

16. 计算机系统安全

1



本章相关的参考文献



- Diffie W, Hellman M. New directions in cryptography[J]. IEEE Transactions on Information Theory, 1976, 22(6): 644-654.
- Rivest R L, Shamir A, Adleman L. A method for obtaining digital signatures and public-key cryptosystems[J]. Communications of the ACM, 1978, 21(2): 120-126.
- Boneh D. Twenty years of attacks on the RSA cryptosystem[J]. Notices of the AMS, 1999, 46(2): 203-213.
- Lampson B. Perspectives on Protection and Security[C]// SOSP History Day 2015. Monterey, California.

性能 → 可靠性 → 一致性 → 安全

计算机系统工程的重要目标：

- 用不可靠的组件构建可靠、高效的系统，即：

- 在高可用性的前提下：

- 服务更多用户，存储更大数据量，提供更快速度

→ 第12章 性能

可靠性工程

- 识别、检测、处理错误，防止失效发生

并发、流水线 ...

→ 第13章 可靠性

- 逐步构造通用的抽象模块，逐层构造可靠性

检错、纠错、冗余 ...

一致性目标

- 分布式环境下如何实现？

→ 第15章 分布式原子性与一致性

影子副本、日志、两段锁 ...

已经完善了吗？假如破坏是精心设计的呢？

辨析：安全与可靠性

系统可靠性通常基于概率进行设计

- 而攻击者有能力改变概率分布

例：

- 位错误随机独立分布，出错且通过校验的概率是万亿分之一
- 输入超长导致错误，但只有输入是特定的指令序列才会产生失效
- 并发锁会产生0.1微妙的权限状态失效，迅速恢复
-

计算机系统安全：内涵与界限

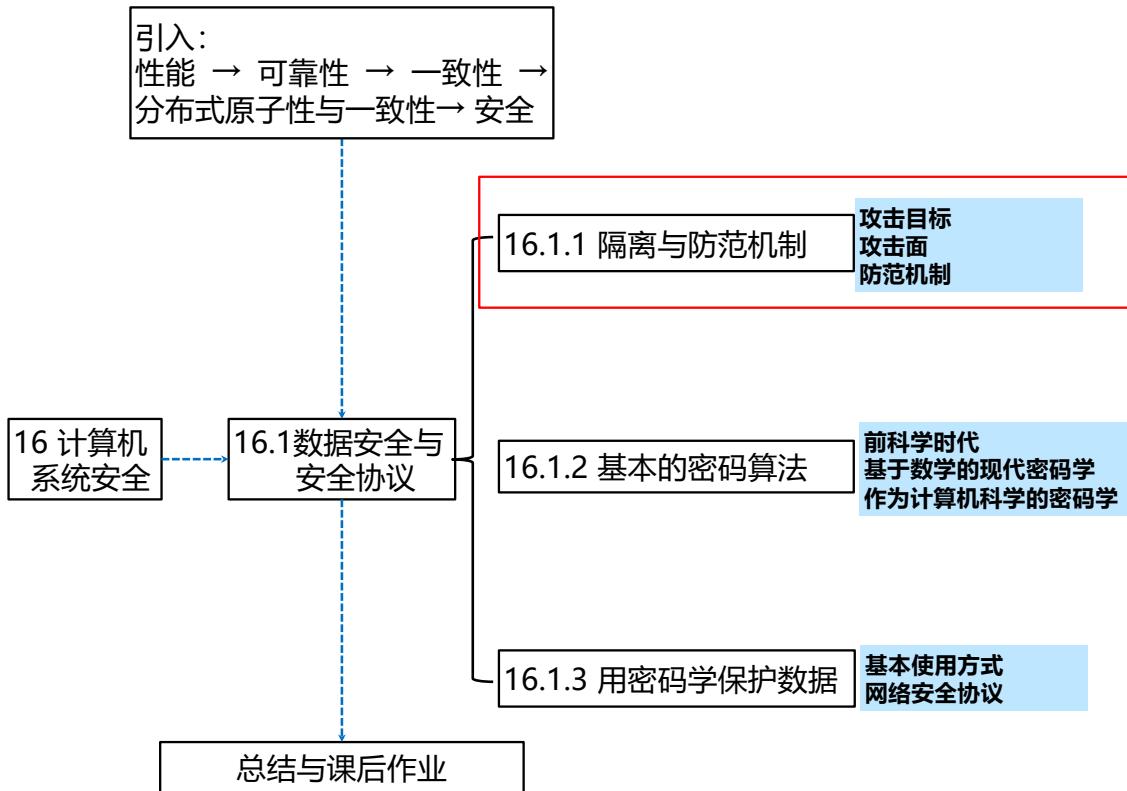
内涵：保护计算机系统、系统数据、用户身份的安全

- 目标：计算机系统不被破坏、数据不被窃取篡改、身份不被假冒
- 16.1 保护数据（今天）
- 16.2 保证身份（下次）
- 16.3 保护软件（下下次）

界限：不涉及“计算机应用于其他领域安全”

- 保护算法、保护隐私、保护社会经济、保护生命财产……
- 不在本课程的考虑之内，可以看做计算机应用学科的主题

16.1 数据安全与安全协议



保护数据：隔离

数据在哪里？如何保护它们？

思考：怎么保护的？

1. 操作系统控制的数据

 · 系统内数据

1. 不安全环境下的数据

 · 系统外数据



传统介质数据

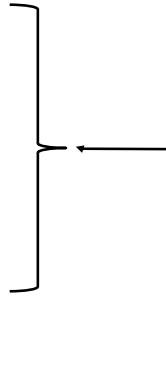
如何做到隔离？

安全基本方法：隔离

隔离数据的几种机制

攻击面分类

- 系统外数据
 - 面向窃听篡改者:
- 系统内数据
 1. 面向远程攻击者:
 2. 面向本地攻击者:
 3. 面向内部攻击者:



