

# 编码、加密、Hash

HenCoder Plus

扔物线

# 涉及内容

- 编码解码 Encoding & Decoding
- 加密解密 Encryption & Decryption
- 数字签名 Digital Signature
- 压缩与解压缩 Compression & Decompression
- 序列化 Serialization
- 哈希 Hash
- 字符集 Charset

# 为什么讲这些

- 因为这些东西经常会被用到，但需要真正理解才能正确使用
- 相关词：MD5、SHA1、RSA、DSA、AES、BASE64、encoded URL、Unicode、UTF-8、GBK、ISO-8859-1.....

# 密码学

# 密码学

- 起源：古代战争

# 密码学

- 起源：古代战争——古典密码学

# 密码学

- 起源：古代战争——古典密码学
- 移位式加密：密码棒

# 密码棒





# 密码学

- 起源：古代战争——古典密码学
- 移位式加密：密码棒

# 密码学

- 起源：古代战争——古典密码学
- 移位式加密：密码棒
  - 加密算法：缠绕木棒后书写

# 密码学

- 起源：古代战争——古典密码学
- 移位式加密：密码棒
  - 加密算法：缠绕木棒后书写
  - 密钥：木棒的尺寸规格

# 密码学

- 起源：古代战争——古典密码学
- 移位式加密：密码棒
  - 加密算法：缠绕木棒后书写
  - 密钥：木棒的尺寸规格
- 替换式加密

# 密码学

- 起源：古代战争——古典密码学
- 移位式加密：密码棒
  - 加密算法：缠绕木棒后书写
  - 密钥：木棒的尺寸规格
- 替换式加密
  - 加密算法：替换文字

# 密码学

- 起源：古代战争——古典密码学
- 移位式加密：密码棒
  - 加密算法：缠绕木棒后书写
  - 密钥：木棒的尺寸规格
- 替换式加密
  - 加密算法：替换文字
  - 密钥：码表

# 现代密码学

# 现代密码学

- 不止可以用于文字内容，还可以用于各种二进制数据。



# 现代密码学

- 不止可以用于文字内容，还可以用于各种二进制数据。
- 对称加密

# 现代密码学

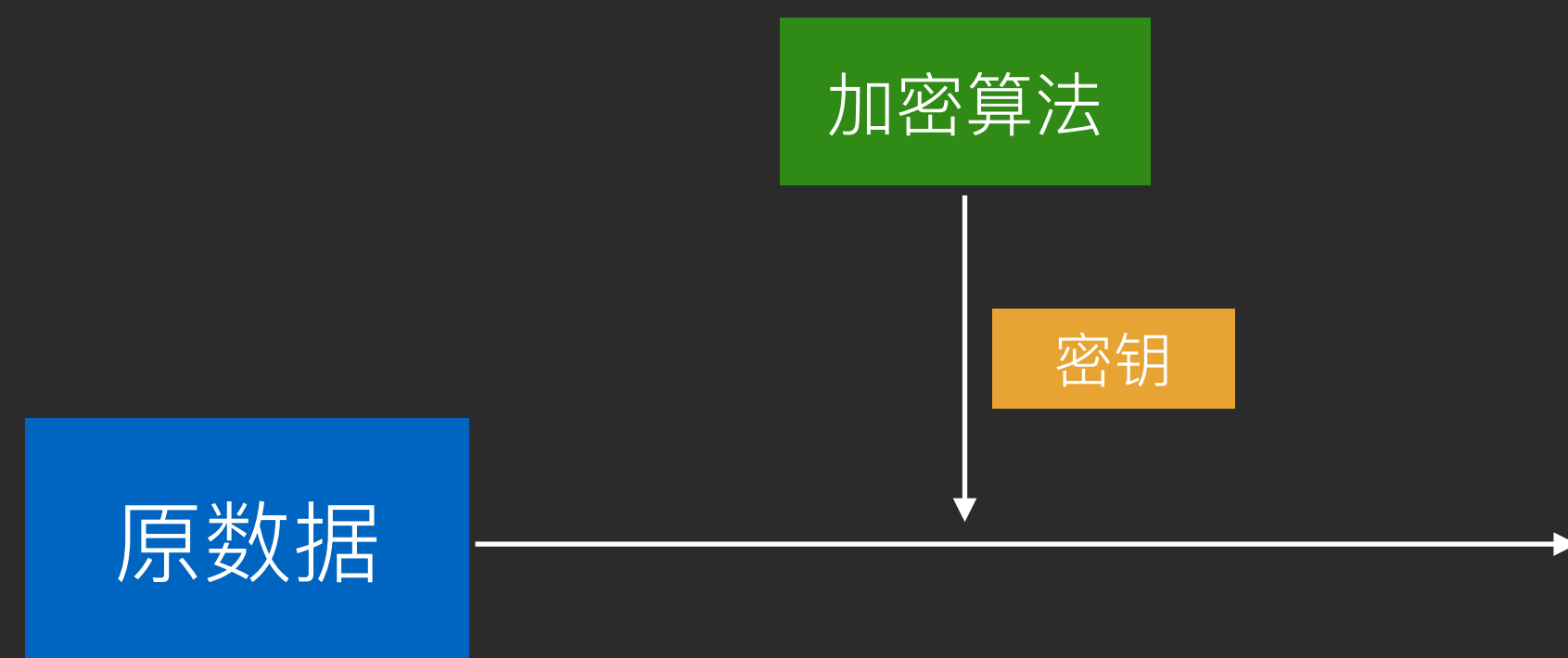
- 不止可以用于文字内容，还可以用于各种二进制数据。
- 对称加密
  - 原理：使用密钥和加密算法对数据进行转换，得到的无意义数据即为密文；使用密钥和解密算法对密文进行逆向转换，得到原数据。

