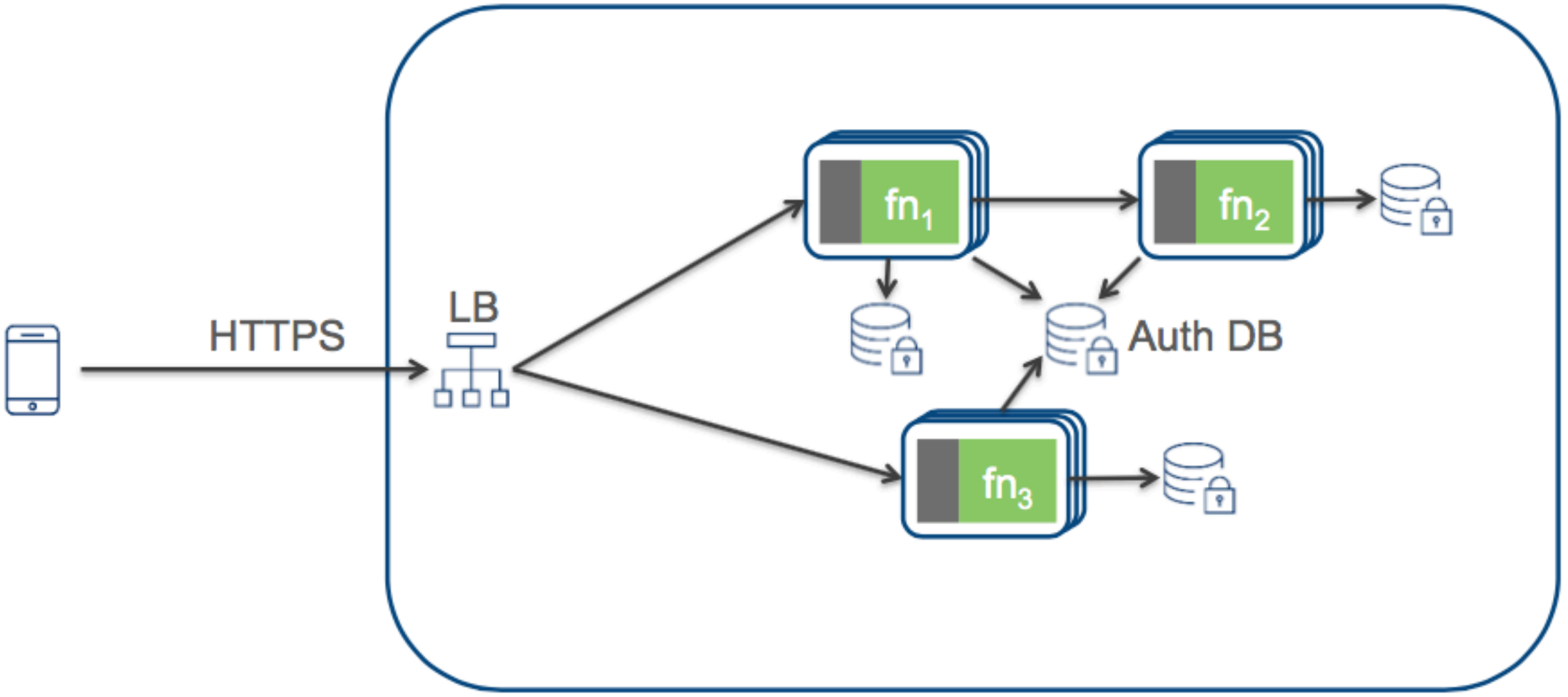
API Gateway Security

- 请求的身份验证和授权由网关进行处理
- 负载均衡器不直接向微服务请求数据
- 服务间相互信任
- 好处：无状态
- 坏处：内部安全威胁


fn1
fn2
fn3
LB
HTTPS

HTTP Basic 每次验证

- 每个服务都进行验证和授权
- 每次在请求头部附加上身份验证信息
- 好处：无状态(每次验证)，接入简单
- 坏处：身份验证信息存储问题、授权管理问题


HTTPS
LB
fn_1
fn_2
fn_3
Auth DB

HTTP Basic + 中心化身份数据库

- 每个服务都进行验证和授权
- 每次在请求头部附加上身份验证信息
- 身份验证信息存储在中心数据库
- 好处：中心化存储、无状态
- 坏处：每次请求就有数据库查询、查询逻辑每个服务都有

每个服务拥有各自Session

- 与上面方案类似，除了每个服务将自行维护session
- 好处：身份验证中心数据库仅在session开始时被请求一次
- 坏处：分散的session难于管理、非单点登录、查询逻辑每个服务都有

API Tokens

- 使用用户名、密码与身份验证服务器通信
- 身份验证服务器将提供token，供其访问服务
- 好处：不用每次提交用户身份信息
- 坏处：身份验证服务器性能瓶颈、权限控制问题