安全基本方法：隔离

机制的设计

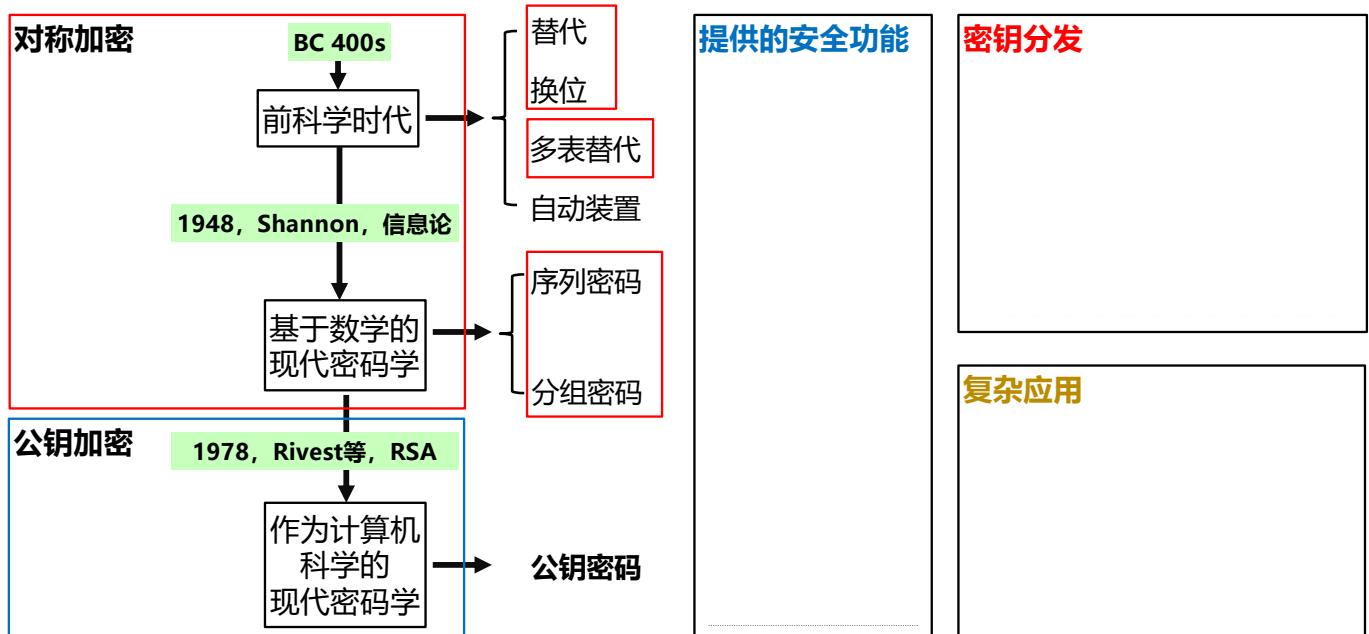
1. 密码算法封装

- 秘密：封锁与开启数据的钥匙
 - 用户：掌握秘密
 - 非用户：不掌握秘密
- 章节与内容：16.1.2 – 16.1.3 密码算法与实际应用

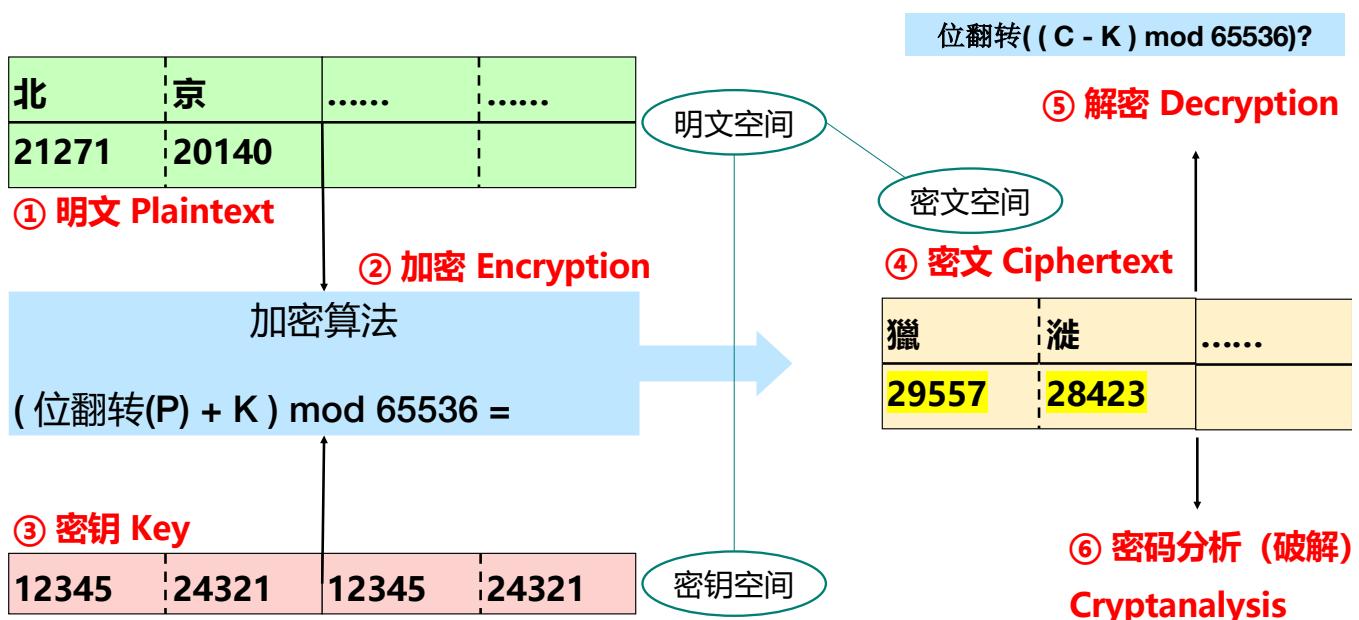
2. 认证与访问控制

- 权限：允许访问的凭证
 - 管理员：高权限
 - 普通用户：低权限
 - 其他人：无权限
 - 细粒度管理：定制权限
- 章节与内容：16.2 认证与访问控制