# 对称加密

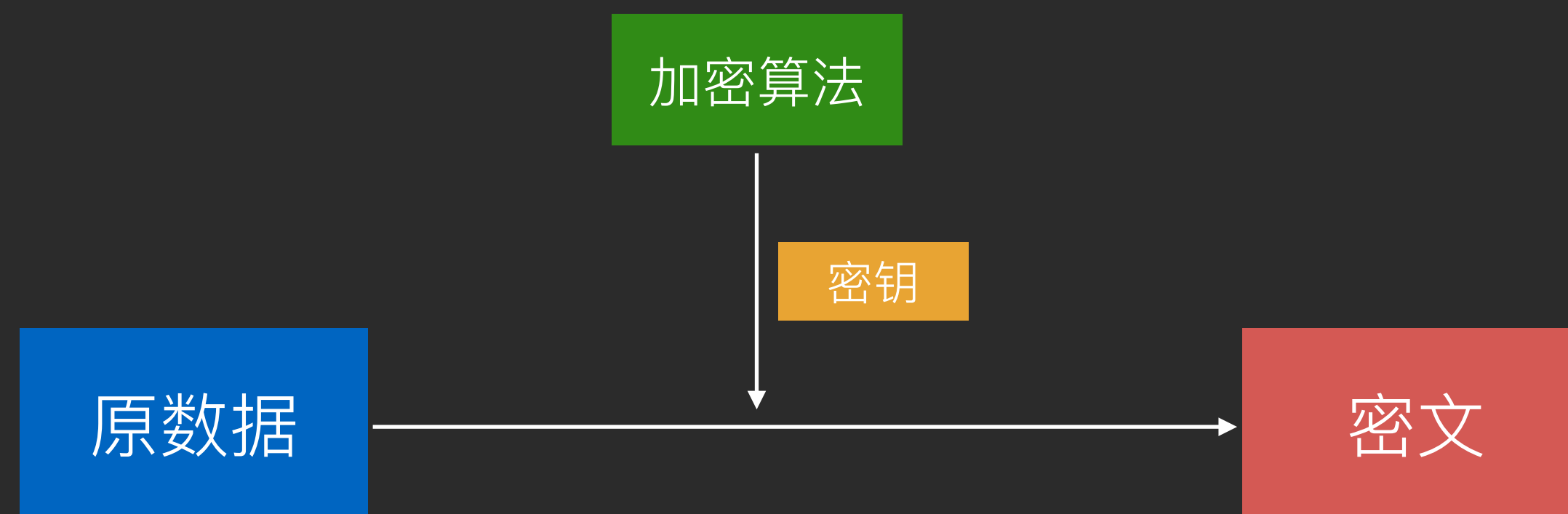
# 对称加密

原数据

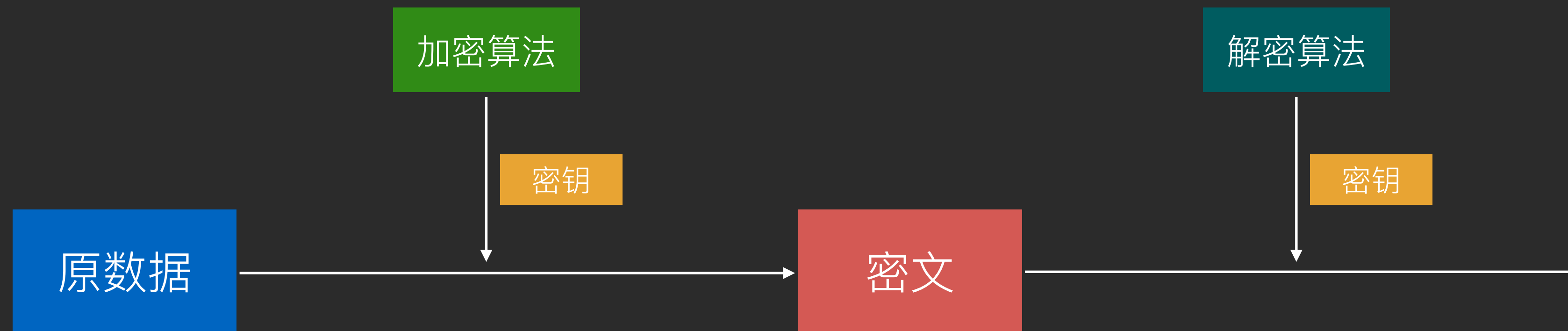
# 对称加密



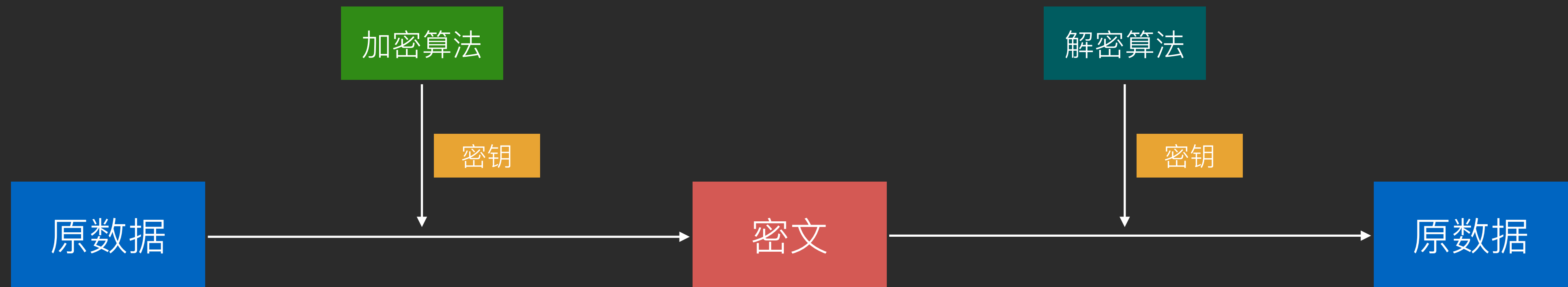
# 对称加密



# 对称加密



# 对称加密





# 现代密码学

- 不止可以用于文字内容，还可以用于各种二进制数据。
- 对称加密
  - 原理：使用密钥和加密算法对数据进行转换，得到的无意义数据即为密文；使用密钥和解密算法对密文进行逆向转换，得到原数据。

# 现代密码学

- 不止可以用于文字内容，还可以用于各种二进制数据。
- 对称加密
  - 原理：使用密钥和加密算法对数据进行转换，得到的无意义数据即为密文；使用密钥和解密算法对密文进行逆向转换，得到原数据。
  - 经典算法：DES，AES

# 现代密码学

- 不止可以用于文字内容，还可以用于各种二进制数据。
- 对称加密
  - 原理：使用密钥和加密算法对数据进行转换，得到的无意义数据即为密文；使用密钥和解密算法对密文进行逆向转换，得到原数据。
  - 经典算法：DES，AES
- 非对称加密

# 现代密码学

- 不止可以用于文字内容，还可以用于各种二进制数据。
- 对称加密
  - 原理：使用密钥和加密算法对数据进行转换，得到的无意义数据即为密文；使用密钥和解密算法对密文进行逆向转换，得到原数据。
  - 经典算法：DES，AES
- 非对称加密
  - 原理：使用公钥对数据进行加密得到密文；使用私钥对数据进行解密得到原数据。

