# 加密实践与商业创新

刘智

cowliucd@gmail.com

2017/01

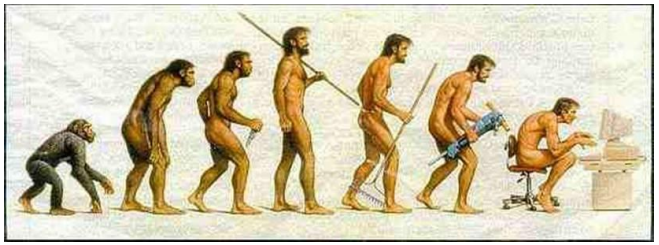
# 关于我

- a security  and crypto zealot
- 电子科技大学信息安全博士(2009-2013)
- 纽约州立大学石溪分校安全研究员(2010-2012)
- 多年安全系统研发经验，精通安全攻防和加密技术
- 专注安全和加密技术及商业创新
- 擅长构建安全产品、设计安全架构

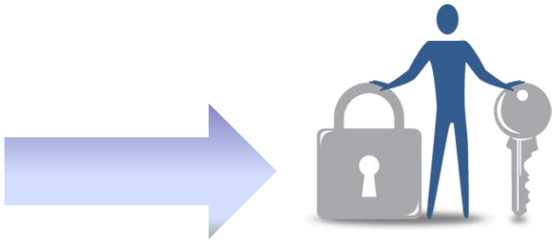
# 大纲

- 加密的应用与意义
- 常见加密算法与陷阱
- https深入分析
- 最佳实践
- 加密应用创新

# 从古代到现在



人类历史



数据/个人隐私保护

Encryption - “the armor in digital age”

# 加密的应用

数据安全

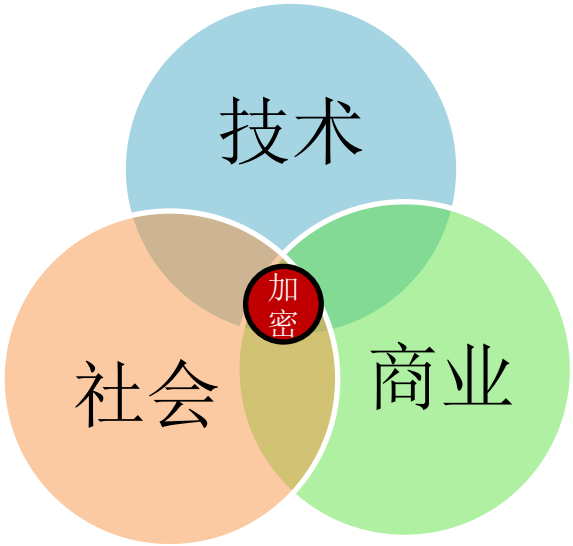
身份认证

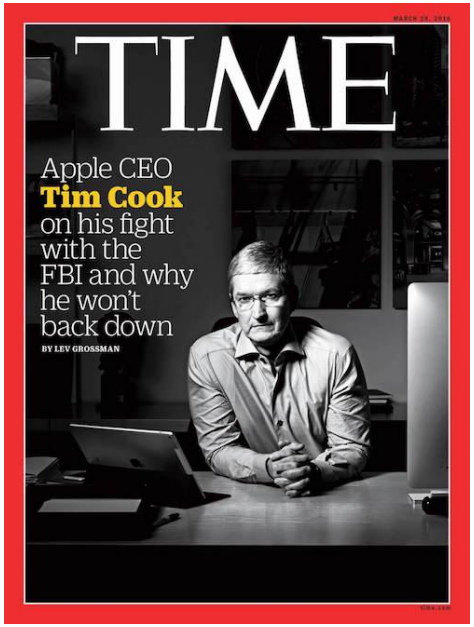
匿名化

区块链

**加密提供无限想象空间！**
**Encryption is where amazing happens**

# 加密的现实意义




苹果 vs FBI

# 加密的现实意义


前国务卿Colin Powell 的邮件：

**From:** CP >
**Sent time:** Sat, 26 Jul 2014 21:46:21 -0400
**To:** Jeffrey Leeds >
**Subject:** Re: Corinthian

I would rather not have to vote for her, although she is a friend I respect. A 70-year person with a long track record, unbridled ambition, greedy,not transformational, with a husband still dicking bimbos at home (according to the NYP).

