

PKI公钥基础架构

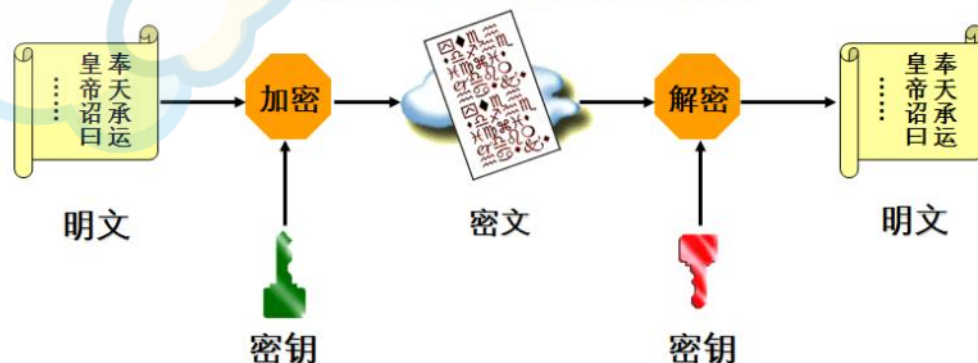
什么是PKI

- ❖ **Public Key Infrastructure**，公钥基础结构
- ❖ 通过**公钥技术**与**数字证书**确保信息安全的体系
 - 由公钥加密技术、数字证书、CA、RA等共同组成
- ❖ PKI体系能够实现的功能有：
 - 机密性
 - 完整性
 - 身份验证
 - 不可否认性

机密性技术：

加密技术简介

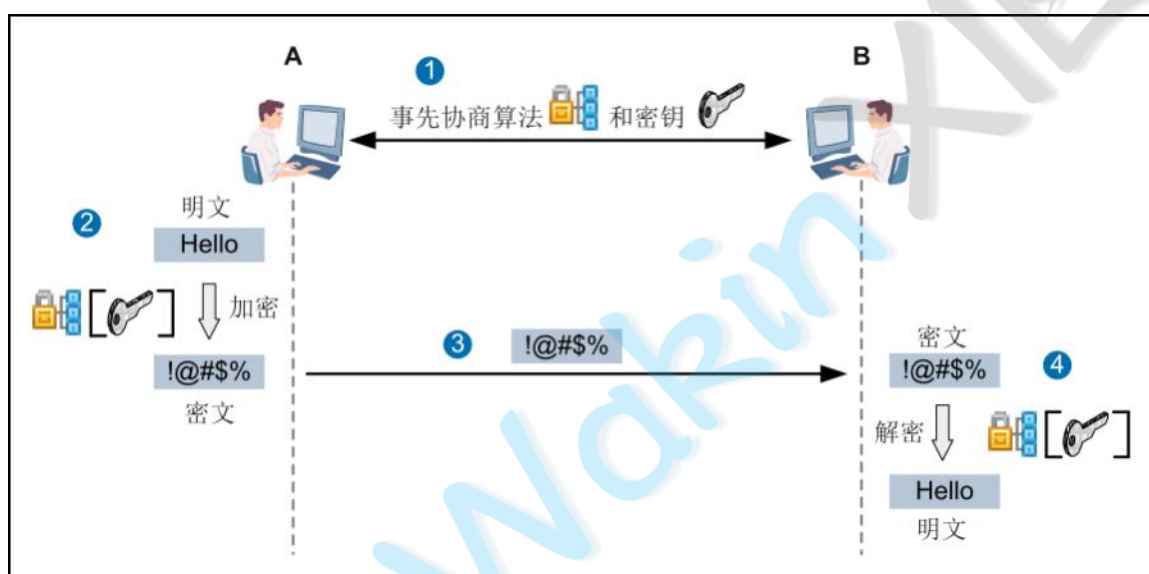
- ❖ **明文**：需要被隐蔽的消息
- ❖ **密文**：明文经变换形成的隐蔽形式
- ❖ **加密**：把明文转化为密文的过程
- ❖ **解密**：把密文还原成明文的过程
- ❖ **密钥**：在加密或解密的算法中输入的参数
- ❖ 加密算法分为两类：对称加密算法和非对称加密算法



