我们在密码技术研究上的进展

主讲：23 电子信息-向嘉豪

2025-09-17

目录

- 密码研究背景
- 密码研究热点
- 论文书写投稿

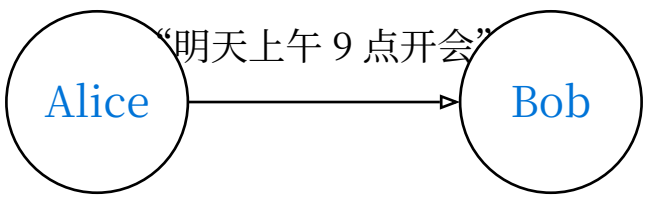
密码研究背景

Alice 与 Bob 的通信困境

想象这样一个场景：Alice 想要向 Bob 发送一条重要的消息

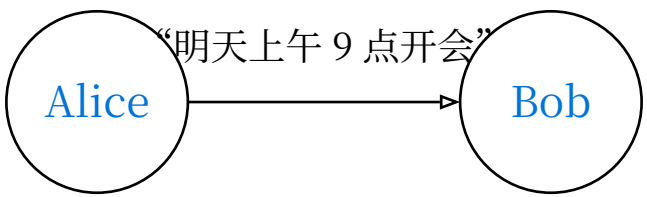
Alice 与 Bob 的通信困境

想象这样一个场景：Alice 想要向 Bob 发送一条重要的消息



Alice 与 Bob 的通信困境

想象这样一个场景：Alice 想要向 Bob 发送一条重要的消息



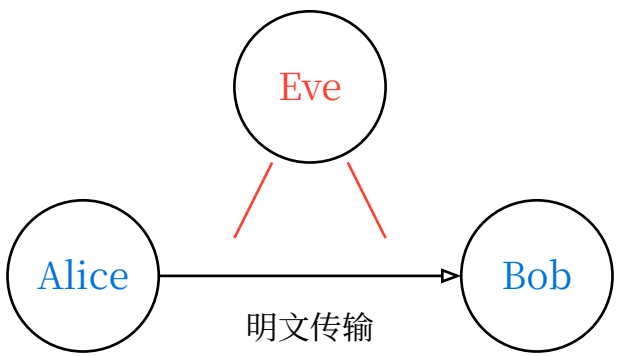
但是，在开放的网络环境中，这条消息面临着多重威胁…

Eve 的窃听威胁

窃听者 Eve 可能截获并读取 Alice 和 Bob 之间的通信

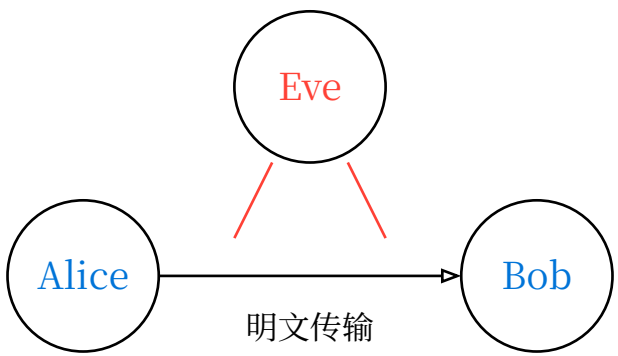
Eve 的窃听威胁

窃听者 Eve 可能截获并读取 Alice 和 Bob 之间的通信



Eve 的窃听威胁

窃听者 Eve 可能截获并读取 Alice 和 Bob 之间的通信



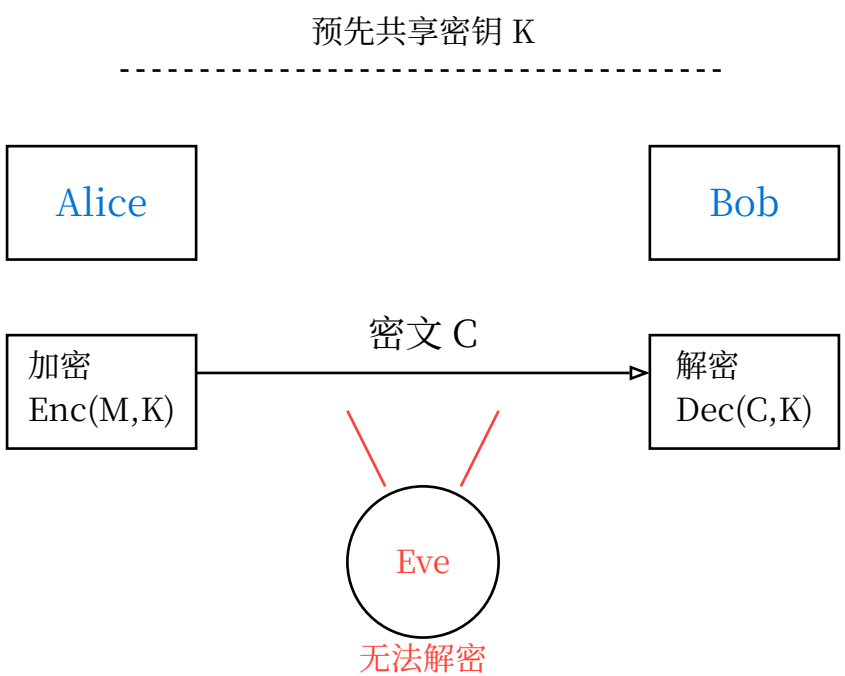
问题 1：机密性丢失 - Eve 可以读取所有通信内容

对称密码解决方案