# 大纲

- 加密的应用与意义
- 常见加密算法与陷阱
- **https**深入分析
- 最佳实践
- 加密应用创新

# 加密算法的问题

“In crypto, I think the biggest weakness right now is not the algorithms, but the implementation. That gets far too little attention. Most people don't even know what authenticated encryption is, and end up implementing something poorly themselves.”

— ***David Brumley***

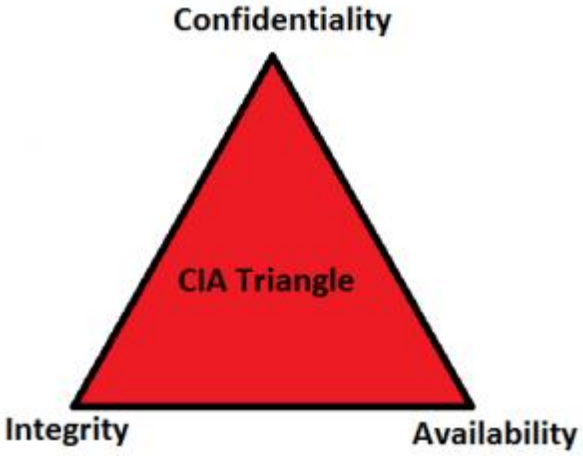
# 安全专家vs密码专家

“Unfortunately, the computer security and cryptology communities have drifted apart over the last 25 years. Security people don’t always understand the available crypto tools, and crypto people don’t always understand the real-world problems.”