加密技术分类：根据密钥的使用方法

对称加密：加密、解密使用**相同的密钥**

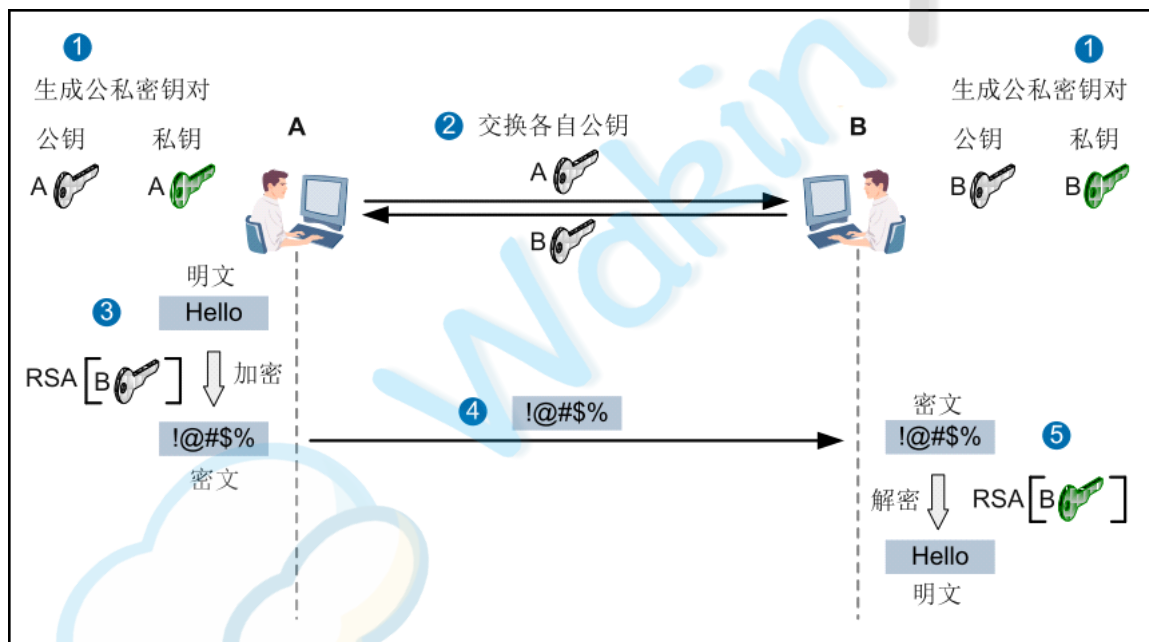
- 特点：速度快、密文紧凑、密钥管理复杂、用于大量数据的传送



| 算法 | 密钥 | 开发者 | 备注 |
|-------------------|---------------|-------------------------------|-----------|
| 数据加密标准 DES | 56位 | IBM为美国政府 (NBS/NIST)开发 | 很多政府强制性使用 |
| 3DES | 3*56位 | IBM为美国政府 (NBS/NIST)开发 | 应用3次DES |
| CAST | 40-256位可 变 | 北方通信 | 比DES稍快 |
| Rivest算法 (RC2) | 可变 | Ron Rivest (RSA数据安全) | 专利细节未公开 |
| RC5 | | Ron Rivest (RSA数据安全) | |
| AES | | Joan Daemen/Vincent Rijmen | |

非对称加密：加密、解密使用不同的密钥（公钥、私钥）

- 公加私解、私加公解
- 特点：速度慢、密文不紧凑、密钥管理简单、通常只用于数字签名



公钥加密法，又称非对称加密法。其工作原理如下。

1. 客户端 A 要向服务器 B 发送信息，B 要产生一对用于加密和解密的公钥和私钥。
2. B 的私钥自己保密，B 的公钥告诉 A。
3. A 要给 B 发送信息时，用 B 的公钥加密信息，因为 A 知道 B 的公钥。
4. B 收到这个信息后，自己的私钥解密。其他所有收到这个报文的人都无法解密，因为只有 B 才有自己的私钥。
5. B 要给 A 发送消息时，亦同理加密。

| 算法 | 设计者 | 用途 | 安全性 |
|-----|----------------|--------------------|--------|
| RSA | RSA数据安全 | 加密 数字签名 密钥交换 | 大数分解 |
| DSA | NSA | 数字签名 | 离散对数 |
| DH | Diffie&Hellman | 密钥交换 | 完全向前保密 |

Diffie-Hellman Key Exchange (Cont.)

Peer A 

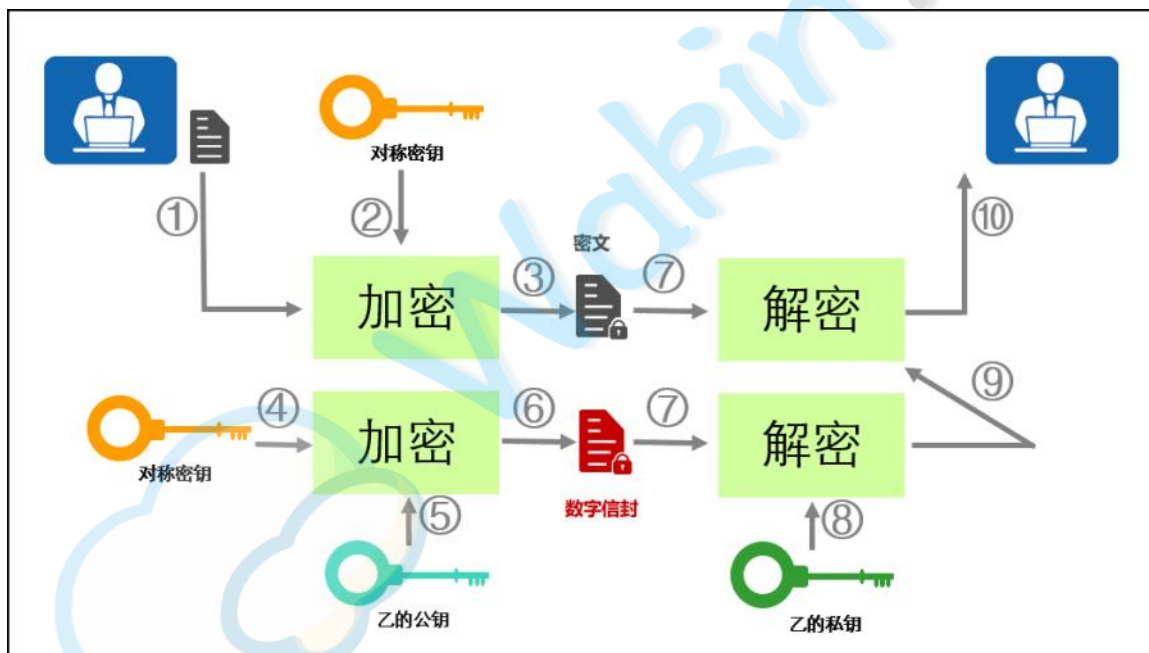
 Peer B

- | | | |
|---|-----------------------|---|
| <ol style="list-style-type: none"> 1. Generate large integer p Send p to peer B Receive q Generate g 2. Generate private key X_A 3. Generate public key $Y_A = g^{X_A} \bmod p$ 4. Send public key Y_A 5. Generate shared secret number $ZZ = Y_B^{X_A} \bmod p$ 6. Generate shared secret key from ZZ (DES, 3DES, or AES) | \longleftrightarrow | <ol style="list-style-type: none"> 1. Generate large integer q Send q to peer A Receive p Generate g 2. Generate private key X_B 3. Generate public key $Y_B = g^{X_B} \bmod p$ 4. Send public key Y_B 5. Generate shared secret number $ZZ = Y_A^{X_B} \bmod p$ 6. Generate shared secret key from ZZ (DES, 3DES, or AES) |
|---|-----------------------|---|