Alice 和 Bob 事先共享一个密钥 K，使用相同密钥进行加密和解密

对称密码解决方案

Alice 和 Bob 事先共享一个密钥 K，使用相同密钥进行加密和解密

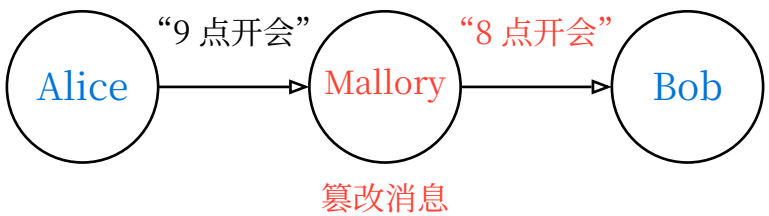


Mallory 的篡改威胁

更危险的是，恶意攻击者 Mallory 不仅能窃听，还能修改消息内容

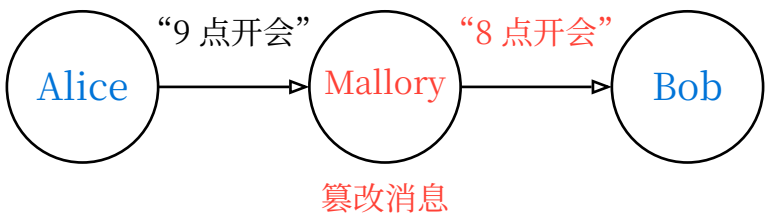
Mallory 的篡改威胁

更危险的是，恶意攻击者 Mallory 不仅能窃听，还能修改消息内容



Mallory 的篡改威胁

更危险的是，恶意攻击者 Mallory 不仅能窃听，还能修改消息内容



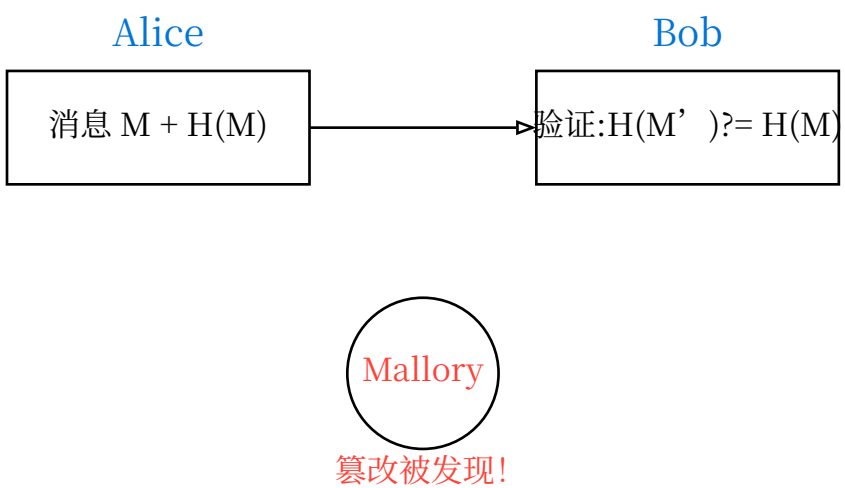
问题 2：完整性丢失 - Mallory 可以修改通信内容

杂凑函数解决方案

使用哈希函数 H 为消息生成“数字指纹”，接收方验证消息是否被篡改

杂凑函数解决方案

使用哈希函数 H 为消息生成“数字指纹”，接收方验证消息是否被篡改

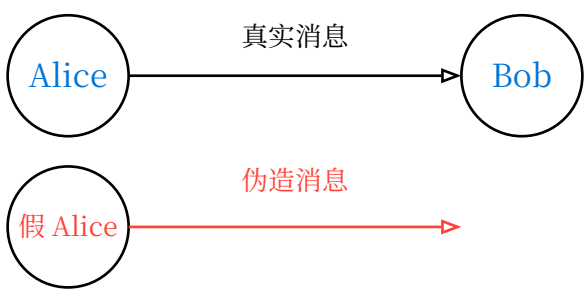


身份伪造威胁

Alice 如何确认消息真的来自 Bob？Bob 如何确认消息真的来自 Alice?

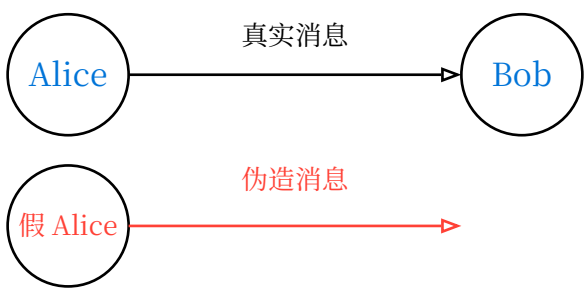
身份伪造威胁

Alice 如何确认消息真的来自 Bob？Bob 如何确认消息真的来自 Alice?