# 非对称加密

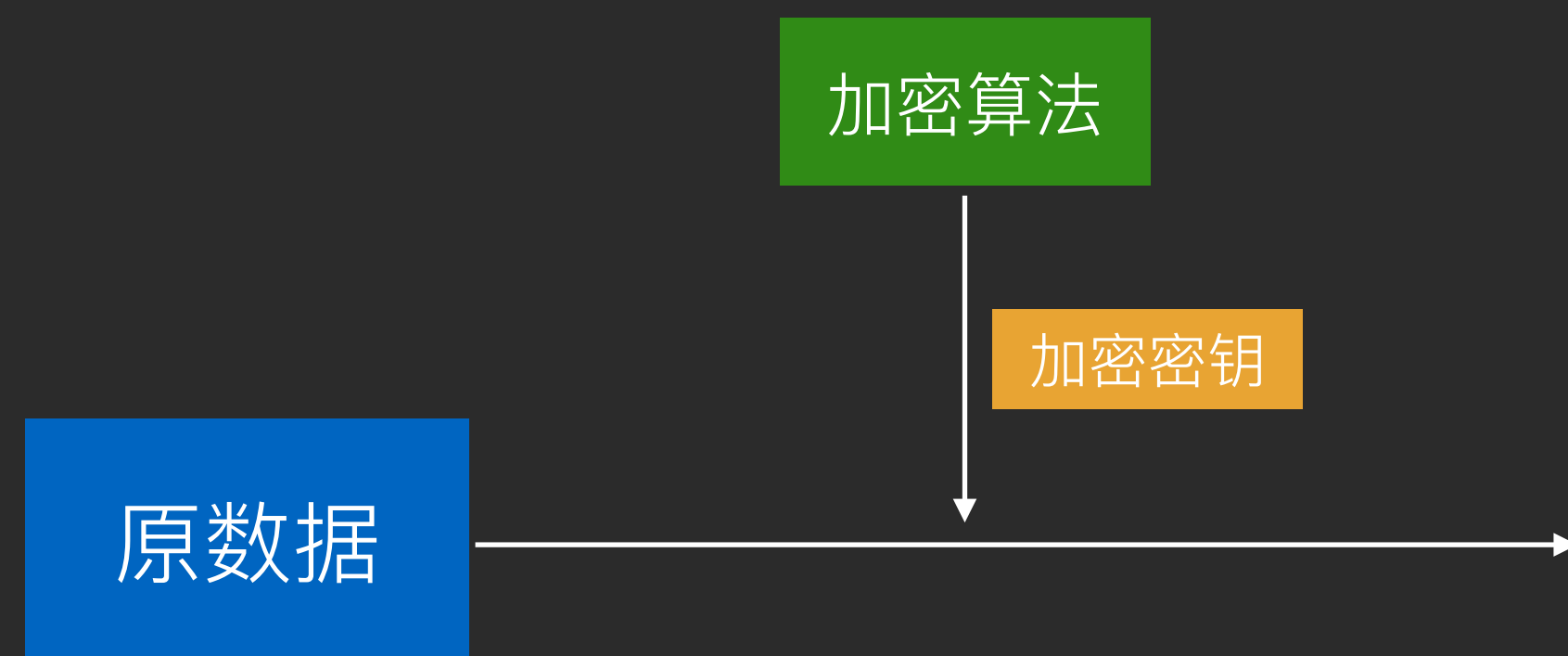
# 非对称加密

加密和解密

原数据

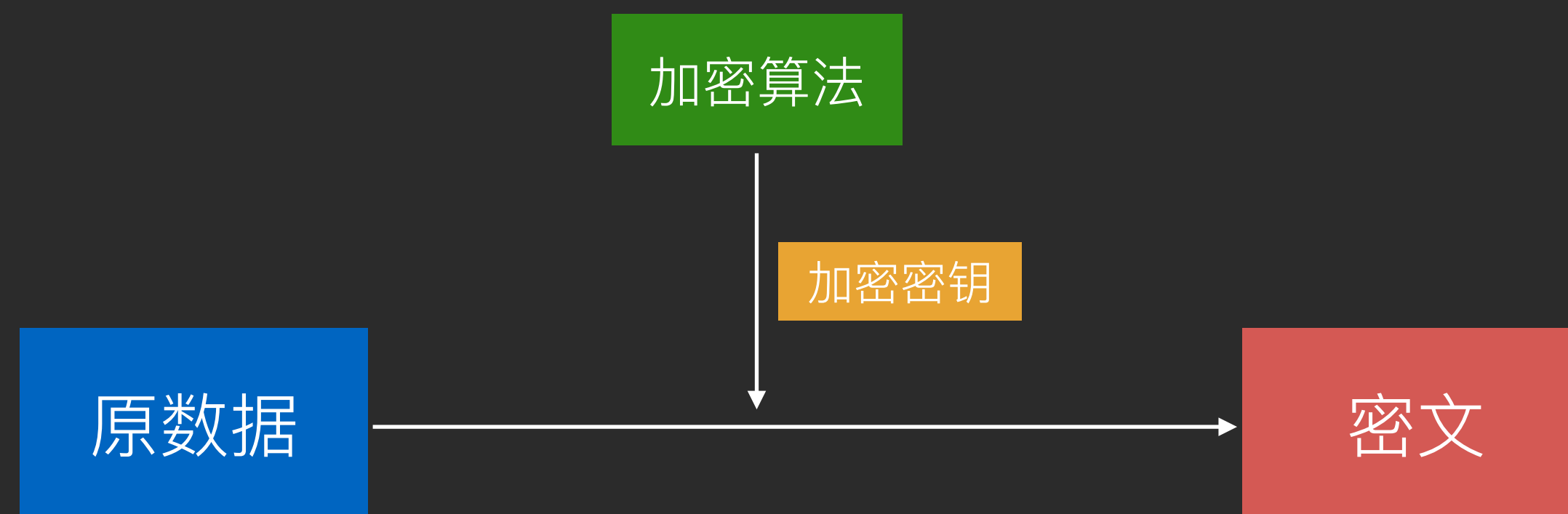
# 非对称加密

# 加密和解密



# 非对称加密

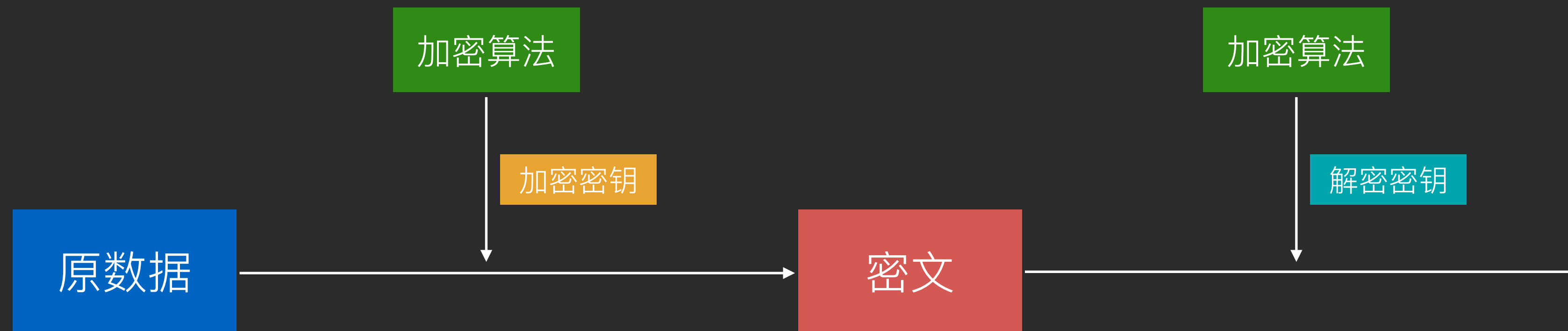
加密和解密





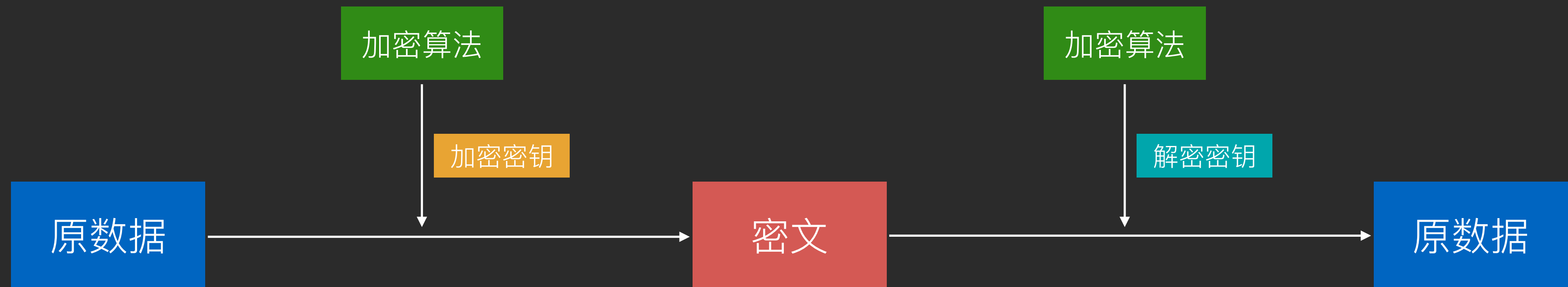
# 非对称加密

加密和解密



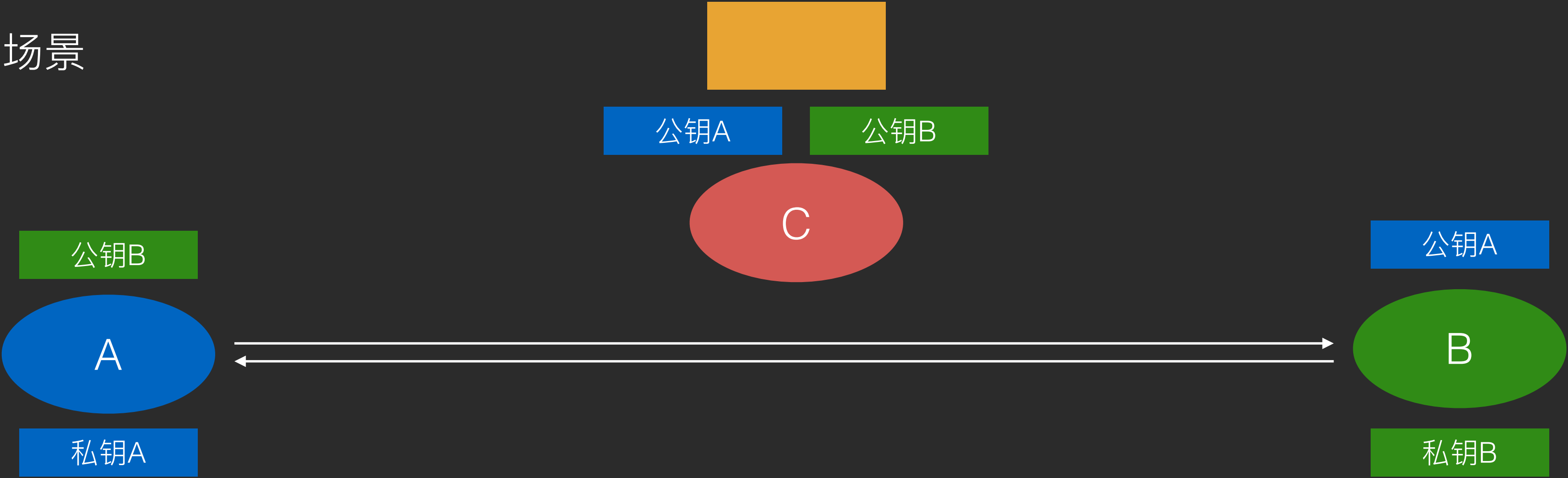
# 非对称加密

加密和解密



# 非对称加密

场景



# 现代密码学

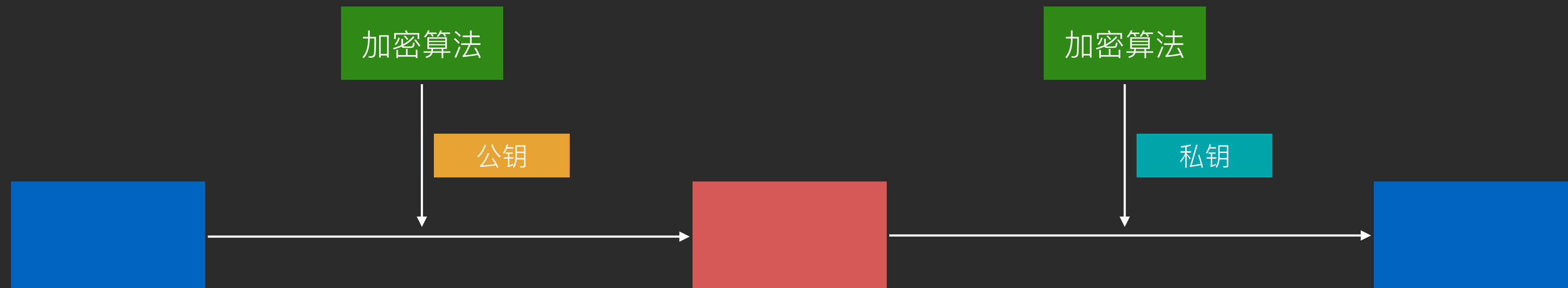
- 不止可以用于文字内容，还可以用于各种二进制数据。
- 对称加密
  - 原理：使用密钥和加密算法对数据进行转换，得到的无意义数据即为密文；使用密钥和解密算法对密文进行逆向转换，得到原数据。
  - 经典算法：DES，AES
- 非对称加密
  - 原理：使用公钥对数据进行加密得到密文；使用私钥对数据进行解密得到原数据。

# 现代密码学

- 不止可以用于文字内容，还可以用于各种二进制数据。
- 对称加密
  - 原理：使用密钥和加密算法对数据进行转换，得到的无意义数据即为密文；使用密钥和解密算法对密文进行逆向转换，得到原数据。
  - 经典算法：DES，AES
- 非对称加密
  - 原理：使用公钥对数据进行加密得到密文；使用私钥对数据进行解密得到原数据。
  - 延伸用途：数字签名。