16.1.2 基本的密码算法



密码学基本术语



前科学时代：基本方法

两个字母的组合有多少个？如何计算？

- 能给我们隐藏信息什么启发吗？

$$P_{26}^2 = \frac{26!}{24!} \text{ 取决于字母集和位置}$$

古典加密两种基本思路

1. 替换
2. 换位

攻击方式：

1. 概率
2. 模式
3. 暴力

$$P(A) * P(B|A) = P(B) * P(A|B)$$

前科学时代： vernam方案

最安全的加密方式

- Vernam方案 (AT&T,1917):

▸ $P \oplus K = C$: 具有完善保密性

- 完善保密性

▸ $\forall p \forall c (p \in P) \wedge (c \in C) \Rightarrow P(p|c) = P(p)$

- 缺点分析：

▸ $\forall p \forall c (p \in P) \wedge (c \in C) \Rightarrow P(c|p) = P(c)$

$\forall p \in P, \forall c \in C \quad P(c|p) = P(c) > 0$

$|K| \geq |C|$

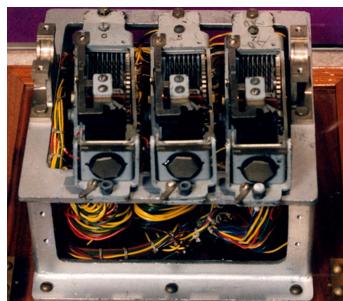
$|K| \geq |C| \geq |P| \quad ! ! !$

$$P(A) * P(B|A) = P(B) * P(A|B)$$

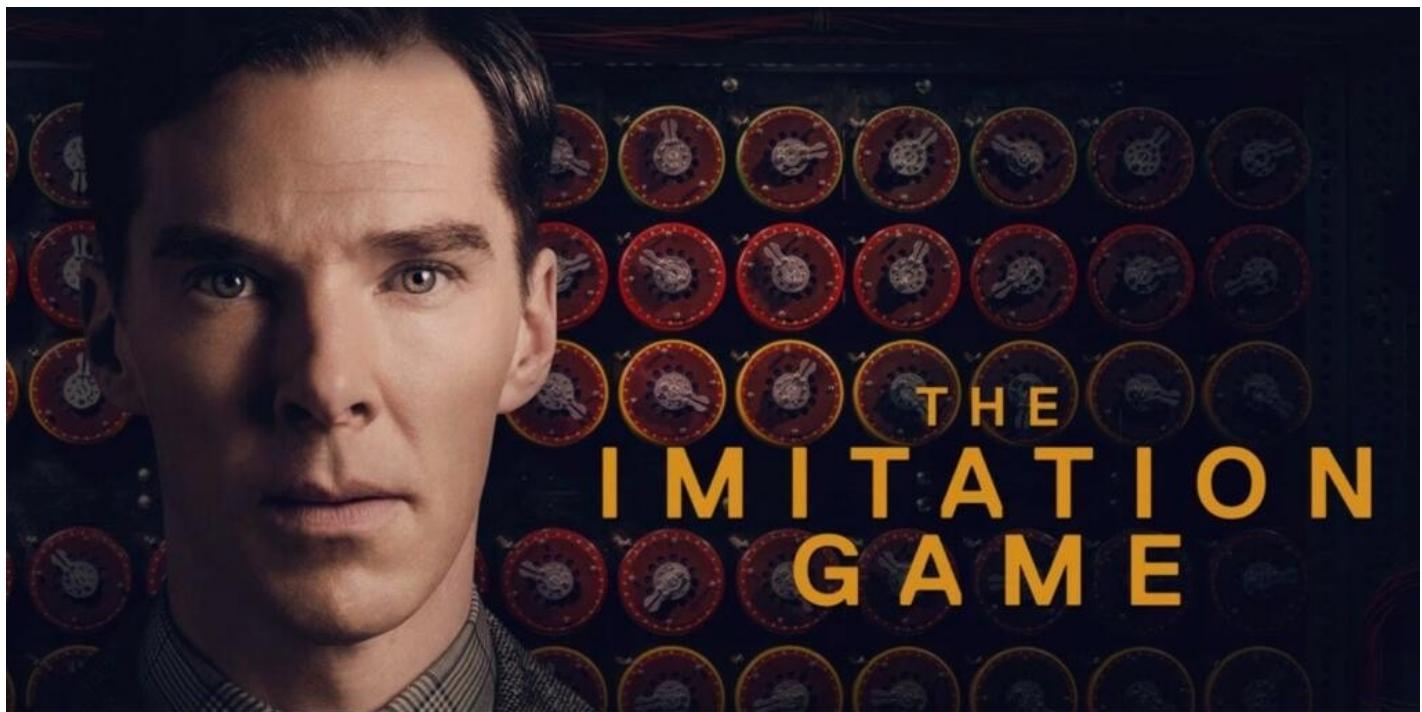
$$P(B|A) = \frac{P(B) * P(A|B)}{P(A)}$$

$$P(A) = \sum P(A|B_i) P(B_i)$$

前科学时代：机械电子装置



模仿游戏 (2014)