身份伪造威胁

Alice 如何确认消息真的来自 Bob？Bob 如何确认消息真的来自 Alice?



问题 3：身份认证缺失 - 无法验证消息发送方的真实身份

公钥密码解决方案

每个用户有一对密钥：公钥（公开）和私钥（保密）

公钥密码解决方案

每个用户有一对密钥：公钥（公开）和私钥（保密）

公钥密码解决方案

每个用户有一对密钥：公钥（公开）和私钥（保密）

应用：数字签名、密钥交换、身份认证

密码技术类别

技术类型	解决问题	典型算法	应用场景
对称密码 Symmetric Crypto	机密性保护 Confidentiality	AES, DES GIFT, PRESENT	数据加密 通信保密
哈希函数 Hash Functions	完整性验证 Integrity	SHA-256, SHA-3 PHOTON, SPONGENT	数字指纹 数据校验
公钥密码 Public Key Crypto	身份认证 Authentication	RSA, ECC 数字签名算法	数字签名 密钥交换
其他技术 Other Tech	随机性+协议 Random & Protocols	PRNG, TLS 密钥协商协议	安全协议 随机生成

IoT 环境下的密码学挑战

前面我们看到密码学为 Alice 和 Bob 提供了完整的安全解决方案，但在物联网环境中面临新的挑战：

IoT 环境下的密码学挑战

前面我们看到密码学为 Alice 和 Bob 提供了完整的安全解决方案，但在物联网环境中面临新的挑战：

过去十年，物联网稳步发展，被纳入：

- 智能电网、智能城市、智能家庭
- 农业、健康、智能交通、交通监控等场景

IoT 环境下的密码学挑战

2024 年：188 亿台互联设备正在使用中（同比增长 13%）

2025 年：约 270-309 亿台设备（IoT Analytics）

2030 年预测：约 400 亿台设备（GSMA Intelligence）

但是，IoT 设备具有严格的资源限制：

IoT 环境下的密码学挑战

2024 年：188 亿台互联设备正在使用中（同比增长 13%）

2025 年：约 270-309 亿台设备（IoT Analytics）

2030 年预测：约 400 亿台设备（GSMA Intelligence）

但是，IoT 设备具有严格的资源限制：

- 计算能力、RAM 大小、ROM 大小
- 寄存器宽度、不同的实现环境等

传统密码 vs 轻量级密码

传统 AES-128 需要：2400+ GE | IoT 设备预算：1000-2000 GE

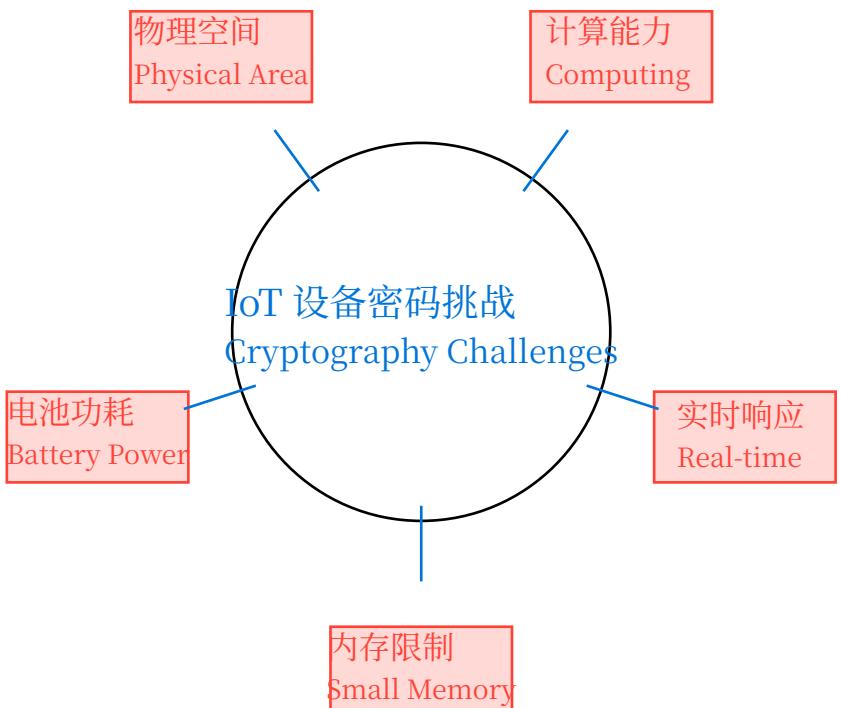
传统密码 vs 轻量级密码

传统 AES-128 需要：2400+ GE | IoT 设备预算：1000-2000 GE

传统密码算法	轻量级密码算法
· AES (高安全)	· GIFT, PRESENT (轻量分组密码)
· RSA (强认证)	· ECC (轻量公钥密码)
· SHA-256 (可靠哈希)	· PHOTON (轻量哈希)
资源需求过高	适配 IoT 约束

传统密码 vs 轻量级密码

解决方案：设计专门针对 IoT 设备资源约束的轻量级密码算法



轻量级密码的标准化工作 - 早期发展

1994 年 Neecham 等人提出描述简洁、实现简单的 Tiny Encryption Algorithm (TEA)