SAML 安全认定标记语言

- 身份验证提供商给应用提供登录服务
- 好处：标准的信任模型
- 坏处：XML过大、对于移动App不友好

OAuth 2.0

OAuth 2.0 是一种给Web、移动或桌面App提供简单、标准且安全的身份授权访问方法的开放协议。

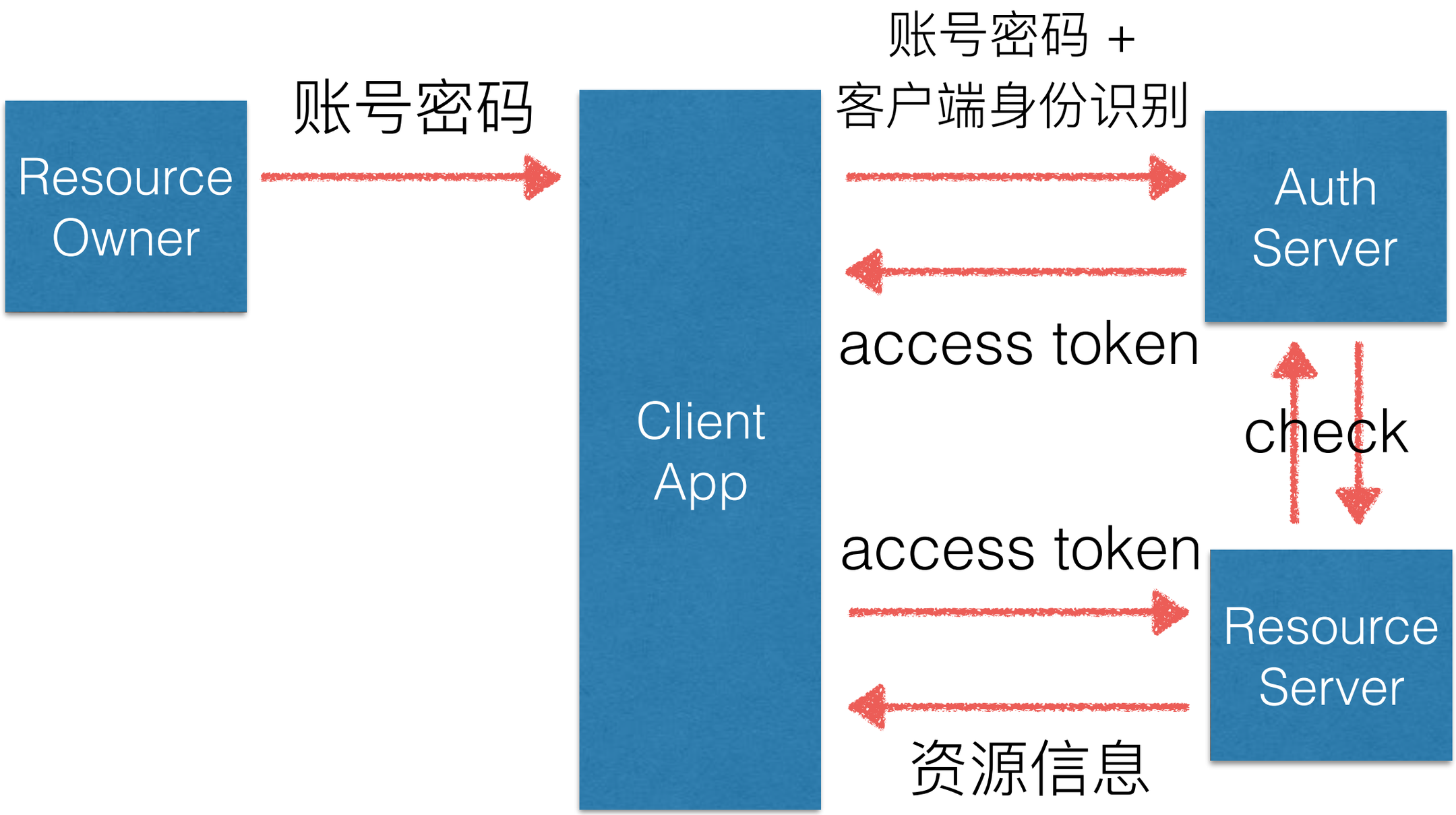
OAuth 2.0 角色

- Resource Owner: **资源所有者**，能够对受保护资源进行授权的实体。例如，终端用户、应用等
- Resource Server: **资源服务器**，持有受保护资源的服务器，使用access token能够接收和响应对受保护资源的请求
- Client: 代表RO去存取受保护资源的应用，指代理人或调用方，不特指客户端
- Authorization Server: **认证服务器**，当成功认证了RO并获得授权后，分发access token给client的服务器

OAuth 2.0 授权模式

- Authorization Code: **授权码模式**
- Implicit: **简化模式** （javascript）
- Resource Owner Password Credentials: **密码模式**
- Client Credentials: **客户端模式**

OAuth 2.0 密码模式



Demo

安全加固贴士

Tip #1: 边际节点进行流量验证，恶意流量不往下传递。

防火墙、流量限制等

Tip #2: 保证数据私密性

- 不要明文传递密码
- 保护私钥
- 使用数据加密技术
- 安全保存密码：加盐hash

Tip #3: 小心日志

Tip #4: 服务间隔离，使用访问控制

Tip #5: 配置、公私钥等存储要分开

StuQ 斯达克学院

THANKS!

— 扫码了解更多 —