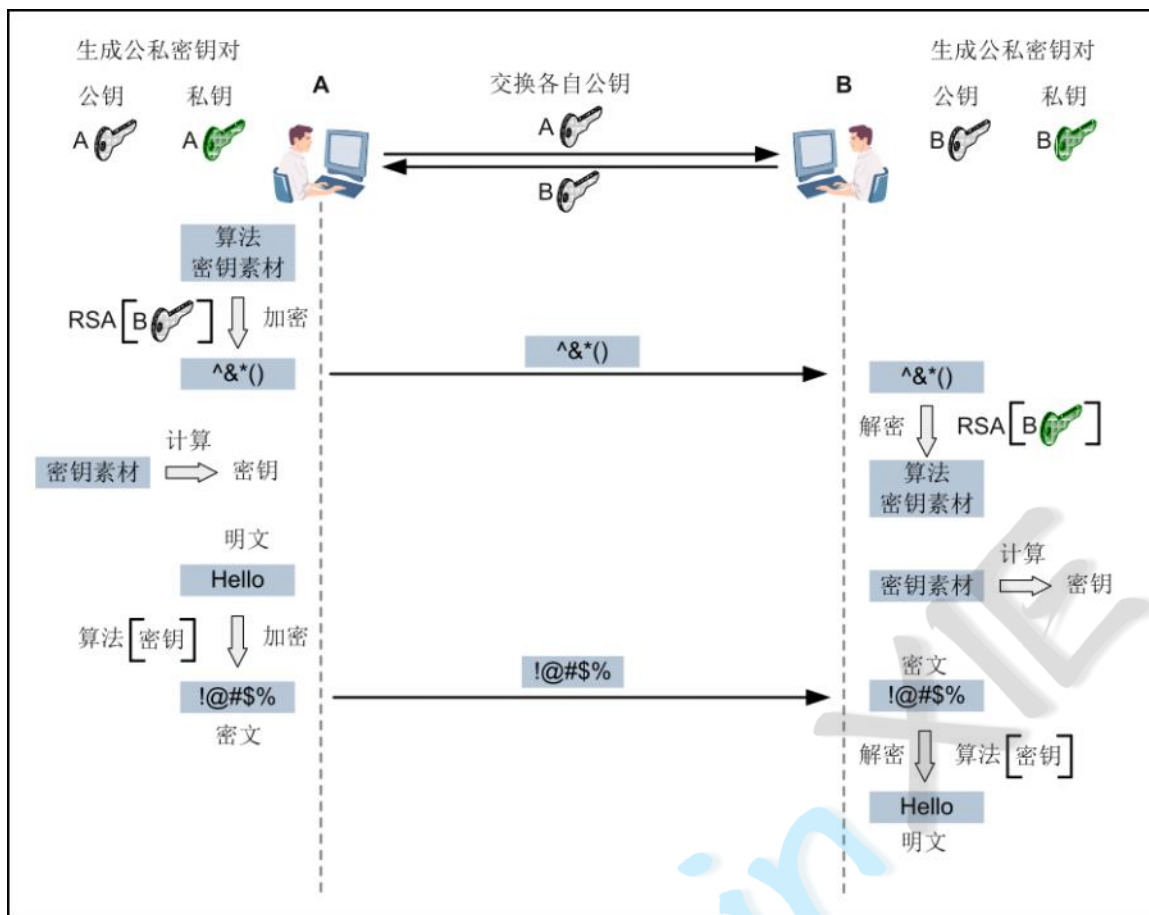
公钥加密技术

- ❖ 非对称加密也叫公钥加密，是PKI的基础
- ❖ 公钥 (Public Key) 和私钥 (Private Key)
 - 公钥和私钥是成对生成，互不相同，可以互相加密和解密
 - 根据一个密钥无法推算出另外一个密钥
 - 公钥公开，私钥保密 (只有持有人才知道)
 - 私钥应该由密钥的持有人妥善保管
- ❖ 根据实现的功能不同，可以分为数据加密和数字签名