轻量级密码的标准化工作 - 早期发展

1994 年 Neecham 等人提出描述简洁、实现简单的 Tiny Encryption Algorithm (TEA)

2004 年 欧洲国家成立 ECRYPT/eSTREAM 项目： 80bits 密钥在受限硬件资源中的应用

轻量级密码的标准化工作 - 早期发展

1994 年 Neecham 等人提出描述简洁、实现简单的 Tiny Encryption Algorithm (TEA)

2004 年 欧洲国家成立 ECRYPT/eSTREAM 项目： 80bits 密钥在受限硬件资源中的应用

2012 年 IEC 发布 29192《轻量级密码》标准系列

轻量级密码的标准化工作 - 早期发展

1994 年 Neecham 等人提出描述简洁、实现简单的 Tiny Encryption Algorithm (TEA)

2004 年 欧洲国家成立 ECRYPT/eSTREAM 项目： 80bits 密钥在受限硬件资源中的应用

2012 年 IEC 发布 29192《轻量级密码》标准系列

2012 年 IEC 发布 29167 标准系列，至今仍在扩展

轻量级密码的标准化工作 - 早期发展

1994 年 Neecham 等人提出描述简洁、实现简单的 Tiny Encryption Algorithm (TEA)

2004 年 欧洲国家成立 ECRYPT/eSTREAM 项目： 80bits 密钥在受限硬件资源中的应用

2012 年 IEC 发布 29192《轻量级密码》标准系列

2012 年 IEC 发布 29167 标准系列，至今仍在扩展

2013 年 NIST 启动轻量级密码研究项目

轻量级密码的标准化工作 - 近期发展

2017 年 NIST 发布轻量级密码调查联合报告 NISTIR 8114

轻量级密码的标准化工作 - 近期发展

2017 年 NIST 发布轻量级密码调查联合报告 NISTIR 8114

2018 年 NIST 发布轻量级密码算法征集需求和评估标准通知

轻量级密码的标准化工作 - 近期发展

2017 年 NIST 发布轻量级密码调查联合报告 NISTIR 8114

2018 年 NIST 发布轻量级密码算法征集需求和评估标准通知

2019 年 4 月 NIST 公布了前两轮候选算法筛选结果

轻量级密码的标准化工作 - 近期发展

2017 年 NIST 发布轻量级密码调查联合报告 NISTIR 8114

2018 年 NIST 发布轻量级密码算法征集需求和评估标准通知

2019 年 4 月 NIST 公布了前两轮候选算法筛选结果

2021 年 3 月 NIST 宣布进入最终轮的 10 个轻量级密码算法

轻量级密码的标准化工作 - 近期发展

2017 年 NIST 发布轻量级密码调查联合报告 NISTIR 8114

2018 年 NIST 发布轻量级密码算法征集需求和评估标准通知

2019 年 4 月 NIST 公布了前两轮候选算法筛选结果

2021 年 3 月 NIST 宣布进入最终轮的 10 个轻量级密码算法

2025 年 8 月 NIST 正式发布轻量级密码算法标准， 选定 Ascon 算法族作为 NIST 轻量级密码算法标准。

设计规范与标准

根据这些轻量级分组密码算法的设计规范与标准， 轻量级分组密码在设计上应考虑以下：

1. 安全强度
2. 灵活性
3. 多重功能下的低开销
4. 密文扩展
5. 侧信道
6. 相关密钥攻击以及其他一些基本的攻击方法

密码研究热点

四类研究方法概述

当前轻量级密码学研究主要围绕以下四个热点方向展开：

1. 优化实现

对已有的轻量级密码算法进行硬件优化实现

2. 结构改进方法

基于分组密码，对密码算法的结构或部件进行改进

3. 应用场景设计

针对特定应用场景或需求进行设计的轻量级算法

4. 量子计算

面对量子计算威胁的后量子密码学研究

优化实现

硬件优化实现

针对 FPGA、ASIC 等硬件平台的优化实现研究

研究热点:

- 面积-吞吐率权衡优化
- 串行/并行架构设计
- S 盒硬件电路优化
- 低功耗设计技术

优化实现

软件优化实现

针对嵌入式处理器、微控制器的软件优化研究

研究热点:

- 位切片实现技术
- 指令级优化策略
- 寄存器分配优化
- 内存访问优化

优化实现

异构优化实现

针对 GPU、多核处理器的并行加速研究

研究热点:

- GPU 并行计算框架
- 内存层次结构利用
- 异构计算平台优化

优化实现 - CRAFT 硬件案例

以 CRAFT 算法为例的硬件优化实现

以《Efficient implementations of CRAFT cipher for Internet of Things》为例

优化实现 - CRAFT 硬件案例

以 CRAFT 算法为例的硬件优化实现

以《Efficient implementations of CRAFT cipher for Internet of Things》为例

针对 CRAFT 算法提出了 3 种新的硬件架构实现。

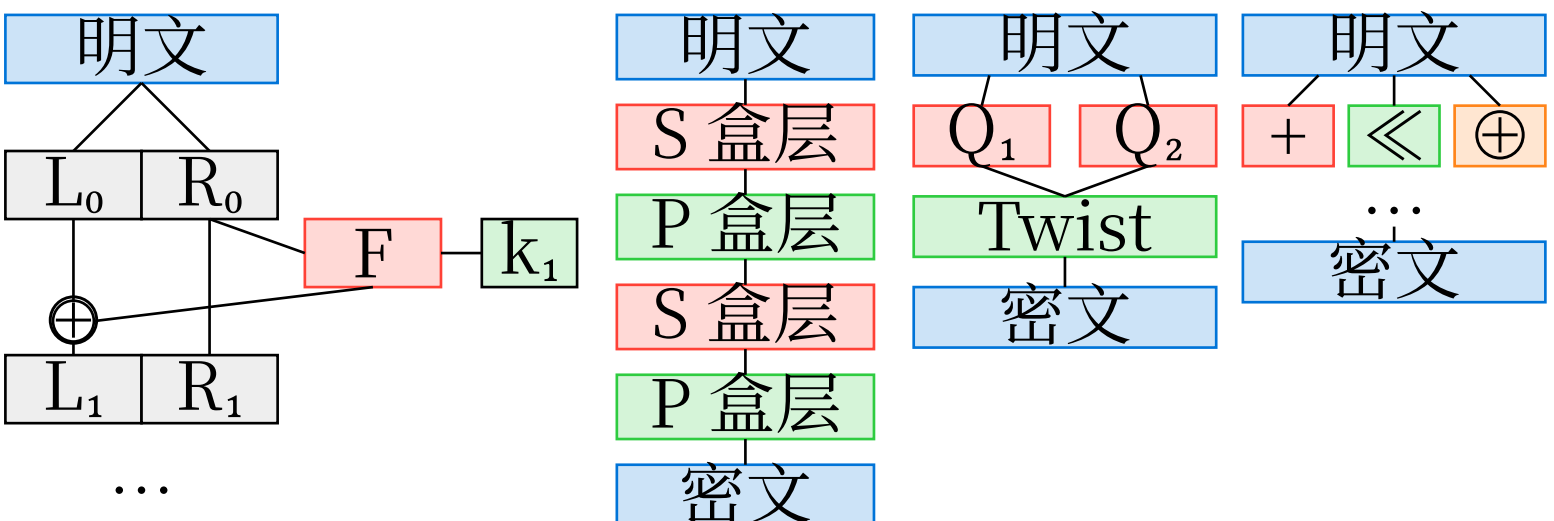
实验量少，关键在论文的书写。

只需进行不同的硬件优化架构实现，比较硬件参数。

结构改进方法

基于分组密码，对密码算法的结构或部件进行改进

Feistel 结构　SPN 结构　QTL 结构　ARX 结构



应用场景设计

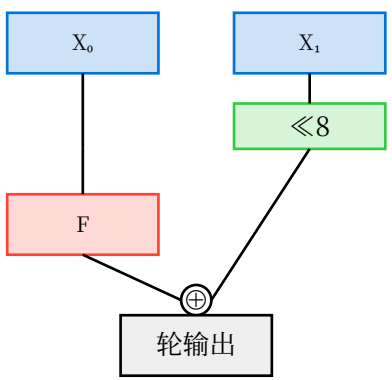
针对特定应用场景或者需求进行设计的轻量级分组密码算法

典型算法示例:

1. 面向软硬件灵活实现的 LBlock 算法
2. 专注低能耗指标设计的 Midori 算法
3. 基于低延迟理念设计的 PRINCE 算法
4. 面向 IC 打印的 PRINTcipher 算法

应用场景设计 - LBlock 算法

LBlock 算法加密流程图



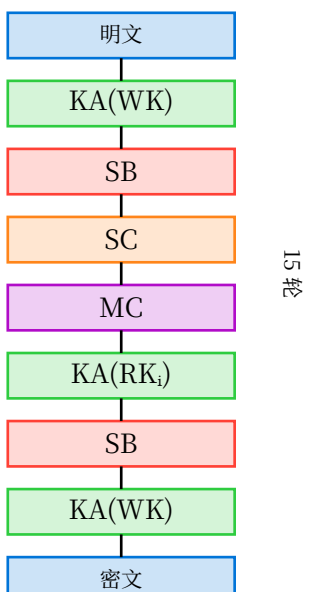
LBlock 采用了 4 位逐字排列，使得算法不仅可以在硬件中廉价实现，而且可以在软件环境中廉价实现。

- 硬件：需要 1320GE，吞吐量 200Kbps
- 软件：8 位微控制器，加密 64 位数据需要 3955 个时钟周期
- 算法每一轮只使用一半数据，另一半使用简单移位
- 密钥调度以流密码方式设计

应用场景设计 - Midori 算法 低能耗

1. 列混淆使用 4×4 几乎 MDS 二进制矩阵，在面积和信号延迟方面比 4×4 MDS 矩阵更有效。
2. 使用了一个轻量级、小延迟的 4 位 S-box。该 S 盒中的信号延迟分别是 PRINCE 和 PRESENT 的 1.5 倍和 2 倍。
3. Midori 算法的加密和解密功能相互转换时，只需要通过在电路中的小调整就可以达到加解密一致。