现代密码学：对称加密算法

1. 分组加密

- 将消息分成多个分组（如：DES 64位，AES128位）
- 算法建立了消息分组到密文分组之间的替换表（密钥）

为什么叫对称加密？

2. 序列加密

- 能用密钥持续产生比特流的算法，用比特流与消息做异或

期望的性质：

- 知道密钥的用户，能将密文转换成消息
- 不知道密钥，通过密文来猜测明文是计算上不可行的
- 拥有部分明密文对，计算密钥仍然是计算上不可行的

计算上不可行：
对加密的攻击，通常在理论上可以暴力求解。
计算上不可行，意即以当前的计算能力，在一定时间内求解成功概率可忽略不计。

分组加密

初始设置

- 寻找复杂的加密函数： $f(P, K) \rightarrow C$ ，且存在可计算的逆函数 f^{-1}
- 产生密钥 k ，并分发到通信双方

加密过程

1. 将明文分为等长分组 $\langle p_1, p_2, \dots, p_n \rangle$
 - 分组可看做整数
2. 将每个分组 p 用函数 f 和 K 转换成密文 c ：

北京
2127120140	



4379289385, 19393214, ...

解密过程

- 使用 f^{-1} 和 K ，将 $\langle c_1, c_2, \dots, c_n \rangle$ 转换成 $\langle p_1, p_2, \dots, p_n \rangle$

分组加密的弱点

例：

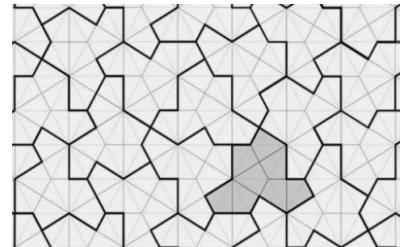
假设分组大小是32bit (2个汉字)

如果：“北京”作为一个分组被映射成“獵灘”

则：“北京”再次作为一个分组时，仍会被映射成“獵灘”

弱点：

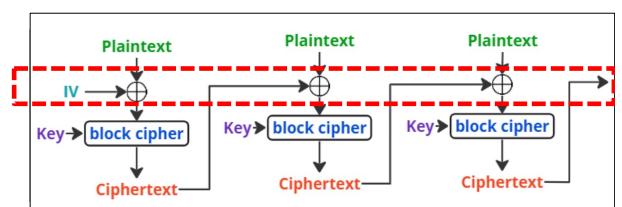
- 存在**可被利用的条件概率**
- **如何消除这一问题？**



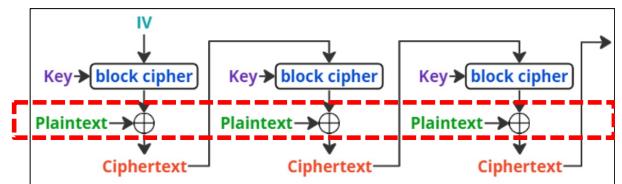
方案：工作模式 (mode of operation)

方案1：分组先与上一组的密文**异或**，再**加密**（密码分组链接CBC）

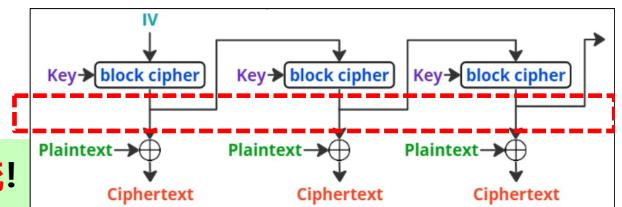
- 收益：同样分组，不同密文
- 缺点：还能并行吗？



方案2：先加密上一组的密文，再与分组**异或**（密码反馈CFB）



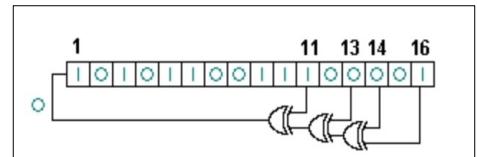
方案3：反复加密某个IV，与分组**异或**（输出反馈OFB）
能持续产生密钥流！



序列加密

初始设置

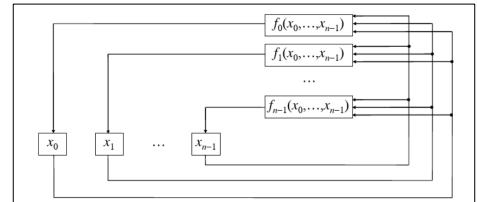
- 算法
- 初始向量 (IV) + 密钥 (key)



linear feedback shift register
LFSR

加密过程

- 使用算法产生持续的密钥比特流
- 将密钥流与消息流异或产生密文流



解密过程

- 解密过程与加密过程相同

特点

- 基于数学方法的安全分析较成熟，使其广泛用于军事、外交等关键领域

对称加密算法的问题

1. 加密密钥必须保密

2. 不能用于证实身份

3. 密钥分发开销大

能否设计一种非对称加密算法，加密和解密使用不同密钥且难以相互推出？

1. 加密密钥无需保密
2. 解密密钥可用于证实身份
3. 加密密钥可以公开分发

RSA公钥加密算法

选择2个大素数： p 和 q

计算密钥：

- $n = pq$
- $\varphi(n) = \text{lcm}(\varphi(p), \varphi(q)) = \text{lcm}(p-1, q-1)$.
 - 欧拉函数：合数——因子的欧拉函数的最小公倍数
- 选择整数e，使 $1 < e < \varphi(n)$ 且 $\text{gcd}(e, \varphi(n)) = 1$ (即二者互素)
- 求解整数d，使 $ed \equiv 1 \pmod{\varphi(n)}$ (即d为e模φ(n)的逆元)