# 非对称加密

公钥能不能解私钥？



# 非对称加密

签名与验证

# 非对称加密

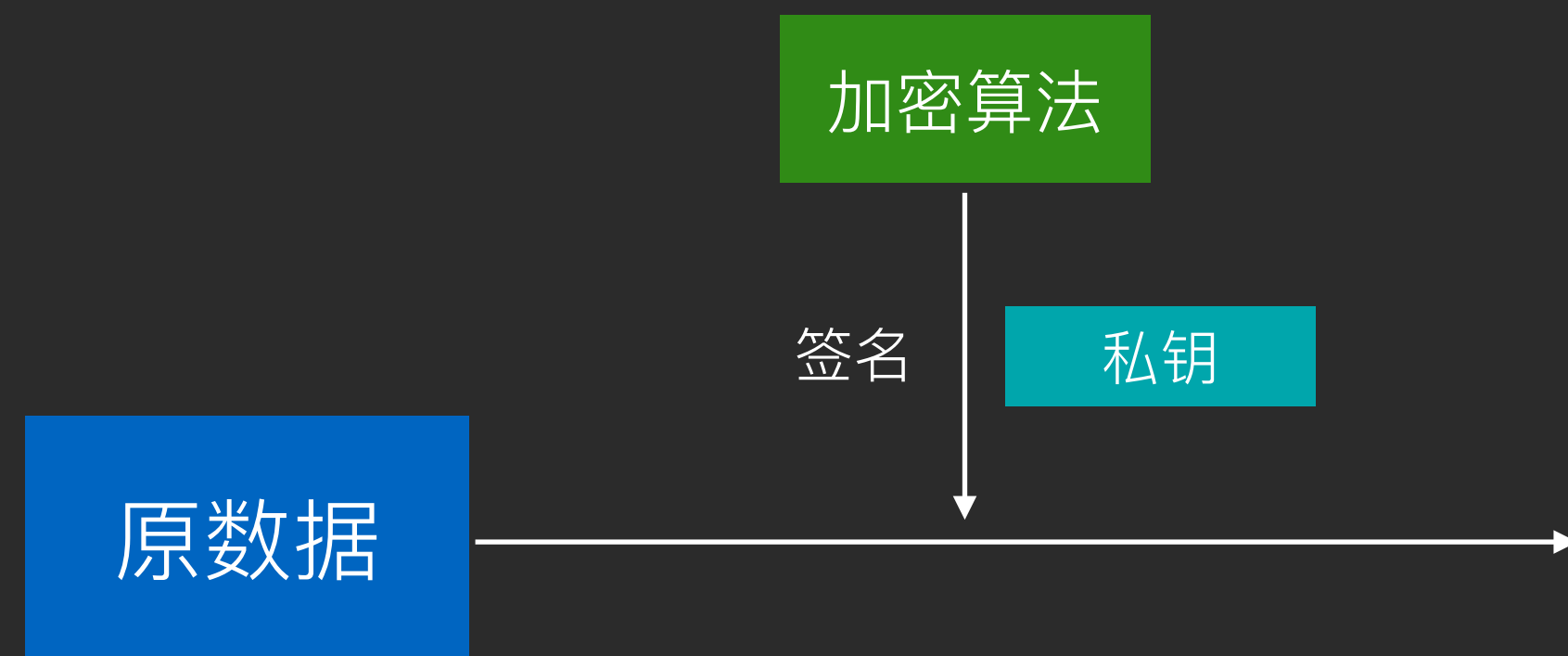
签名与验证

原数据



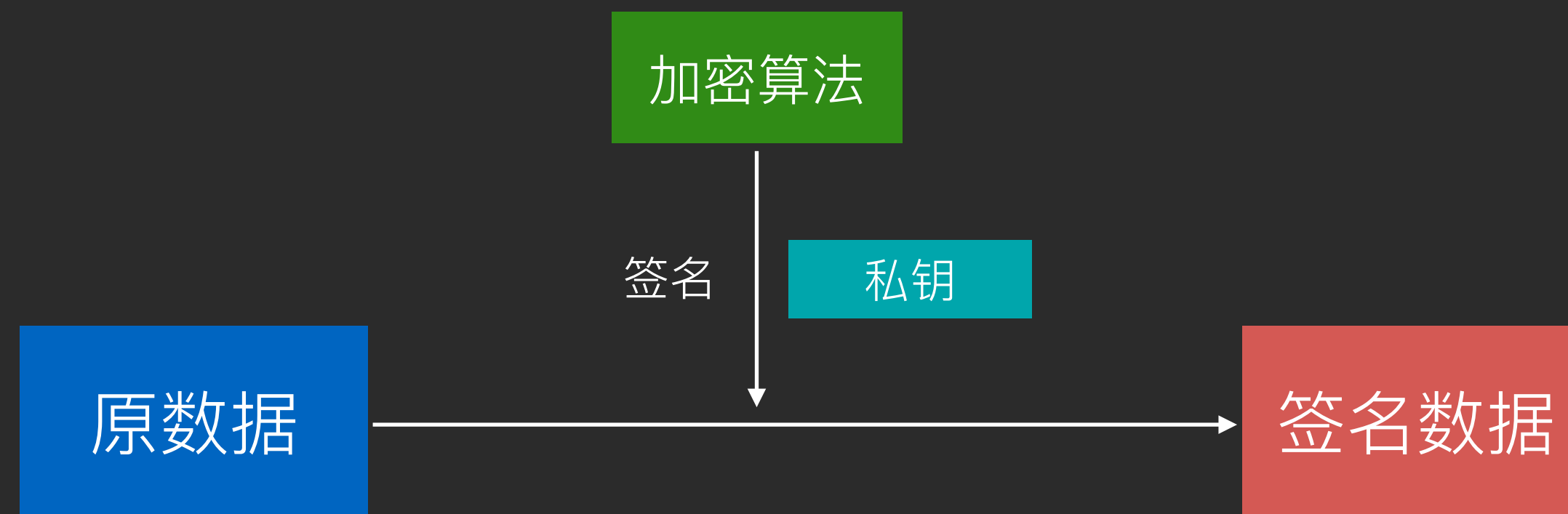
# 非对称加密

签名与验证



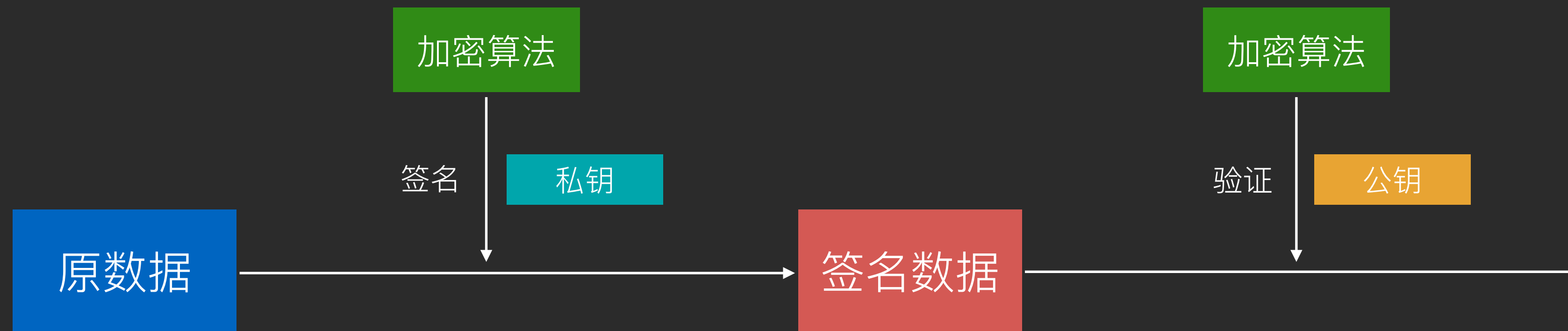
# 非对称加密

签名与验证



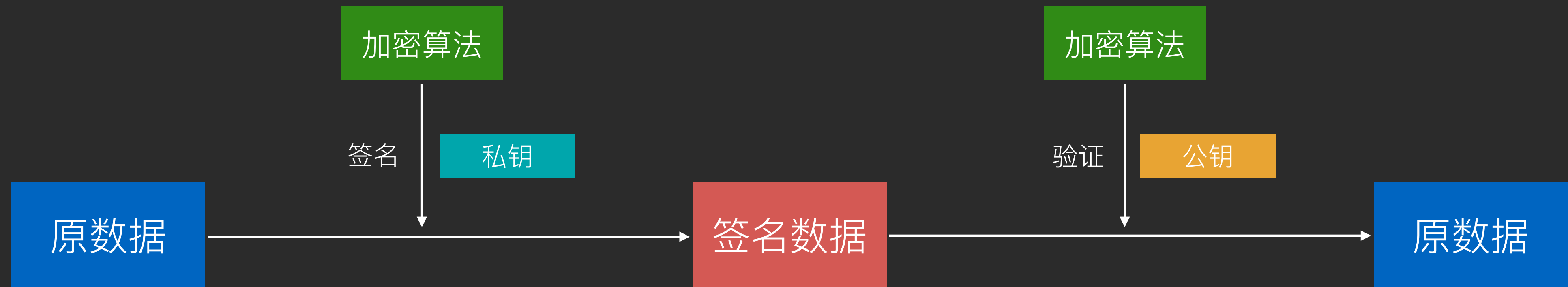
# 非对称加密

签名与验证



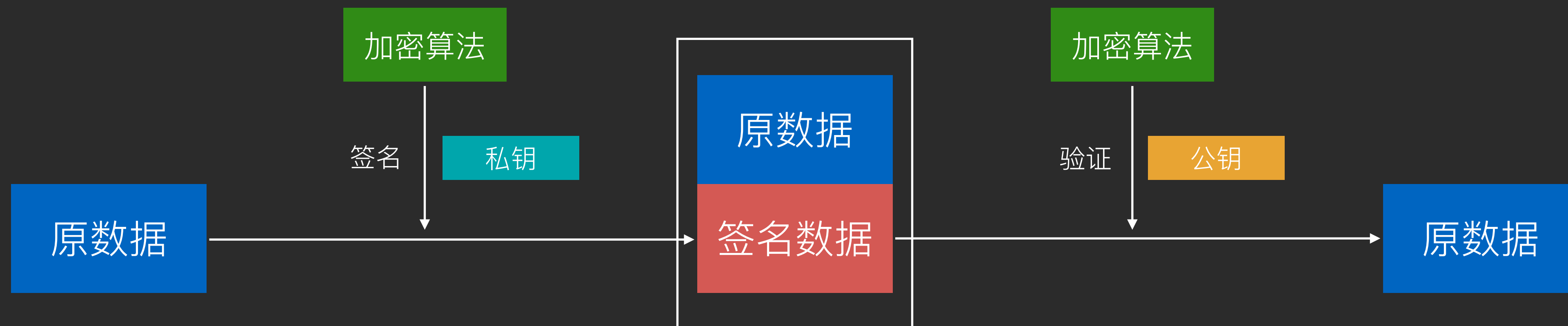
# 非对称加密

签名与验证



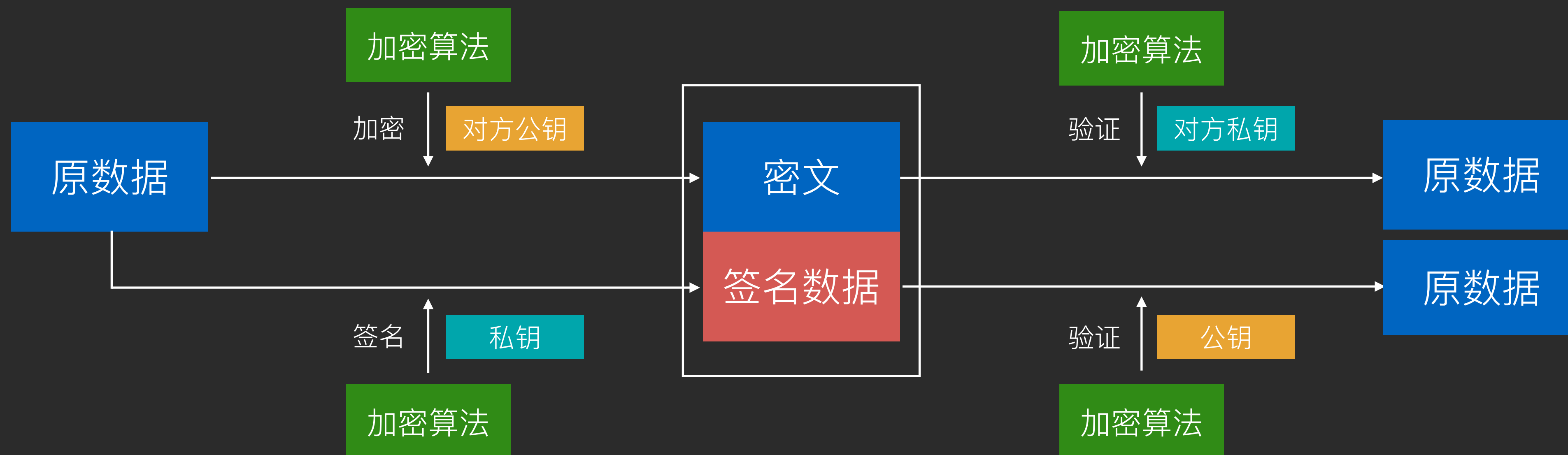
# 非对称加密

签名与验证



# 非对称加密

加密 + 签名



# 现代密码学

- 不止可以用于文字内容，还可以用于各种二进制数据。
- 对称加密
  - 原理：使用密钥和加密算法对数据进行转换，得到的无意义数据即为密文；使用密钥和解密算法对密文进行逆向转换，得到原数据。
  - 经典算法：DES，AES
- 非对称加密
  - 原理：使用公钥对数据进行加密得到密文；使用私钥对数据进行解密得到原数据。
  - 延伸用途：数字签名。

# 现代密码学

- 不止可以用于文字内容，还可以用于各种二进制数据。
- 对称加密
  - 原理：使用密钥和加密算法对数据进行转换，得到的无意义数据即为密文；使用密钥和解密算法对密文进行逆向转换，得到原数据。
  - 经典算法：DES，AES
- 非对称加密