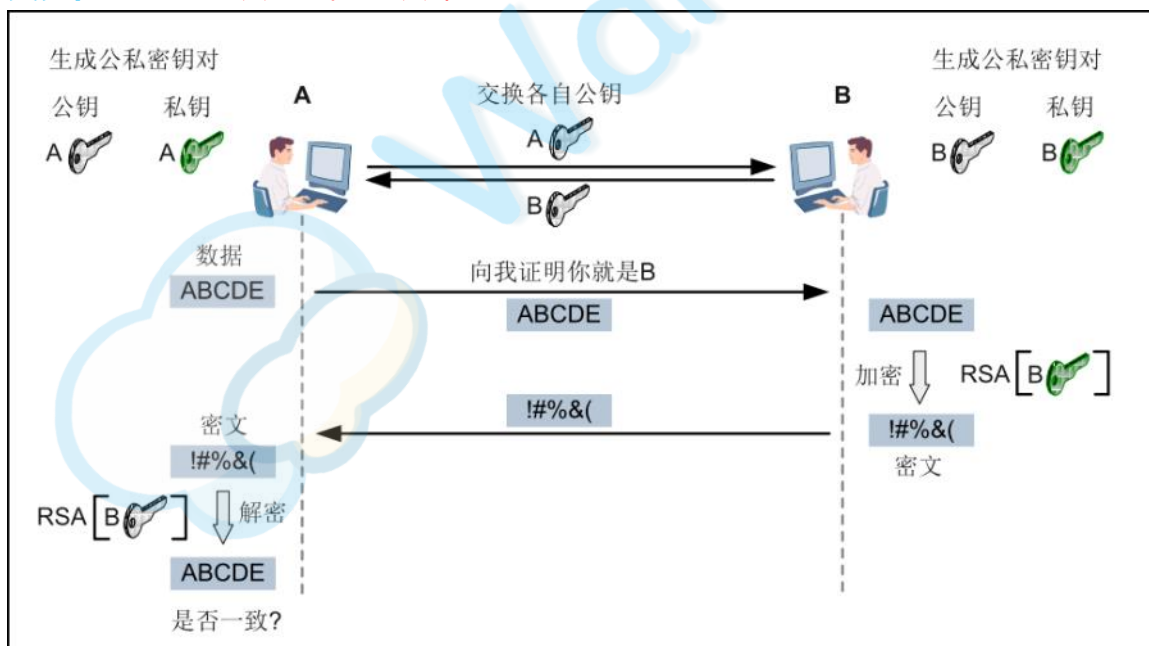
数据加密 - 数字信封：发送方采用接收方的公钥来加密对称密钥。采用数字信封时，接收方需要使用自己的私钥才能打开数字信封得到对称密钥。



两种加密算法的结合：在实际使用中，通信双方通常会使用公钥密码学来交换密钥素材，双方最终计算出密钥，而用对称密码学来加密实际的数据，两者配合使用，保证了加密速度和安全性。



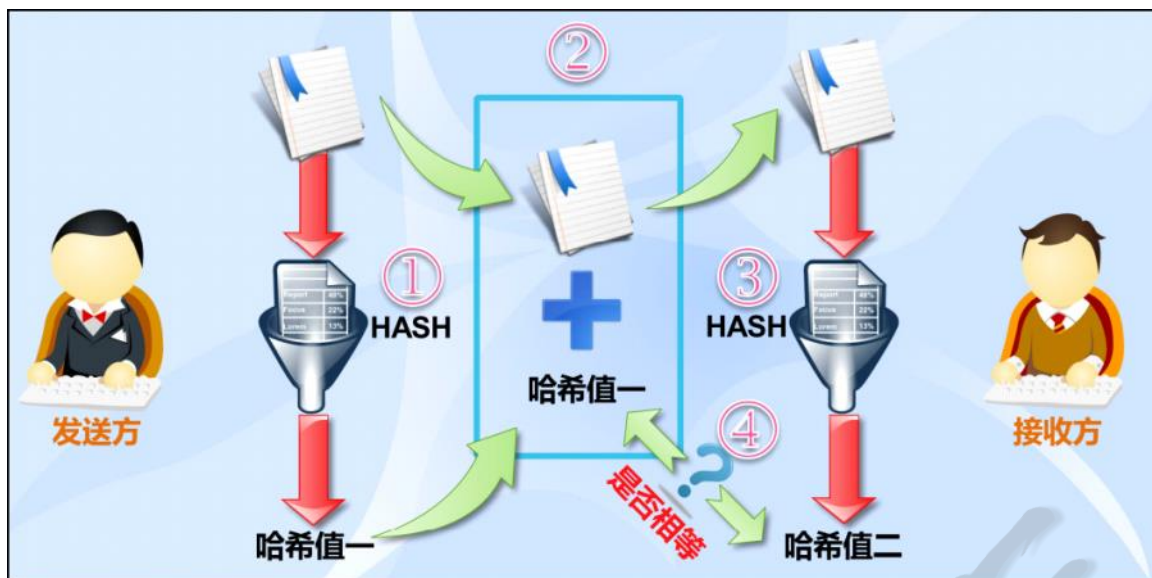
真实性验证：私钥加密，公钥解密



完整性技术：

信息摘要：单向散列函数，哈希 (Hash)

- 将任意长的字符串通过哈希计算出固定长度字符串，类似指纹、DNA；
- 特点：不可逆、雪崩效应。



| 项目 | MD5 | SHA1 | SHA2 |
|------|------------------|-------------------------|---|
| 全称 | Message Digest 5 | Secure Hash Algorithm 1 | Secure Hash Algorithm 2 |
| 签名长度 | 128位 | 160位 | SHA2-256 : 256位 SHA2-384 : 384位 SHA2-512 : 512位 |
| 安全级别 | 低 | 中 | 高 |

数字签名

❖ Digital Signature

- 发送方使用私钥对信息摘要进行加密的一个过程
- 过程中所得到的密文即称为签名信息
- ❖ 发送方将签名信息与原始数据发送给接收方
- ❖ 接收方对原始数据进行摘要计算，得出的值和签名信息进行比对
- ❖ 保证数据的完整性、身份验证和不可否认