例：

$$\begin{aligned}p &= 61, q = 53, \\n &= 3233, \varphi(n) = 780 \\e &= 17, d = 413\end{aligned}$$

分发密钥：

- 将 (n, e) 公开发布， (n, d) 秘密保存

加密/解密：

- $(m^e)^d \equiv m \pmod{n}$

例：

$$\begin{aligned}m &= 65 \\c &= 2790\end{aligned}$$

1. 系统与复杂性 → 2. 构造抽象计算机系统 → 3. 命名



8. 线程

目前获得图灵奖的唯一华人学者：姚期智

因其对计算理论的奠基性贡献获2000年图灵奖。

包括：伪随机数生成、密码学和通信复杂性。

中国学者何时能够获得图灵奖？任重道远！

9. 网络设计思想

希望大家：

① 多做一些有意义的基础领域的工作，耐得住寂寞。

② 多做一些面向计算机本原的工作，敢于面对挑战。



块化



可靠性



2004

16. 计算机系统安全 ← 15. 分布式系统 ← 14. 原子性与一致性



2012



2002



2015



2014



1998

16.1.2 用密码学保护数据

① 基本使用方式

② 用于通信协议

① 基本使用方式：消息加密

初始设置

- Alice 产生公私钥 $\langle K_{\text{pub}}, K_{\text{priv}} \rangle$, 将 K_{pub} 公开

1. 保护机密性

- 发送者 Bob 产生随机的对称加密算法的密钥 K
- 使用 K 对消息 m 进行加密 $\{ m \}_K$
- 使用 K_{pub} 对 K 进行加密 $\{ K \}_{K_{\text{pub}}}$
- Bob 发送 $\{ m \}_K$ 和 $\{ K \}_{K_{\text{pub}}}$ 给 Alice
- 为什么要这样做？

① 基本使用方式：消息鉴别

初始设置

- Alice 产生公私钥对 $\langle K_{\text{pub}}, K_{\text{priv}} \rangle$, 将 K_{pub} 公开

2. 保护完整性

- Alice 求 消息 m 的哈希值 $h = H(m)$
- 使用 K_{priv} 对 h 进行加密 $\{ h \}_{K_{\text{priv}}}$
- Alice 发送 m 和 $\{ h \}_{K_{\text{priv}}}$ 给 Tom、John、Mike
- 为什么可以保护完整性？

② 将加密算法用于协议

挑战：

1. 如何选择相同的算法、参数？
2. 如何产生和分发密钥？
3. 如何协同工作？
 - › 不加密→加密的转换
 - › 问题报告
4. 如何传输数据？
5. 如何感知对方存在（有连接）？

用安全协议保护数据传输

数据传输的安全需求

1. 数据机密性
2. 数据完整性
3. 身份鉴别

SSL/TLS协议



安全传输协议SSL/TLS

1994年，Netscape公司发布了SSL2.0协议

- 微软在SSL2.0基础上发布了PCT协议

1996年，Netscape公司发布了SSL3.0协议

1997年，IETF基于SSL3.0发布了TLS1.0协议

- 微软放弃PCT，开始支持TLS标准

1999年，RFC2246发布（TLS协议 1.0）

2006年，RFC4346（TLS 1.1）

2008年，RFC5246（TLS 1.2）

2018年，RFC8446（TLS 1.3）



发明了：

HTTP cookie

JavaScript

SSL

jar

Mozilla基金会：

Firefox

ThunderBird

协议设计

挑战:

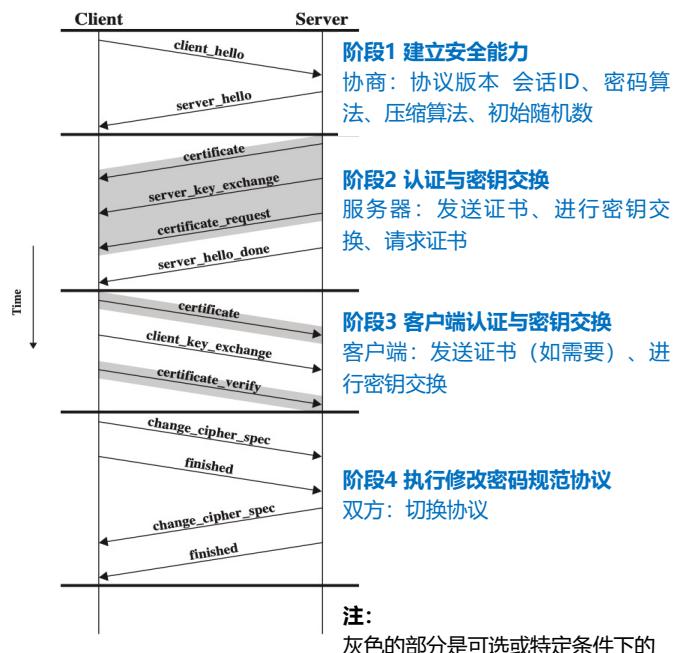
1. 如何选择相同的算法、参数? → **握手协议 (handshake)**
2. 如何产生和分发密钥?
3. 如何协同工作?
 - › 不加密→加密的转换
 - › 问题报告→ **更改密码规范协议**
4. 如何传输数据? → **警告协议**
5. 如何感知对方存在(有会话)? → **记录协议**
-
-
-
-