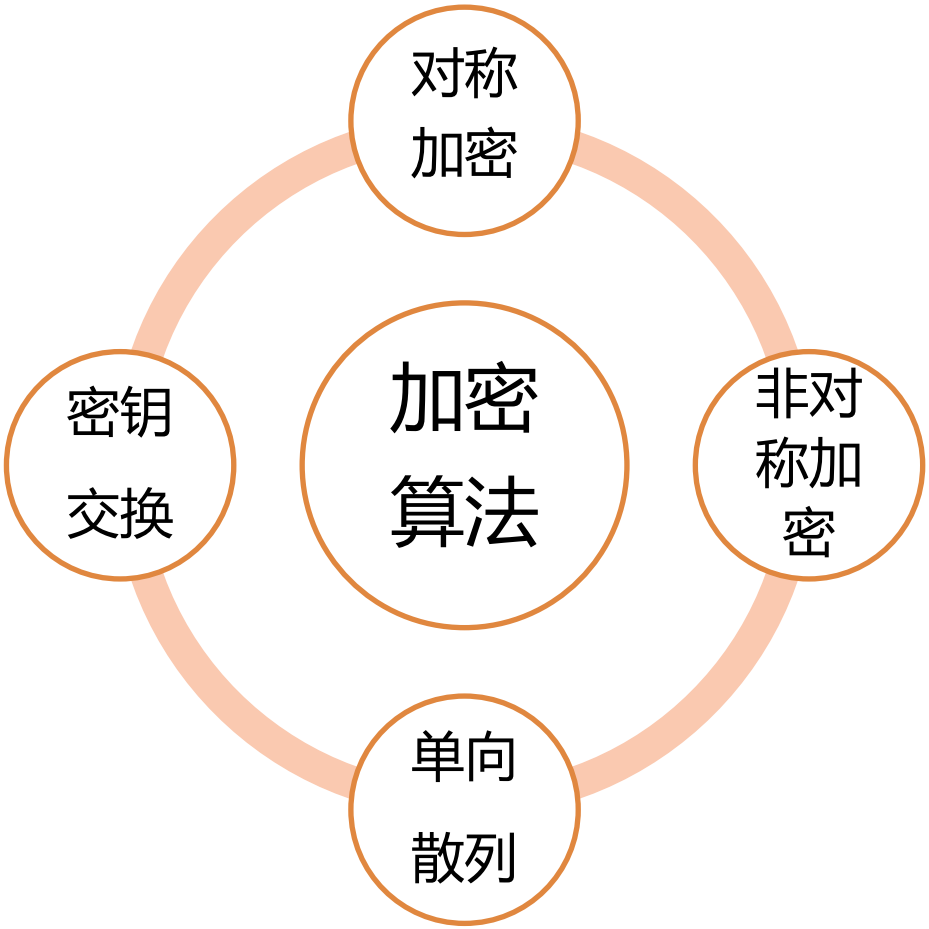
— ***Ross Anderson***

# 安全要素

- 保密性(Confidentiality)
- 完整性(Integrity)
- 可用性(Availability)
- 身份可认证性(Authentication ability)



# 加密算法



# 对称加密(1)

- **加密和解密共用一个密钥**
- **常见对称加密算法: AES, TEA**
- **速度快，但密钥保存困难**
- **常见陷阱**
  - 密钥长度不够，推荐1024bit或更高
  - 算法不安全，如DES
  - **明文存储密钥**
  - 使用弱随机数，攻击者很容易猜测

# 对称加密(2)

## 高级话题

- **对齐处理**
  - 加密数据长度须为密钥长度整数倍, 不足则补齐
- **AES加密模式**
  - ECB分组加密、CBC流加密、GCM高级加密
  - ECB最简单但安全性低
  - CBC考虑初始向量保存
  - GCM已成为主流标准，安全性较高、速度快

# 非对称加密(1)

- **一对密钥**(公钥, 私钥)：公钥公开，私钥自己保存
- **加密和签名的用法**
  - 加密：对方公钥加密
  - 签名：自己私钥加密、公钥验签
- **使用场景**
  - **仅加密短字节，签名速度慢**、验签很快(签名比验签慢数百倍)
- **对称加密和非对称加密通常混合使用**(见CipherSuite)
- **常见非对称加密算法: RSA, EC(椭圆曲线)**
  - RSA较为成熟，EC速度更快
  - 达到相同安全性EC密钥长度更短(区块链使用EC 的原因)

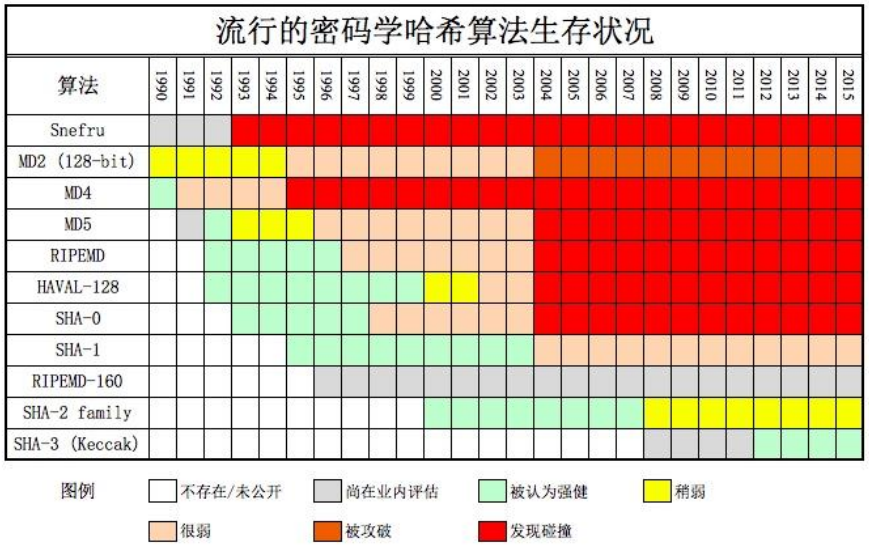
# 非对称加密(2)

常见陷阱

- **加密/签名大量字节**
  - 几何倍数的性能下降，通常只用于加密密钥或随机数
  - 频繁加密导致性能急剧下降
- **RSA密钥强度不够**(推荐1024/2048bit)
- **对齐**
  - PKCS15/OAEP, 加密短字节不存在此问题
- **加密和签名术语混用**
- **各种密钥格式转换**
  - pem, der, JKS等