  - 原理：使用公钥对数据进行加密得到密文；使用私钥对数据进行解密得到原数据。
  - 延伸用途：数字签名。
  - 经典算法：RSA，DSA



# 密码学密钥和登录密码

- 密钥
- 登录密码

# 密码学密钥和登录密码

- 密钥 (key)
- 登录密码

# 密码学密钥和登录密码

- 密钥 (key)
- 登录密码 (password)

# 密码学密钥和登录密码

- 密钥 (key)
  - 场景：用于加密和解密
- 登录密码 (password)

# 密码学密钥和登录密码

- 密钥 (key)
  - 场景：用于加密和解密
- 登录密码 (password)
  - 场景：用户进入网站或游戏前的身份验证

# 密码学密钥和登录密码

- 密钥 (key)
  - 场景：用于加密和解密
  - 目的：保证数据被盗时不会被人读懂内容
- 登录密码 (password)
  - 场景：用户进入网站或游戏前的身份验证

# 密码学密钥和登录密码

- 密钥 (key)
  - 场景：用于加密和解密
  - 目的：保证数据被盗时不会被人读懂内容
- 登录密码 (password)
  - 场景：用户进入网站或游戏前的身份验证
  - 目的：数据提供方或应用服务方对账户拥有者数据的保护，保证「你是你」的时候才提供权限

# 密码学密钥和登录密码

- 密钥 (key)
  - 场景：用于加密和解密
  - 目的：保证数据被盗时不会被人读懂内容
  - 焦点：数据
- 登录密码 (password)
  - 场景：用户进入网站或游戏前的身份验证
  - 目的：数据提供方或应用服务方对账户拥有者数据的保护，保证「你是你」的时候才提供权限