握手协议

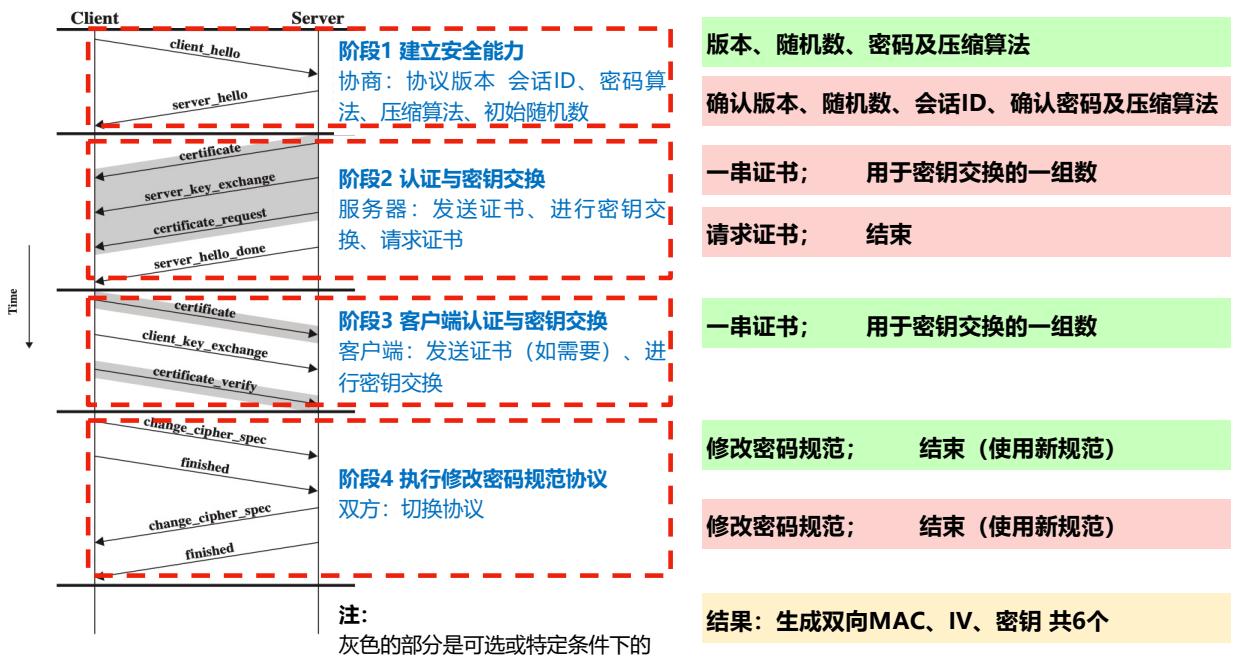
用于

- 进行互相认证
- 协商算法参数
- 启动其他协议

四阶段消息交换



四阶段消息交换



握手协议的安全性

握手协议基本使用明文

安全机制

- finished包中：
 - $PRF(master_secret, finish_label, HASH(handshake_messages))$
- 可以发现内容篡改

修改密码规范协议 (change cipher spec protocol)

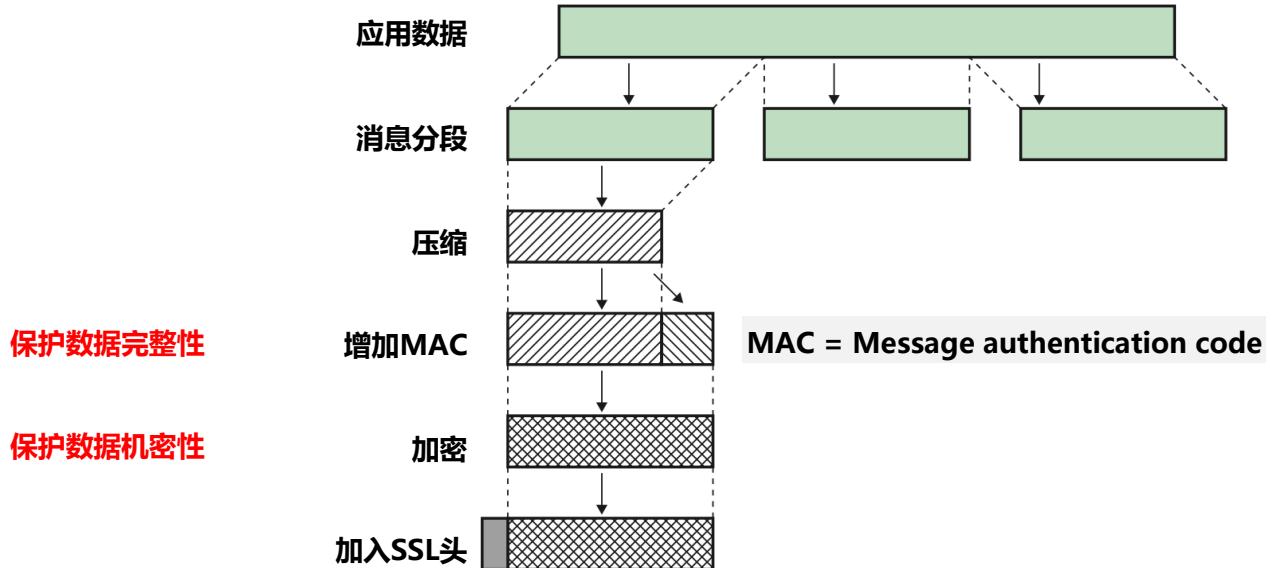
- 使用记录协议通信
- 仅包含1字节: payload=1
- 将挂起状态改变为当前状态 (原子操作)
- 更新密码算法组

TLS会话与TCP连接

TLS会话

1. 使用握手协议建立TLS会话
2. 建立在TCP连接之上
3. 可以重用于多次连接
4. 客户端-服务器之间的单向关联
5. 规定了密码相关的参数

记录协议 (record protocol)