# 单向hash

- **接收方很容易验证信息完整性，速度快**
- **使用最简单但应用范围广泛的算法**
  - 比特币使用的主要技术
- **常见误区**
  - 使用MD5等不安全算法，建议使用SHA256
  - 密码存储使用 hash+salt，防止彩虹表攻击



# 密钥交换

- **Diffie-Hellman(DH)算法**
  - 双方在不共享任何秘密的情况下协商出一个密钥
  - SSL/https基石，2015图灵奖授予DH发明者
- **适合一次一密场景(如https)**
- **密钥交换算法**
  - DH, RSA, ECDH(EC+DH), …
  - RSA也可安全传输密钥，DH更简洁

# 大纲

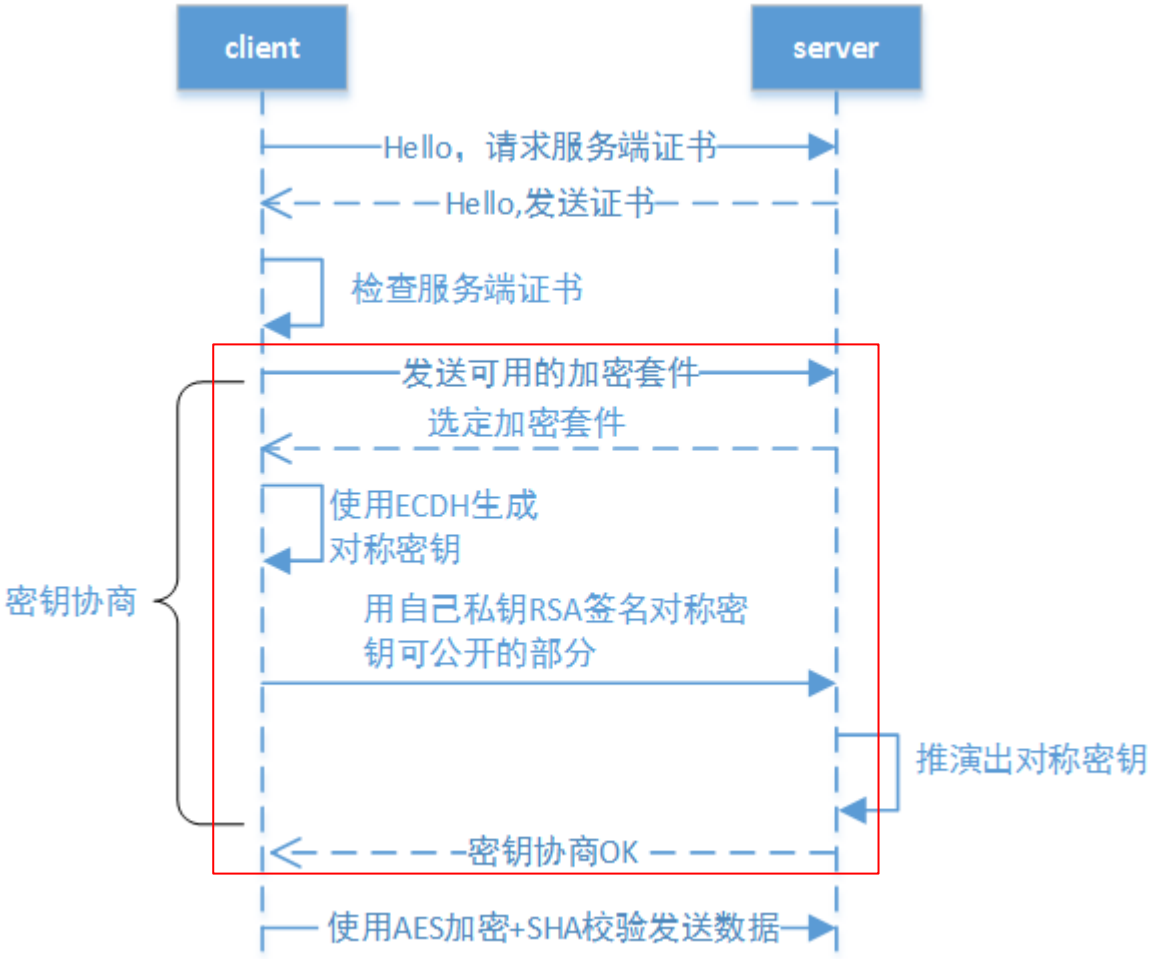
- 加密的应用与意义
- 常见加密算法与陷阱
- **https深入分析**
- 最佳实践
- 加密应用创新