Midori 算法加密流程图



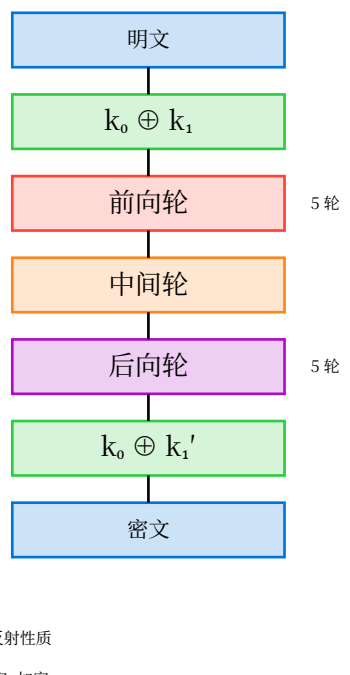
应用场景设计 - PRINCE 算法 低延迟

PRINCE 算法低延迟主要的方式是轮数尽可能地少，轮函数中的部件尽量采用低延迟。

因此，PRINCE 的轮数只有 11 轮（5+1+5），还采用了一个几乎 MDS 矩阵，这样有助于为各种类型攻击提供更好边界，进而可以允许减少轮数，从而减少延迟。

除此之外，PRINCE 的加密结构也很新颖，具有 α-反射性质：

PRINCE 算法加密流程图



应用场景设计 - PRINTcipher 算法

由于 IC 打印中使用到的电子产品代码（EPC）的长度为 96 位，因此 PRINTcipher 使用的明文长度为 48bit 和 96bit 两个版本，密钥长度为 160bit，两个版本分别对应的轮数为 48，96 轮。

常规的 IC 为了节省开销，一般要求 IC 中的使用密钥不进行更改，因此，PRINTcipher 算法的没有密钥扩展部分，设计者通过使用一种排列方法，使得算法可以根据不同的密钥具有不同的加密流程。

密码组件 - 侧重低延迟或侧重轻量的 S 盒构造方法

目前，为了能快速优化 4×4 的 S 盒，研究人员主要采用自动化的方法搜索 S 盒， 具体可以细分为两个方向。

方向一：首先获得具有良好密码特性的 S 盒，然后通过某种方法优化 S 盒的硬件逻辑电路

Jean 等人应用 LIGHTER 搜索给定 4×4 S 盒的面积优化实现。但 LIGHTER 的一个缺失考虑因素是实现延迟的度量。Stoffelen 将寻找最佳位片实现的整个问题建模为 SAT 求解器可以解决的问题。

密码组件 - 侧重低延迟或侧重轻量的 S 盒构造方法

方向二：首先从硬件逻辑层创建紧凑的 S 盒，然后检查其密码特性

Watanabe 等人使用对基本可逆函数的迭代来生成初始 S 盒集，然后将约束添加到初始 S 盒集中以获得目标 S 盒。

Guo 等人提出了一种在 ASIC 中寻找电路深度优化的实现的方法。本质上，该工具首先给定 S 盒和每个单元操作的成本，作为初始 S 盒集。然后，输出查询结果和四个坐标的深度中的最大值。

密码组件 - 侧重低延迟和比特切片的线性矩阵构造方法

为了快速找到性能优良的矩阵，研究人员主要采用启发式算法搜索矩阵， 具体可以细分为三个方向。

方向一： 首先搜索一个性能好的矩阵，然后再搜索该矩阵的良好硬件实现

方向二： 首先从逻辑层搜索可行的矩阵构造，然后检查其矩阵分支数特性

方向三： 首先确定轻量的小规格矩阵，然后通过递归、子域构造的方法构造性能良好的大规格矩阵

量子计算 - 密码技术

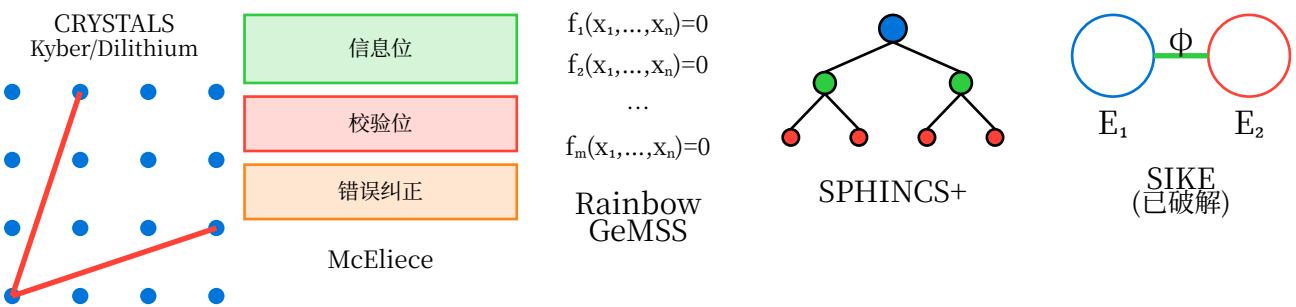
面对量子计算威胁的后量子密码学研究

随着量子计算技术的快速发展，传统的 RSA、ECC 等公钥密码算法面临被量子计算机破解的威胁。 2024 年 Google 的 Willow 芯片和 2025 年 Microsoft 的 Majorana 1 等量子计算突破， 使得后量子密码学成为当前最热门的研究方向。