数字签名流程

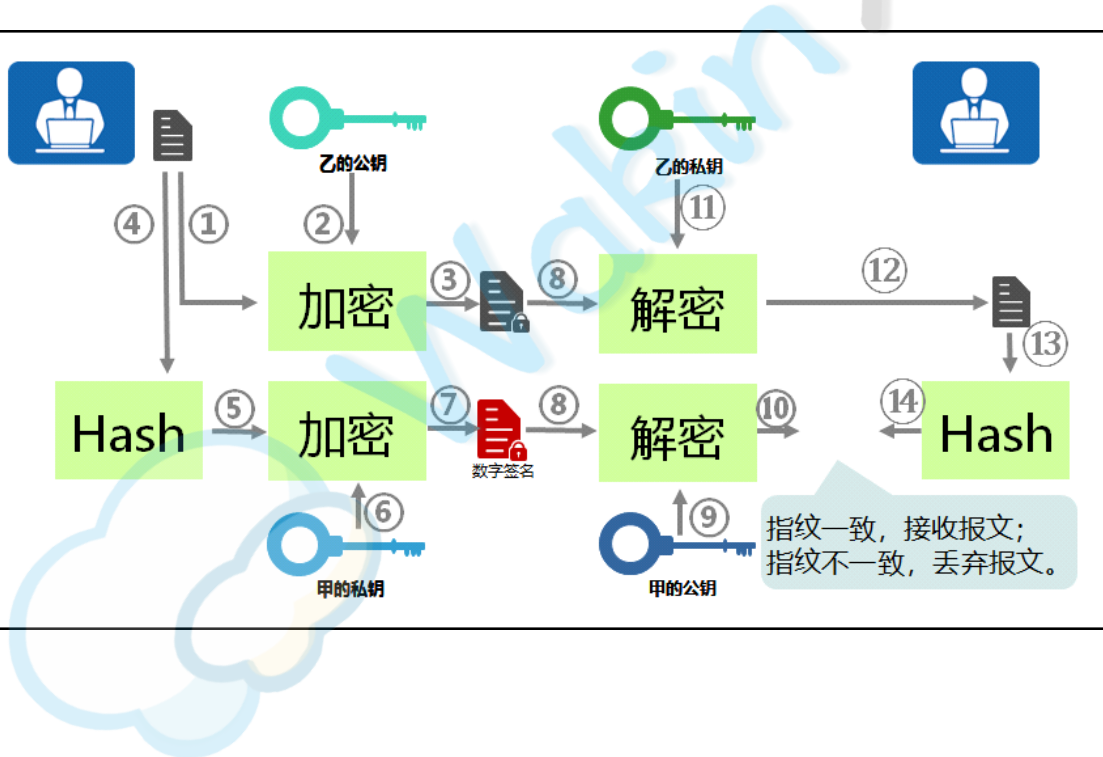
The diagram illustrates the digital signature process in two parts, A and B, separated by a cloud representing network transmission.

Part A: Signing

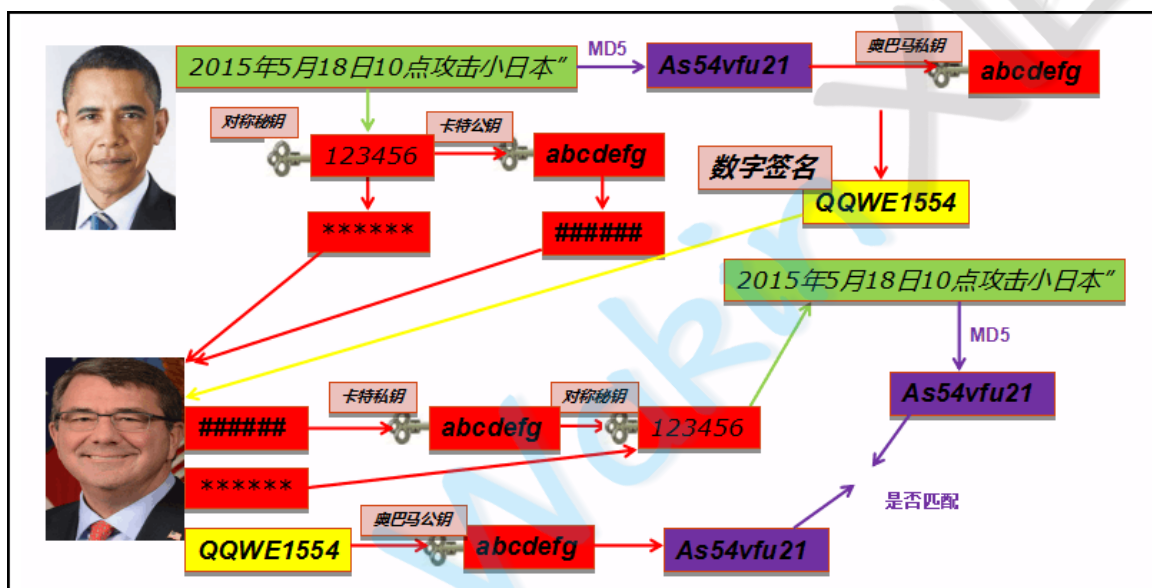
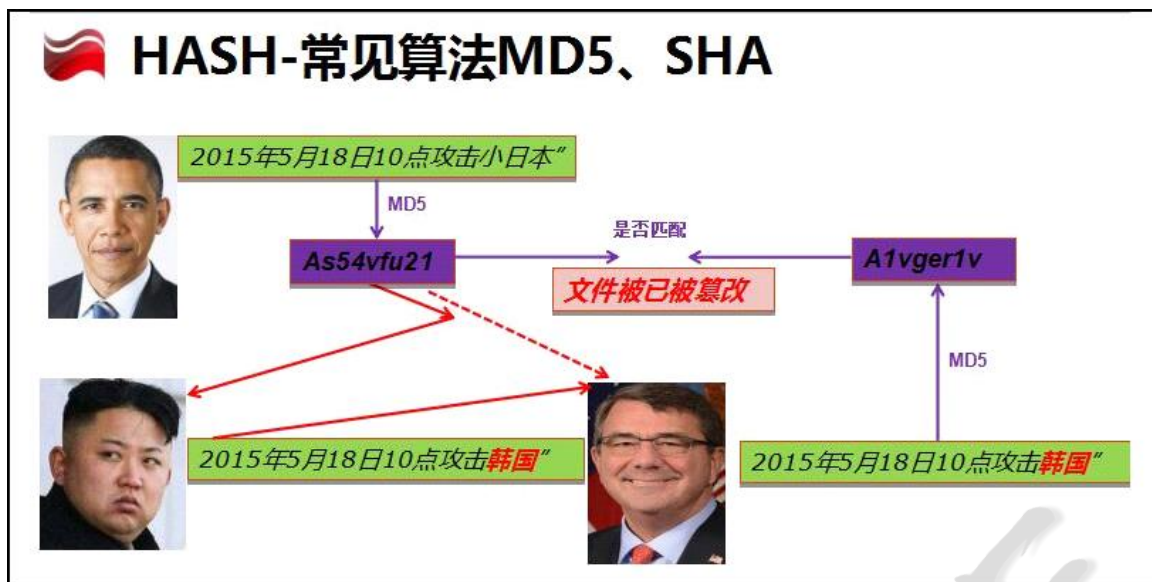
- 明文 (Plaintext):** A document with the text "皇奉", "帝天", "诏承", and "目运".
- 摘要 (Hashing):** The plaintext is processed by a "摘要" (hash) function, resulting in a hash value "gJ39vz&mp4x".
- 加密 (Encryption):** The hash value is combined with "A的私钥" (A's private key) and processed by an "加密" (encryption) function to create the "数字签名" (digital signature) "Ourij9rRr%9\$".
- 打包 (Packaging):** The original plaintext and the digital signature are combined into a single package.