安全性问题

1. 身份的安全

2. 算法与协议的安全

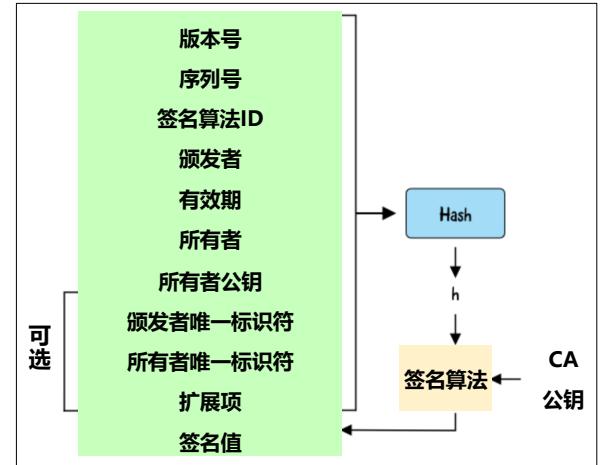
鉴别服务器身份

朴素的想法：

- amazon.com Inc发布公钥并签署其所有网站，以此类推
- 问题：公钥分发的公信力怎么解决？

成熟的方法：

- 请知名证书机构(CA)签发公钥
 - 有多知名？
- 浏览器怎么判断CA的身份？
- 证书格式：ITU X.509(ISO 9594-8)



鉴别用户

多种方法：

- 证书
 1. 与服务器鉴别相同的方法
 2. 机构内部的CA与证书
 3. 手工生成并部署 (例：github, git clone private repo)
- IP地址或口令
 - 不安全
 - cookie (导致用户信息安全问题)

信任是安全的基石

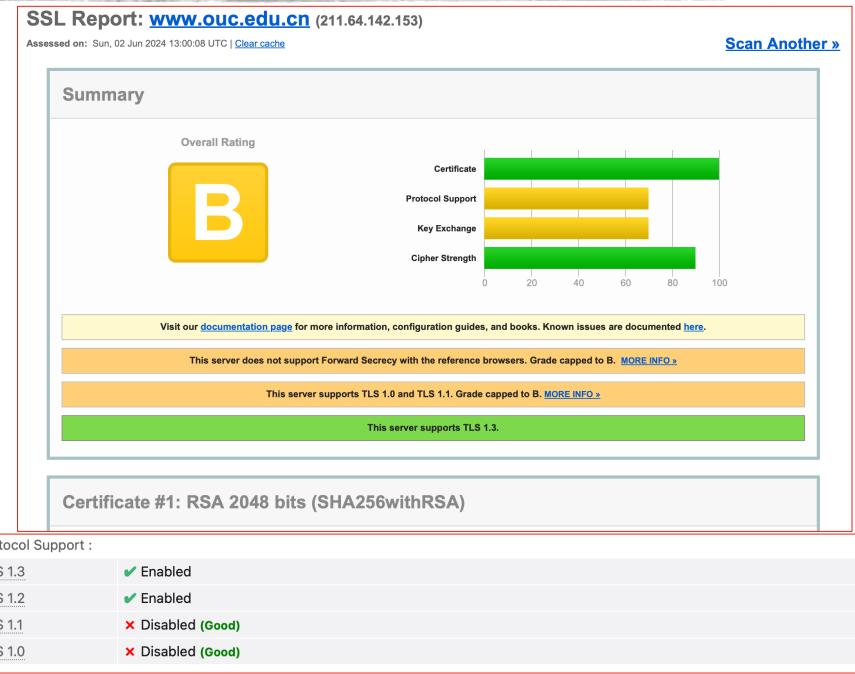
Blocking Trust for WoSign CA Free SSL Certificate G2

Certif
for th
trustee
to est
In light
trust
To av
Apple
publis
until t
As the
produ
Further
After
of the
We ar
certifi
00:00
Update - April 1: As a result of a joint investigation of the events surrounding this incident by Google and CNNIC, we have decided that the CNNIC Root and EV CAs will no longer be recognized in Google products. This will take effect in a future Chrome update. To assist customers affected by this decision, for a limited time we will allow CNNIC's existing certificates to continue to be marked as trusted in Chrome, through the use of a publicly disclosed whitelist. While neither we nor CNNIC believe any further unauthorized digital certificates have been issued, nor do we believe the misissued certificates were used outside the limited scope of MCS Holdings' test network, CNNIC will be working to prevent any future incidents. CNNIC will implement Certificate Transparency for all of their certificates prior to any request for reinclusion. We applaud CNNIC on their proactive steps, and welcome them to reapply once suitable technical and procedural controls are in place.

算法与协议的安全

Version	Additional Monitoring	Traffic Response	Asset Response	
SSL 2.0	N/A	Immediate block	Disable service until reconfigured to only support TLS 1.2 and TLS 1.3.	
Cipher Suite	Additional Monitoring	Traffic Response	Asset Response	
NULL	N/A	Immediate block	Disable or quarantine until reconfigured.	
RC2	N/A	Immediate block	Disable or quarantine until reconfigured.	
Key Exchange Method	Additional Monitoring	Traffic Response	Server Response	Client Response
ANON	N/A	Immediate block	Disable or quarantine until reconfigured.	Disable or quarantine until reconfigured.
EXPORT	N/A	Immediate block	Disable or quarantine until reconfigured.	Disable or quarantine until reconfigured.
RSA with keys < 1024 bits (common sizes 512, 768)	N/A	Immediate block	Disable or quarantine until reconfigured. Install enterprise approved certificate.	Review configuration to ensure it is up to date. Reconfigure as necessary.
DHE with keys < 1024 bits (common sizes 512, 768)	N/A	Immediate block	Disable or quarantine until reconfigured.	Review configuration to ensure it is up to date. Reconfigure as necessary.
ECDHE with custom curves	N/A	Immediate block	Disable or quarantine until reconfigured.	Reconfigure to only offer recommended curves.
RSA with keys between 1024 and 2048 bits (common sizes 1024, 1536)	N/A	Detect/Block ¹³	Reconfigure or update to support 3072 bit RSA, DHE with 3072 bits, and/or ECDHE with p384. Install enterprise	Reconfigure or update to support 3072 bit RSA, DHE with 3072 bits, and/or ECDHE with p384.