# 密码学密钥和登录密码

- 密钥 (key)
  - 场景：用于加密和解密
  - 目的：保证数据被盗时不会被人读懂内容
  - 焦点：数据
- 登录密码 (password)
  - 场景：用户进入网站或游戏前的身份验证
  - 目的：数据提供方或应用服务方对账户拥有者数据的保护，保证「你是你」的时候才提供权限
  - 焦点：身份

# Base64

# Base64

- 将二进制数据转换成由 64 个字符组成的字符串的编码算法

# Base64

- 将二进制数据转换成由 64 个字符组成的字符串的编码算法
- 什么是二进制数据？

# Base64

- 将二进制数据转换成由 64 个字符组成的字符串的编码算法
- 什么是二进制数据？
- 用途：

# Base64

- 将二进制数据转换成由 64 个字符组成的字符串的编码算法
- 什么是二进制数据？
- 用途：
  - 让原数据具有字符串所具有的特性，如可以放在 URL 中传输、可以保存到文本文件、可以通过普通的聊天软件进行文本传输。

# Base64

- 将二进制数据转换成由 64 个字符组成的字符串的编码算法
- 什么是二进制数据？
- 用途：
  - 让原数据具有字符串所具有的特性，如可以放在 URL 中传输、可以保存到文本文件、可以通过普通的聊天软件进行文本传输。
  - 把原本人眼可以读懂的字符串变成读不懂的字符串，降低偷窥风险

# Base64

- 将二进制数据转换成由 64 个字符组成的字符串的编码算法
- 什么是二进制数据？
- 用途：
  - 让原数据具有字符串所具有的特性，如可以放在 URL 中传输、可以保存到文本文件、可以通过普通的聊天软件进行文本传输。
  - 把原本人眼可以读懂的字符串变成读不懂的字符串，降低偷窥风险
- 「Base64 加密传输图片，可以更安全和高效」，真的吗？



# Base64

- 将二进制数据转换成由 64 个字符组成的字符串的编码算法
- 什么是二进制数据？
- 用途：
  - 让原数据具有字符串所具有的特性，如可以放在 URL 中传输、可以保存到文本文件、可以通过普通的聊天软件进行文本传输。
  - 把原本人眼可以读懂的字符串变成读不懂的字符串，降低偷窥风险
- 「Base64 加密传输图片，可以更安全和高效」，真的吗？
- 变种：Base58

# URL encoding

# URL encoding

- 将 URL 中的保留字符使用百分号 "%" 进行编码

# URL encoding

- 将 URL 中的保留字符使用百分号 "%" 进行编码
- 目的：消除歧义，避免解析错误

# URL encoding

- 将 URL 中的保留字符使用百分号 "%" 进行编码
- 目的：消除歧义，避免解析错误
- `http://hencoder.com/user/?name=隐匿&伟大`

# URL encoding

- 将 URL 中的保留字符使用百分号 "%" 进行编码
- 目的：消除歧义，避免解析错误
- `http://hencoder.com/user/?name=隐匿&伟大` ->
- `http://hencoder.com/user/?name=隐匿%26伟大`

# 压缩与解压缩

# 压缩与解压缩

- 压缩：把数据换一种方式来存储，以减小存储空间



# 压缩与解压缩

- 压缩：把数据换一种方式来存储，以减小存储空间
- 解压缩：把压缩后的数据还原成原先的形式，以便使用

# 压缩与解压缩

- 压缩：把数据换一种方式来存储，以减小存储空间
- 解压缩：把压缩后的数据还原成原先的形式，以便使用
- 常见压缩算法：DEFLATE、JPEG、MP3

# 压缩与解压缩

- 压缩：把数据换一种方式来存储，以减小存储空间
- 解压缩：把压缩后的数据还原成原先的形式，以便使用
- 常见压缩算法：DEFLATE、JPEG、MP3
- 压缩属于编码吗？

# 压缩与解压缩

- 压缩：把数据换一种方式来存储，以减小存储空间
- 解压缩：把压缩后的数据还原成原先的形式，以便使用
- 常见压缩算法：DEFLATE、JPEG、MP3
- 压缩属于编码吗？
  - 编码到底是什么意思？

# 压缩与解压缩

- 压缩：把数据换一种方式来存储，以减小存储空间
- 解压缩：把压缩后的数据还原成原先的形式，以便使用
- 常见压缩算法：DEFLATE、JPEG、MP3
- 压缩属于编码吗？
  - 编码到底是什么意思？
  - 那么，压缩属于编码吗？

# 媒体数据的编解码

# 媒体数据的编解码

- 什么是图片、音频、视频的编解码？

# 媒体数据的编解码

- 什么是图片、音频、视频的编解码？
- 图片的编码：把图像数据写成 JPG、PNG 等文件的编码格式。



# 媒体数据的编解码

- 什么是图片、音频、视频的编解码？
- 图片的编码：把图像数据写成 JPG、PNG 等文件的编码格式。
- 图片的解码：把 JPG、PNG 等文件中的数据解析成标准的图像数据。

# 媒体数据的编解码

- 什么是图片、音频、视频的编解码？
- 图片的编码：把图像数据写成 JPG、PNG 等文件的编码格式。
- 图片的解码：把 JPG、PNG 等文件中的数据解析成标准的图像数据。
- 音频、视频的编解码

# 序列化

# 序列化

- 序列化：把数据对象（一般是内存中的，例如 JVM 中的对象）转换成字节序列的过程

# 序列化

- 序列化：把数据对象（一般是内存中的，例如 JVM 中的对象）转换成字节序列的过程
- 反序列化：把字节序列重新转换成内存中的对象

# 序列化

- 序列化：把数据对象（一般是内存中的，例如 JVM 中的对象）转换成字节序列的过程
- 反序列化：把字节序列重新转换成内存中的对象
- 目的：让内存中的对象可以被存储和传输

# 序列化

- 序列化：把数据对象（一般是内存中的，例如 JVM 中的对象）转换成字节序列的过程
- 反序列化：把字节序列重新转换成内存中的对象
- 目的：让内存中的对象可以被存储和传输
- 序列化是编码吗？

# Hash



# Hash

- 定义：把任意数据转换成指定大小范围（通常很小）的数据

# Hash

- 定义：把任意数据转换成指定大小范围（通常很小）的数据
- 作用：摘要、数字指纹

# Hash

- 定义：把任意数据转换成指定大小范围（通常很小）的数据
- 作用：摘要、数字指纹
- 经典算法：MD5、SHA1、SHA256 等

# Hash

- 定义：把任意数据转换成指定大小范围（通常很小）的数据
- 作用：摘要、数字指纹
- 经典算法：MD5、SHA1、SHA256 等
- 实际用途

# Hash

- 定义：把任意数据转换成指定大小范围（通常很小）的数据
- 作用：摘要、数字指纹
- 经典算法：MD5、SHA1、SHA256 等
- 实际用途
  - 数据完整性验证

# Hash

- 定义：把任意数据转换成指定大小范围（通常很小）的数据
- 作用：摘要、数字指纹
- 经典算法：MD5、SHA1、SHA256 等
- 实际用途
  - 数据完整性验证
  - 快速查找：hashCode() 和 HashMap