Part B: Verification

- 接收 (Reception):** The package is received, containing the "数字签名" and the "明文".
- 解密 (Decryption):** The digital signature is decrypted using "A的公钥" (A's public key) to reveal the original hash value "gJ39vz&mp4x".
- 摘要 (Hashing):** The received plaintext is processed by a "摘要" (hash) function to create a "新摘要" (new hash) "gJ39vz&mp4x".
- 验证 (Verification):** The two hash values are compared. A starburst indicates they are "相同" (the same), confirming the document has not been altered.



HASH-常见算法MD5、SHA



公钥技术的规模应用难题

- ❖ 如何解决公钥的传播问题
 - 如何把自己的公钥告诉别人
 - 得到一个公钥后如何验证其真实性
- ❖ 公钥如何管理
- ❖ 如何实现不可否认服务

载体：证书

证书

- ❖ 证书用于保证密钥的合法性
- ❖ 证书的主体可以是用户、计算机、服务等
- ❖ 证书格式遵循X.509标准
- ❖ 数字证书包含信息
 - 使用者的公钥值
 - 使用者标识信息（如名称和电子邮件地址）
 - 有效期（证书的有效时间）
 - 颁发者标识信息
 - 颁发者的数字签名
- ❖ 数字证书由权威公正的第三方机构即CA签发

数字证书

Name: Xiaobao Wei
Serial number: 484865
Issued by: ABC corp CA
Issue date: 2013 01 01
Expiration date: 2013 12 31
Public key: 38ighwejb
Digital Signature: hwefdsaf

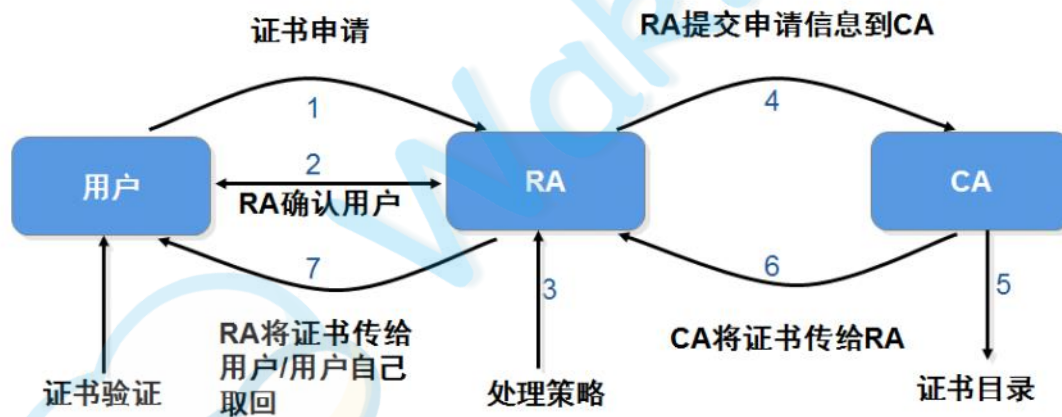


- 网络世界的电子身份证
 - 与现实世界的身份证类似
 - 能够证明个人、团体或设备的身份
- 包含相关信息：
 - 包含姓名、地址、公司、电话号码、Email地址、...
 - 包含所有者的公钥
- 证书的序列号
- 证书的有效期限
- 由可信的颁发机构颁发
 - 比如身份证由公安局颁发一样
- 颁发机构对证书进行签名
 - 与身份证上公安局的盖章类似
 - 可以由颁发机构证明证书是否有效
 - 可防止篡改证书上的任何资料