# https深入分析

## https核心功能

- **https = http + SSL, TLS是SSL3.0后续分支**
  - 目前主要使用TLS
  - 大部分基于openssl代码实现，也有其它实现
- **每次会话(session)进行握手连接**
  - https最耗时部分，也是https精髓
- **提供认证/加密/数据校验**
  - 各种组合称为*CipherSuite*(加密套件)

# https握手流程




(本图为原创)

# 大纲

- 加密的应用与意义
- 常见加密算法与陷阱
- **https**分析
- 最佳实践
- 加密应用创新

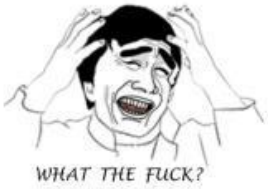
# 最佳实践(1)

“黄金法则”- 摘自《现代加密应用指南》

- **不要尝试自己编写加密算法**
  - 加密算法原理和实现十分复杂，很难穷举测试
- **不要尝试自己设计加密协议**
  - 未经严格验证的协议存在漏洞

# 最佳实践(2)

- **明确各类算法使用场景**
- **加密强度与性能是一对天然矛盾**
- **密钥安全保护**
  - 对称密钥一次一密, 私钥使用ACL(如SeLinux)/代码混淆保护
  - 加壳/混淆通过静态/动态分析仍可破解
- **跨平台算法实现存在差异**
  - 操作系统：Windows, Android, iOS
  - 语言： C, C++, Java, Python, …
  - 即使同一语言的不同版本实现也有差异(如x86 Java和Android， Java JCE与BouncyCastle库)



# 最佳实践(3)

- **国密算法**(SM)
  - 基于椭圆曲线原理
  - 标准已公开，源码需自己全部实现
  - 相比国际通用算法RSA/EC有更好安全性**(?)**
- **专利/许可证**
  - 注意第三方加密库代码license
  - 通常开源代码license较宽松