服务器/浏览器的安全设置



警告协议 (Alert Protocol)

发送安全警告信息

- 字节1：
 - 1表示普通警告
 - 2表示致命警告 (**立即停止**)
- 字节2：
 - 警告代码

心跳协议 (Heartbeat Protocol)

定期发送，表明协议实体的可用性，2012年在 RFC 6250中增加。
在握手协议中，协商是否使用（支持）

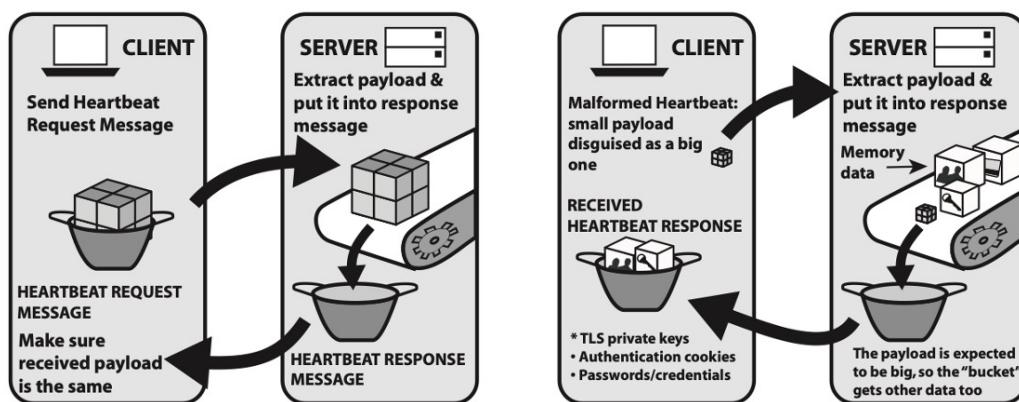
功能：

- 确认存活性
- 在连接空闲时产生消息活动

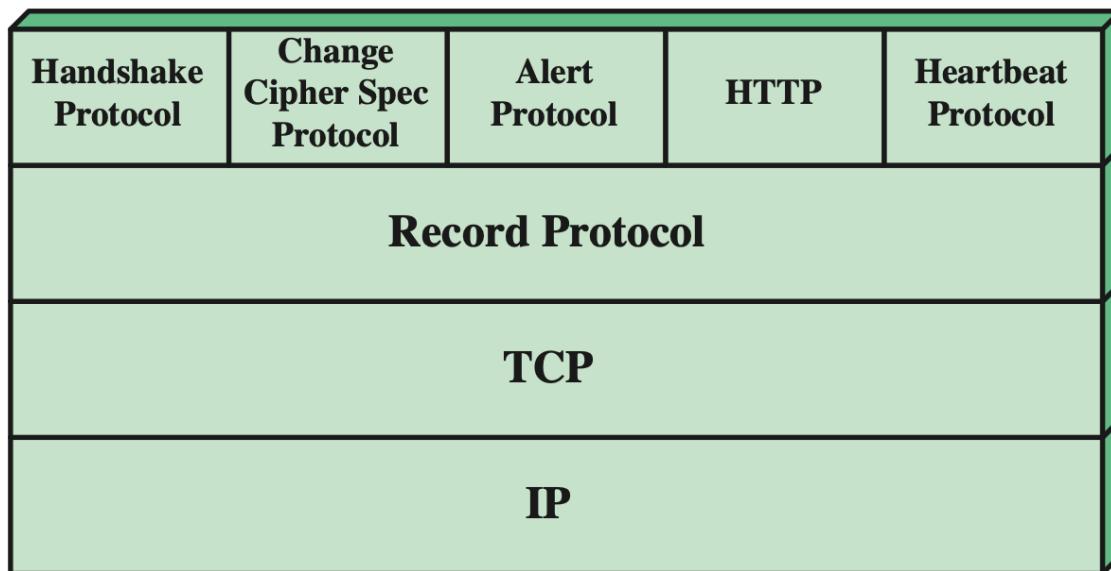
心脏滴血(Heartbleed)攻击

心脏滴血 (openssl: CVE-2014-0160)

- 回复包使用memcpy复制请求包中长度为"payload"的payload



TLS 协议组成



重要术语中英文对照



明文: Plaintext
密文: Ciphertext
密钥: Key
加密: Encryption
解密: Decryption
机密性: Confidentiality
完整性: Integrity
认证: Authentication
授权: Authorization
审计: Audit
潜信道: Covert Channel

自主访问控制: DAC
强制访问控制: MAC
基于角色的访问控制: RBAC
策略: Policy
访问控制列表: ACL
信息流控制: Flow Control
剩余信息: Residue
证书机构: CA
工作模式: mode of work



本章相关的参考文献



W. Diffie and M. Hellman, New directions in cryptography[J]. IEEE Transactions on Information Theory, 1976, 22(6): 644-654.

Rivest R L, Shamir A, Adleman L. A method for obtaining digital signatures and public-key cryptosystems[J]. Communications of the ACM, 1978, 21(2): 120-126.

Boneh D. Twenty years of attacks on the RSA cryptosystem[J]. Notices of the AMS, 1999, 46(2): 203-213.

Butler Lampson. 2015. Perspectives on Protection and Security[C]//SOSP History Day 2015. Monterey, California.

16.1 结束