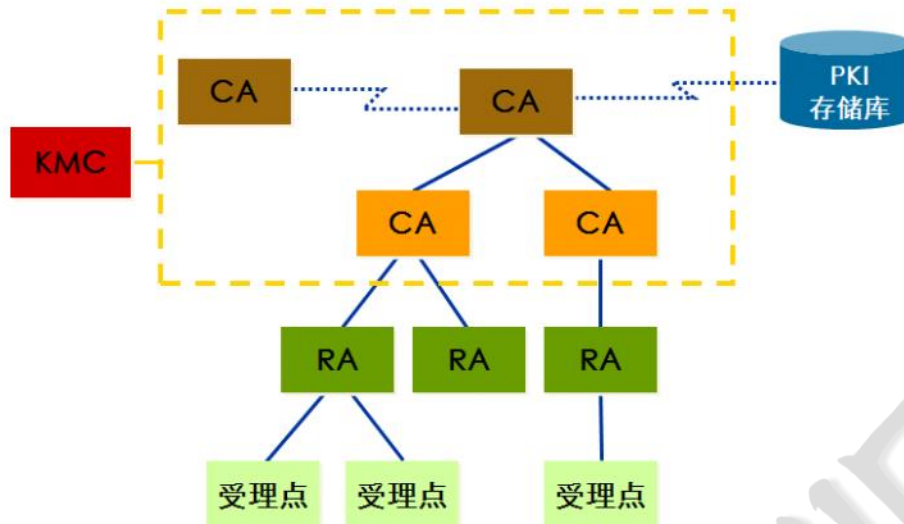
CA的作用

- ❖ Certificate Authority , 证书颁发机构
- ❖ CA的核心功能是颁发和管理数字证书
- ❖ CA的作用
 - 处理证书申请
 - 发放证书
 - 更新证书
 - 接受最终用户数字证书的查询、撤销
 - 产生和发布证书吊销列表 (CRL)
 - 数字证书归档
- ❖ 如果把证书比作**身份证** , CA就是**公安局**

证书的颁发过程



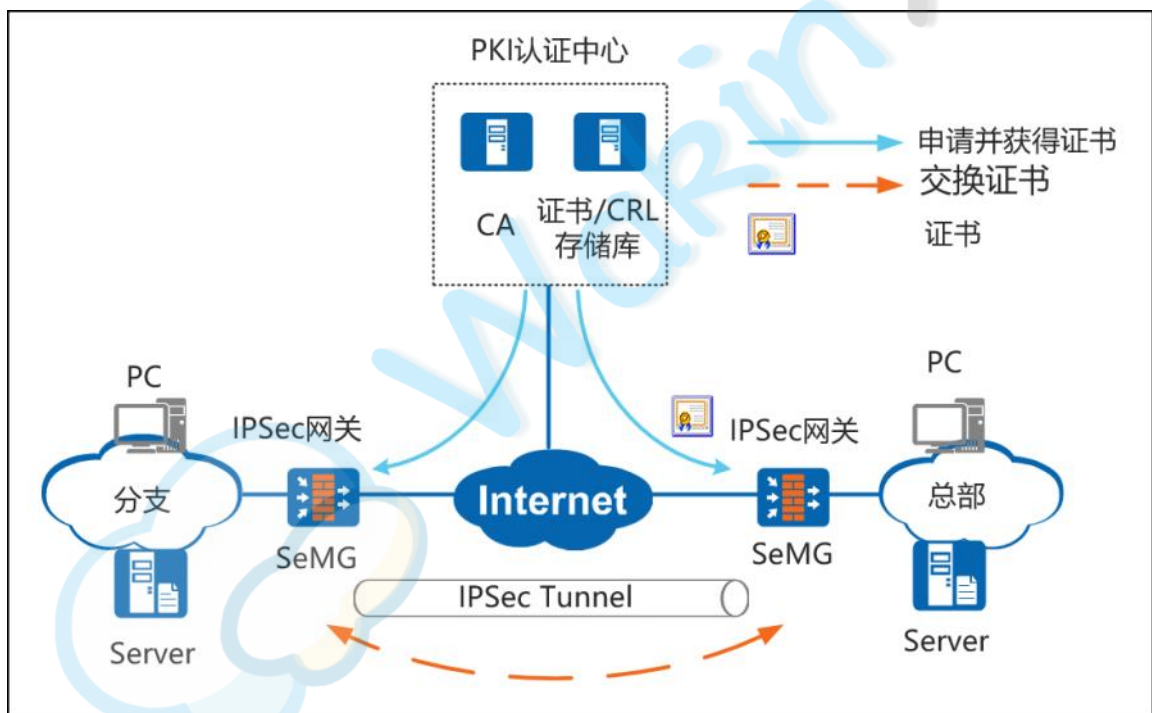
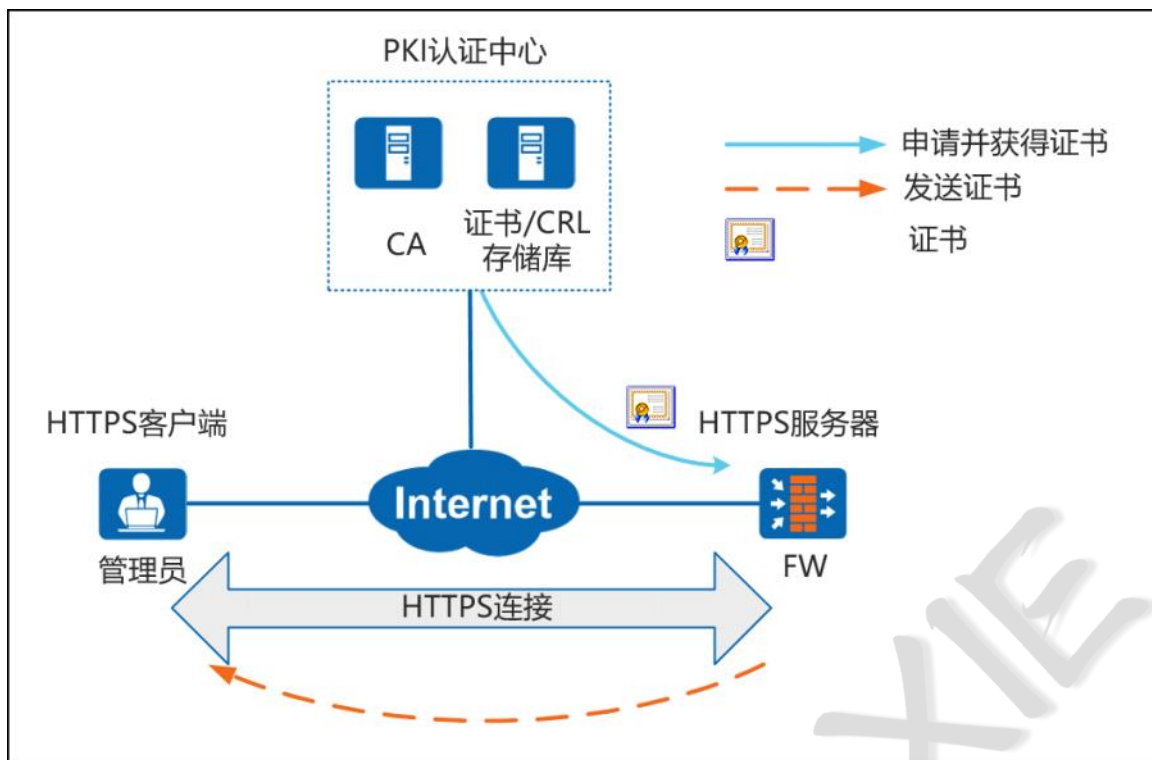
PKI体系结构