# 大纲

- 加密的应用与意义
- 常见加密算法与陷阱
- **https**分析
- 最佳实践
- 加密应用创新

# 加密应用创新

- 匿名化
- 数据安全
- 隐私保护
- 区块链
- **IoT**

# 几点思考

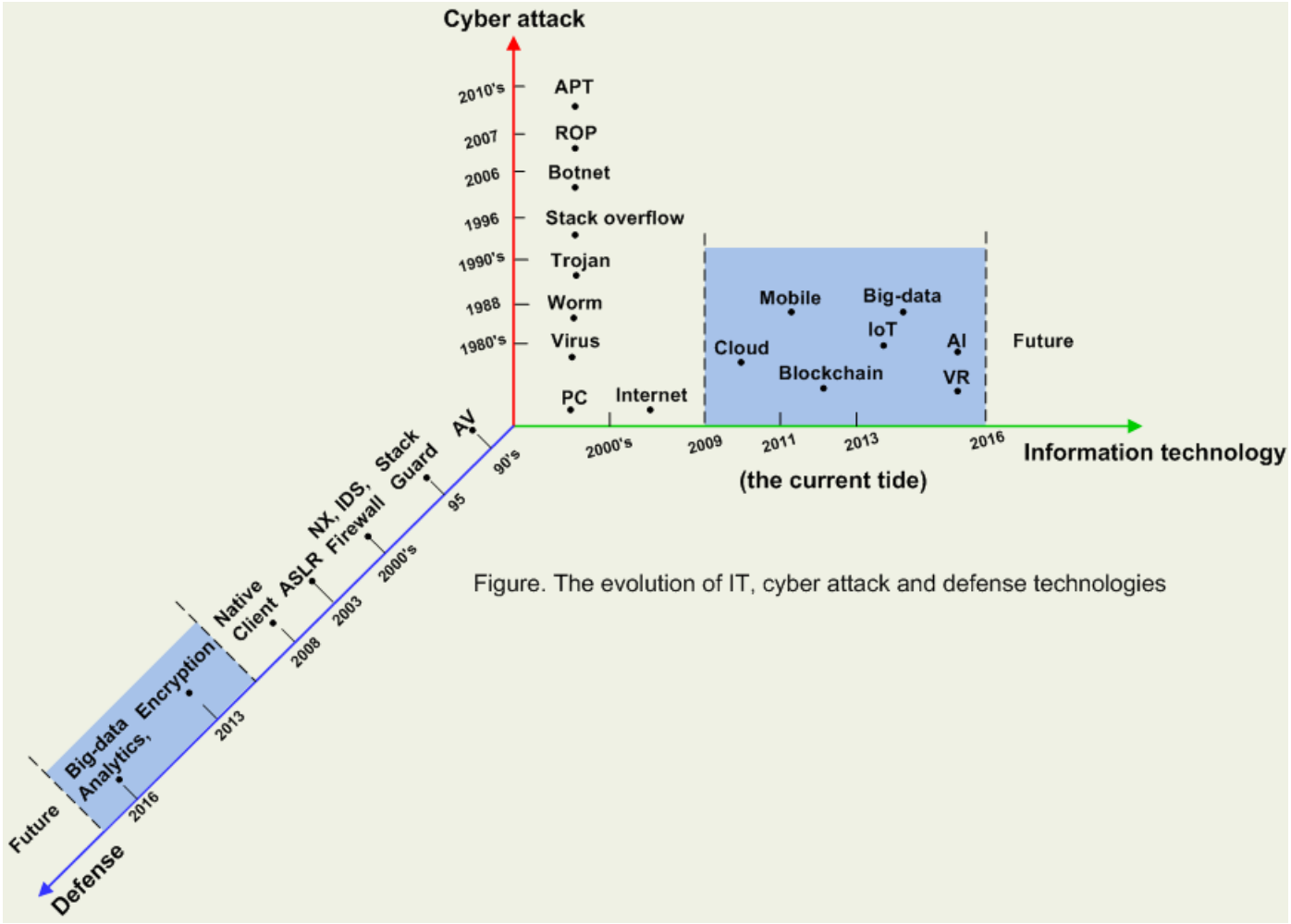



Figure. The evolution of IT, cyber attack and defense technologies

# 几点思考

- 互联网共享精神与加密的关系
- 技术并不保证好的商业模式/用户
- 区块链是良药还是止痛剂？
  - 要解决的核心问题是什么？
  - 使用其它技术能否达到?

# 几点思考

## 什么才是叫“叫好又叫座“的创新？

| Problem | Technique | Interesting problem (factor 3) | Technical approach (factor 5) | Implementation simplicity (factor 7) | Practical value (factor 10) | Impact |
|---|---|---|---|---|---|---|
| Crypto | End-to-end encryption | 4 | 3 | 3 | 3 | |
| | HTTPS | 5 | 5 | 4 | 5 | |
| | Hash-cash | 3 | 3 | 5 | 2 | |
| | Blockchain | 5 | 4 | 3 | 5 | |
| | Zcash | 5 | 4 | 3 | 4 | |

# “Stay paranoid, stay inquisitive”

***Matthew Dickson*, 微软资深产品经理**

# 参考资料

- **文学读物**
  - 《失控》、《千禧年三部曲》
- **常见加密算法库**
  - Openssl, Polarssl(mbedtls), Nacl(读”salt”)
  - 国密SM: SM2标准，算法库GmSSL
- Applied Crypto Hardening(非常好的加密介绍资料)
- 现代密码学实践指南(适合进阶阅读)

# Copyrights

- 本文仅代表作者个人立场
- 欢迎交流
  - 联系方式：cowliucd@gmail.com, 微信yurenliu
  - 个人公众号(见二维码)
- 您可任意复制、转载本文
  - 转载时若能联系作者，十分感谢

# Q&A