# Hash

- 定义：把任意数据转换成指定大小范围（通常很小）的数据
- 作用：摘要、数字指纹
- 经典算法：MD5、SHA1、SHA256 等
- 实际用途
  - 数据完整性验证
  - 快速查找：hashCode() 和 HashMap
  - 隐私保护

# Hash

- 定义：把任意数据转换成指定大小范围（通常很小）的数据
- 作用：摘要、数字指纹
- 经典算法：MD5、SHA1、SHA256 等
- 实际用途
  - 数据完整性验证
  - 快速查找：hashCode() 和 HashMap
  - 隐私保护
- Hash 是编码吗？

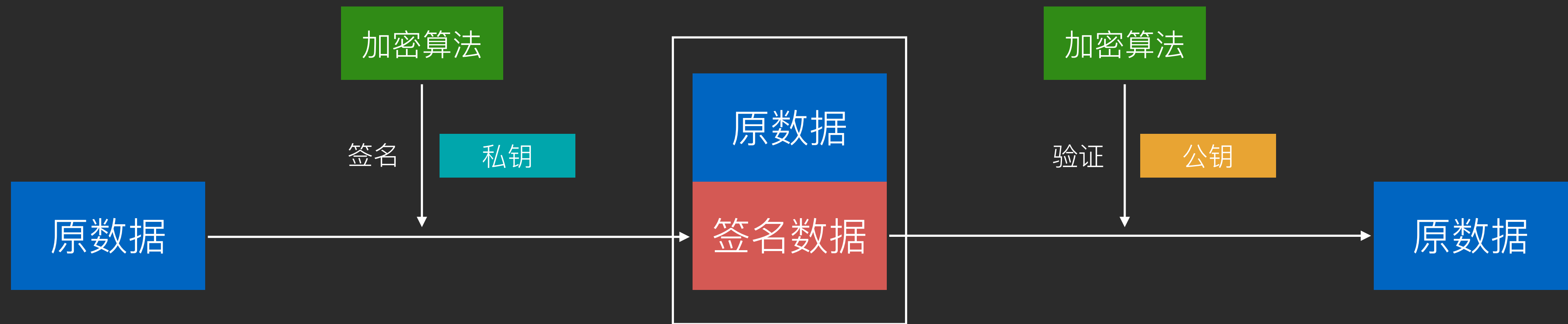


# Hash

- 定义：把任意数据转换成指定大小范围（通常很小）的数据
- 作用：摘要、数字指纹
- 经典算法：MD5、SHA1、SHA256 等
- 实际用途
  - 数据完整性验证
  - 快速查找：hashCode() 和 HashMap
  - 隐私保护
- Hash 是编码吗？
- Hash 是加密吗？据说 MD5是「不可逆加密」？

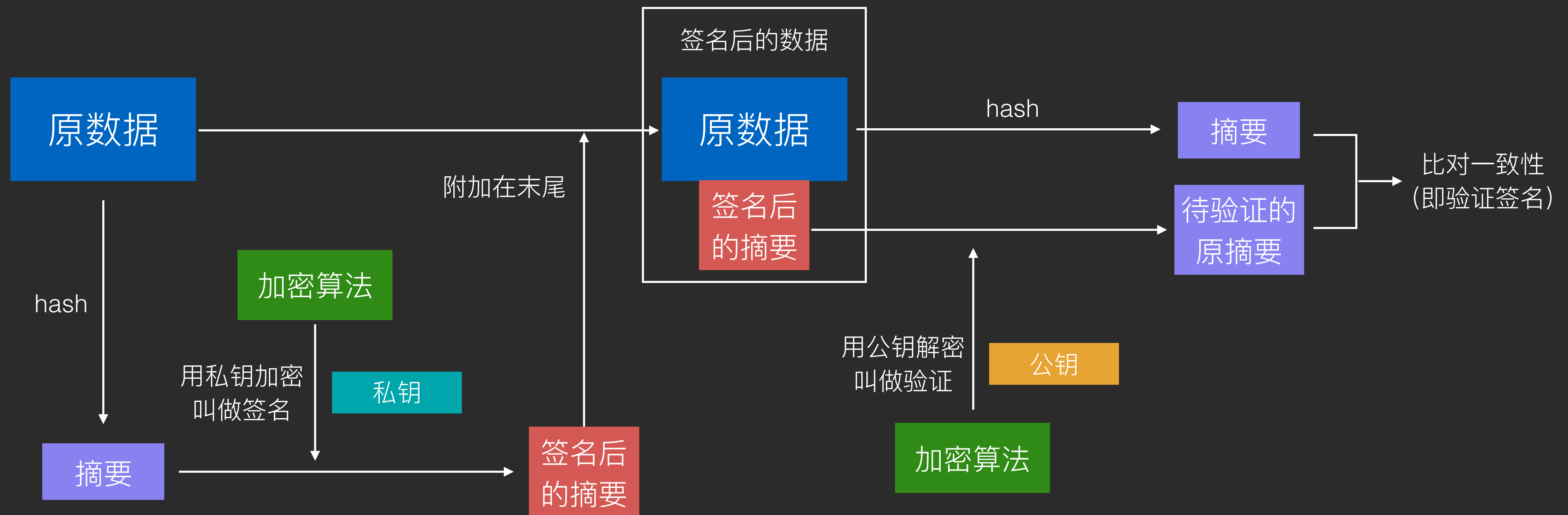
# 非对称加密

签名与验证



# 非对称加密

## 签名与验证（完整）



# 字符集

# 字符集

- 含义：一个由整数向现实世界中的文字符号的 Map

# 字符集

- 含义：一个由整数向现实世界中的文字符号的 Map
- 分支：

# 字符集

- 含义：一个由整数向现实世界中的文字符号的 Map
- 分支：
  - ASCII：128 个字符，1 字节

# 字符集

- 含义：一个由整数向现实世界中的文字符号的 Map
- 分支：
  - ASCII：128 个字符，1 字节
  - ISO-8859-1：对 ASCII 进行扩充，1 字节



# 字符集

- 含义：一个由整数向现实世界中的文字符号的 Map
- 分支：
  - ASCII：128 个字符，1 字节
  - ISO-8859-1：对 ASCII 进行扩充，1 字节
  - Unicode：13 万个字符，多字节


# 字符集

- 含义：一个由整数向现实世界中的文字符号的 Map
- 分支：
  - ASCII：128 个字符，1 字节
  - ISO-8859-1：对 ASCII 进行扩充，1 字节
  - Unicode：13 万个字符，多字节
    - UTF-8：Unicode 的编码分支
    - UTF-16：Unicode 的编码分支

# 字符集

- 含义：一个由整数向现实世界中的文字符号的 Map
- 分支：
  - ASCII：128 个字符，1 字节
  - ISO-8859-1：对 ASCII 进行扩充，1 字节
  - Unicode：13 万个字符，多字节
    - UTF-8：Unicode 的编码分支
    - UTF-16：Unicode 的编码分支
  - GBK / GB2312 / GB18030：中国自研标准，多字节，字符集 + 编码

# 下期内容

- 登录和授权、HTTPS、TCP/IP 协议族
- 问题和建议：丢物线 
- 网站：hencoder.com
- 微信公众号：HenCoder