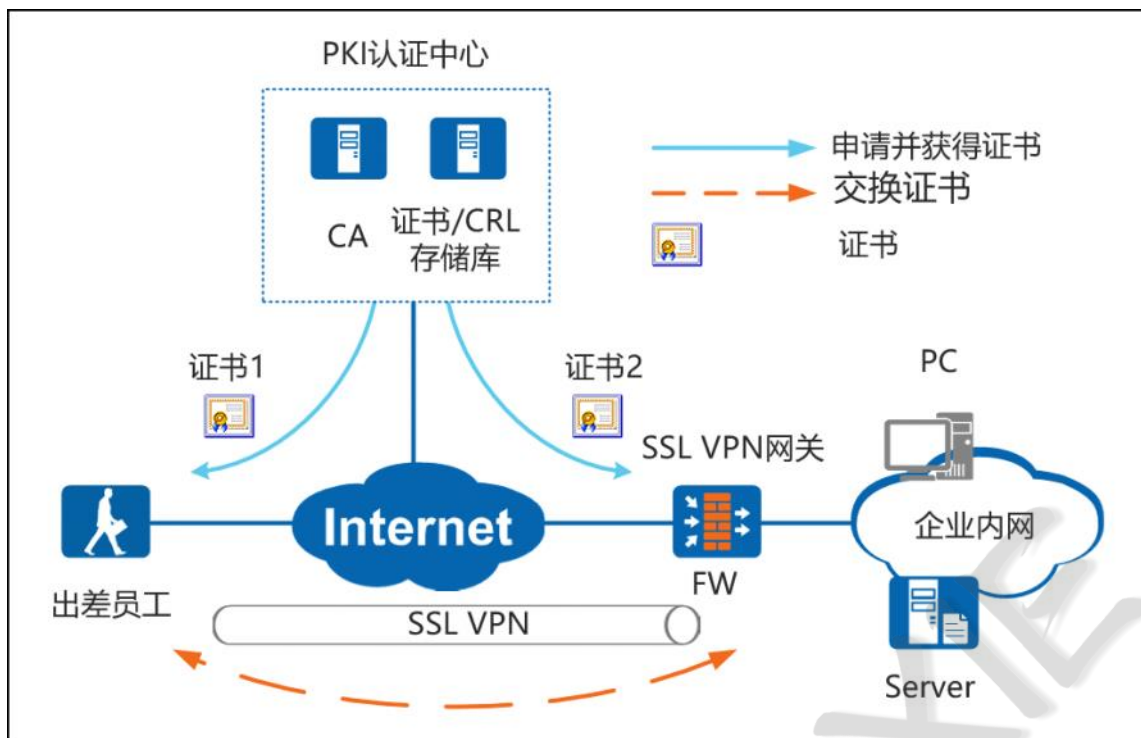


- ❖ CA接受用户的证书请求，签发用户证书，是PKI的核心
- ❖ 接受、验证用户的申请，将验证通过的申请提交给CA，由CA签发证书

PKI协议

- ❖ **SSL (Secure Sockets Layer , 安全套接层)**
 - 认证用户和服务端，确保数据发送到正确的客户机和服务器
 - 加密数据以防止数据中途被窃取
 - 维护数据的完整性，确保数据在传输过程中不被改变
- ❖ **HTTPS (Hypertext Transfer Protocol Secure)**
 - 使用SSL来实现安全的通信
- ❖ **IPSec (Internet Protocol Security)**
 - 目前主流的VPN解决方案





总结:

| 术语 | 备注 |
|------|----------------------------------|
| 数字信封 | 结合对称和非对称加密，保证数据传输的机密性。 |
| 数字签名 | 采用散列算法，保证数据传输的完整性。 |
| 数字证书 | 通过第三方机构（CA）对公钥进行公证，保证数据传输的不可否认性。 |