量子计算 - 密码技术

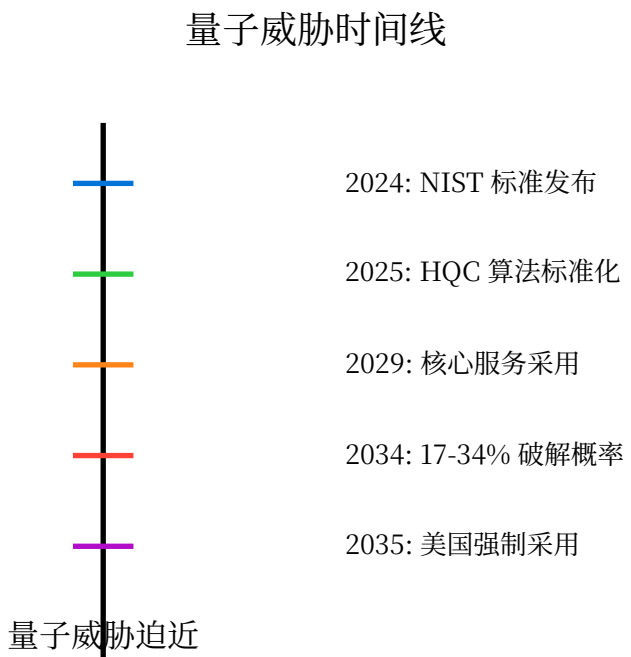
后量子密码技术路线分类

格密码 编码理论 多变量 哈希函数 同源理论



量子计算 - NIST 标准化算法（2024-2025）

- ML-KEM（基于 CRYSTALS-Kyber）
  - 密钥封装机制
- ML-DSA（基于 CRYSTALS-Dilithium）
  - 数字签名算法
- SLH-DSA（基于 SPHINCS+）
  - 无状态哈希签名



量子计算 - 产业应用现状与挑战

技术突破:

- Google Willow 芯片：减少量子噪声和错误
- 理论预测：100 万量子比特可在 1 周内破解 RSA-2048
- 中国“本源悟空”装备 PQC“抗量子攻击护盾”

应用挑战:

- 算法性能开销：后量子算法密钥长度和计算复杂度显著增加
- 迁移复杂性：需要重新设计整个密码基础设施
- 标准不统一：不同应用场景需要不同的后量子算法
- 人才短缺：后量子密码学专业人才严重不足

论文书写及投稿

研究热点锁定 - 科学问题寻找

如何寻找科学问题：

(1) 在亲身研究和实践中，遇到的问题（主要方法）。

(2) 通过读文献，察觉到前人未解决的问题或未意识到的问题。

(3) 借鉴“它山之石”，有了攻克悬而未决老问题的奇思妙想。

研究热点锁定 - 读论文

如何在入门阶段快速锁定研究热点：

步骤 1 看综述论文。

- 近一年的博士、硕士大论文综述部分。
- 近一年的高水平期刊收录的综述论文。

步骤 2 看相关领域顶会顶刊收录的论文（近两年）。

论文阅读概述

好的创意来自于好的论文阅读。

怎样寻找好论文以及阅读论文：

下载相关论文集　三步阅读法　文献管理

论文阅读 - 论文集下载

主要下载近 3 年的文献，可以通过以下几种方式获得论文：

(1) 综述论文中的参考文献（入门时使用）。

(2) 熊猫(谷歌)学术或三大数据库搜索关键字。

下载论文可以使用：

- (1) 熊猫学术（https://panda321.com/）
- (2) SPIS 学术下载（http://spis.hnlat.com/）
- (3) 衡阳师范学院文献互助群（572761699）

论文阅读 - 三步阅读法 - 粗读

第一步（粗读）

(1) 仔细阅读标题、摘要与引言。

(2) 细读每个章节与子章节的标题（忽略其他的内容）。

(3) 总结。

(4) 浏览参考文献，标记自己已经读过的。

论文阅读 - 三步阅读法 - 粗读

第一步完成后对论文将会有如下认知，或者提出一些问题：

(1) 文章类别：这篇文章是什么种类的？轻量级分组密码？分组密码？优化？密码分析？侧信道分析？

(2) 文章内容：有哪些其他相关文章？问题分析基于哪个理论？

(3) 正确性：文中的假设或提出的理论是否合理正确？

(4) 贡献：文章的主要贡献是什么？清晰性：文章的写作是否足够好？

论文阅读 - 三步阅读法 - 细读

第二步（细读）

(1) 细心阅读图、表

(2) 记得标记相关的未读的参考文献（这是一种非常好的了解论文背景的方法）

在第二步中，更细心地读文章，特别是背景，可忽略一些细节如理论的证明（可以作一些注释）。

论文阅读 - 三步阅读法 - 细读

完成第二步，论文中仍然会有许多你不理解的地方，比如一些细分领域的背景理解不够或是证明的理论或实验不能理解。这时可以:

(a) 把这篇论文先放边。

(b) 稍后再拿起来认真阅读，特别注意背景部分。

(c) 直接进入第三步。

论文阅读 - 三步阅读法 - 实验复现

第三步（实验复现）

为了完全地理解这篇文章，并能够在此基础上进行改进，需要做的最关键的步骤是对论文实验的重构，即站在作者的角度，重复他的工作。

文献管理

每读一篇好论文，需要对其进行管理，这有两个好处，一是可以方便再调出来看，二是引用的时候非常方便。主要的管理方法：

(1) 建立自顶向下建立文件夹。大方向→小方向→好论文。论文的文件夹命名（年份+名字+期刊/会议）

(2) 画思维导图

(3) 文件管理软件（Mendeley Desktop，JabRef）

写论文概述

通过前面的论文阅读与文献收集管理，此时应该内心会产生一些想法和思路，先在理论上对自己提出的想法进行推论验证。然后开始设计实验，由于前期做了复现论文的准备，因此在实验上会相对比较顺利。

论文结构

在完成最初的想法构思或创新的方法后，在理论上对方法进行抽象，实验验证方法的正确性与优势，开始撰写论文。一般，实证型论文的结构包括：

(1) 题目/摘要
(2) 引言
(3) 方法
(4) 实验结果和讨论
(5) 结论
(6) 致谢
(7) 参考文献

摘要 五个语步

摘要部分通常涉及（或者说就是）五个语步：

语步 1：概述研究现状或主要问题。

语步 2：介绍本研究的内容和目的。

语步 3：介绍研究方法。

语步 4：指出主要研究发现和结果。

语步 5：总述结论、研究价值、应用前景或建议。

引言 三个语步

通常引言部分有三个语步：

语步 1：介绍研究领域。突出研究话题的意义、价值和重要性，回顾相关文献。

语步 2：确定研究动机。指出先前研究的不足或扩展现有知识。

语步 3：描述当前研究。提出研究目的、研究性质或研究问题，或宣布重要的研究结果，明确研究的意义，概括全文的框架结构。

方法部分语步

方法部分一般如下：

语步 1：介绍研究目的、研究问题或假设。

语步 2：介绍研究步骤。

语步 3：介绍数据分析方法。

结果讨论语步

实验结果和讨论部分一般如下：

语步 1：提供背景信息。实验环境，实验参数设置。

语步 2：以文本形式呈现研究结果，并进行讨论（一般使用对比法）

语步 3：以非文本形式（如图、表等）呈现研究结果。

结论部分语步

结论部分通常是文章正文最后一节，说明研究的意义或应用前景。

语步 1：概括当前的研究。回顾研究动机、目的、主要的研究结果。

语步 2：评价研究结果的价值。

语步 3：讨论研究的不足之处。指出研究存在的问题并给予解释。（这一步通常不写）

语步 4：对研究做出推论，指出未来的研究启示及方向。

写作工具

写论文过程中一些非常有用的工具:

文字处理：word

文字排版：latex: TeX live + TeX studio、CTeX、VsCode 等

实验绘图：matlab、python、Origin 等

结构绘图：Visio、smartdraw、Drawio 等

投稿过程

论文初稿完成后，进行修改与优化，下一个步骤便是进行投稿，这一步骤涉及：

(1) 期刊选择。借助 LetPub 网站查询期刊相关信息。（中科院分区，影响因子，年文章数，投稿周期，网友经验等） https://www.letpub.com.cn/ 期刊官网。了解期刊投稿范围和要求。

(2) 提交相关文档。

审稿返修

如何回复审稿人意见

1. 所有问题必须逐条回答。如果存在一段审稿意见多个问题，进行拆分回复。回复格式一般为：感谢+回复+附上手稿中的修改部分。
2. 尽量满足意见中需要补充的实验。
3. 满足不了的也不要回避，说明不能做的合理理由。
4. 审稿人推荐的文献一定要引用，并讨论透彻。
5. 回复审稿人关于稿件的修改一定要全部使用现在完成时，不要使用一般过去时。

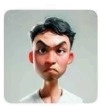
常见审稿人意见

1. 文章格式错误、拼写错误、图表标错、语句有歧义等等。
2. 要求加文献。仔细阅读审稿人推荐的参考文献，在意见回复中要讨论透彻。
3. 要求补充或修改手稿的内容。比如引言增加相关文献讨论，摘要或总结修改。要求语法修改。要求调整说法。
4. 质疑内容的创新性。感谢审稿人的评论+阐述论文的创新点和优势，适当引用顶会顶刊的论文做依据。建议补充实验。

审稿意见回复

- 做这个实验。同意审稿人的观点+新的实验数
- 没有实验条件，或者不能在短时期内做这个实验。看看现有数据能不能同样说明科学问题，也可以补充其他相关实验数据辅助说明，实在不行，可以查阅文献（最好是顶会或顶刊），告诉审稿人其他文献也存在同样的情况。
- 觉得没有必要补实验。这个时候需要给出自己的理由。最好罗列一些证据，或者引用其它文献（最好是顶会或顶刊）来支撑自己的观点。

Q&A



isomo

Shaoyang, Hunan

Scan the QR code to add me as a friend.

Bilibili isomo