再论 Hash-ECB-Hash 结构在线密码的构造

刘 刚[1,2], 王 鹏[1,2], 魏 荣[3], 叶顶锋[1,2]

[1]中国科学院信息工程研究所信息安全国家重点实验室 北京 中国 100093
[2]中国科学院大学网络空间安全学院 北京 中国 100049
[3]北京卫星信息工程研究中心 北京 中国 100086

摘要 在线密码是众多密码方案如认证加密方案等中使用的重要组件。考虑到运算性能和安全性, Hash-ECB-Hash 结构为构造并行计算的且在选择密文攻击下安全的在线密码提供了潜在的可能性。本文我们从分析在线密码 POE 开始, POE 是到目前为止已有文献中唯一使用 Hash-ECB-Hash 结构的在线密码, 然而, POE 中哈希层使用的哈希函数的 AXU 抗碰撞性质不能像它声称的那样保证其安全性。Nandi 给出了一种有效的区分攻击, 仅需一次加密询问。为了防止对 POE 的攻击, 其哈希层的分量函数在同一和不同加密询问的输出之间碰撞概率都应该是可忽略的。然后我们针对哈希层提出了在线泛哈希函数(OUHF)的概念来满足这种条件, 包括 OAU 函数和 OAXU 函数, 并且证明如果哈希层使用 OAU 函数且底层分组密码是在选择密文攻击下安全的, 则 Hash-ECB-Hash 结构在选择密文攻击下也是安全的。我们给出了几种 OAU 函数的构造, 包括 CFB 和 CBC 模式, 还给出了两种新的构造, 其一是基于有限域上乘法函数的构造 MCFB, 另一种是使用输入输出异或链接方式的构造 XCH。之后, 基于 CCA 安全的在线密码 OC, 通过添加 Nonce、关联数据、认证码的生成等处理过程到在线密码中, 我们构造了一个简单的在线认证加密方案 OAE[OC]。然后我们对在线认证加密方案的安全性重新定义, 并使用归约证明技术论证了其安全性, 包括机密性和完整性。最后, 我们总结了从在线密码到在线认证加密方案的一些设计理念。

关键词 在线密码; POE; Hash-ECB-Hash 结构; 在线泛哈希函数; 在线认证加密方案
中图法分类号 TP309.2 **DOI 号** 10.19363/J.cnki.cn10-1380/tn.2026.01.01

Revisiting Construction of Online Cipher in Hash-ECB-Hash Structure

LIU Gang[1,2], WANG Peng[1,2], WEI Rong[3], YE Dingfeng[1,2]

[1] State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China
[2] School of Cyber Security, University of Chinese Academy of Sciences, Beijing 100049, China
[3] Beijing Satellite Information Engineer Institute, Beijing 100086, China

Abstract Online cipher is an important primitive in many cryptographic schemes, such as authenticated encryption schemes. Considering performance and security, the Hash-ECB-Hash structure provides a potential way to construct parallelizable and CCA secure online cipher. In this paper, we start from the analysis of online cipher POE, which is the only instantiation of Hash-ECB-Hash structure in the literature. However, the AXU property of hash function in the hash layer cannot guarantee the security of POE as it claimed. Nandi gave an efficient distinguishing attack which needs just one encryption query. In order to thwart the attack to POE, the output-collision probability of the component function of the hash layer should be negligible in both same and different encryption queries. Then we propose a new concept of online universal hash function (OUHF) including online almost universal (OAU) and online almost XOR universal (OAXU) hash functions for the hash layer to meet the condition and prove that the Hash-ECB-Hash structure is CCA secure if the hash layer is online almost universal (OAU) and the underlying block cipher is CCA secure. We give several concrete constructions of OAU hash functions, including the CFB and CBC modes. We also give two new constructions, one named MCFB based on finite field multiplication function, and another construction named XCH by chaining the operation XOR of input and output. After that, using the online cipher OC with CCA secure, we give a new and simple construction of online authenticated encryption schemes OAE[OC] by adding the processes of dealing with nonce, the associated data and tag generating to the online cipher. Then we revisit the security notions of online authenticated

通讯作者: 王鹏, 博士, 副研究员, Email: wpeng@iie.ac.cn。
本课题得到国家自然科学基金(No. 61732021, No. 61472415)和国家重点研发计划(No. 2018YFA0704704, No. 2018YFB0803801)资助。
本文是会议论文[1]的进一步扩展完善, 包含会议论文的内容: 对 POE 的分析; 对 OAU 函数和 OAXU 函数的定义和对 OAU 函数的实例化; 以及对 HEH 结构在线密码的安全性证明。本文提出了一种新的 OAU 函数的实例, 即 XCH; 完善了 HEH 结构在线密码的安全性证明; 并从 OPRP-CCA 安全的在线密码 OC 构造了在线认证加密方案 OAE, 给出了 OAE 的安全性定义, 并证明了其安全性。
收稿日期: 2020-12-02; 修改日期: 2021-01-25; 定稿日期: 2023-02-22

encryption and prove our scheme is secure for its privacy and integrity using the technique of reduction proof. Finally, we conclude some ideas in the design from online cipher to online authenticated encryption schemes.

Key words online cipher; POE; Hash-ECB-Hash structure; online universal hash function; online authenticated encryption

1 引言

当今世界网络无处不在, 即时数据传输无时无刻不在进行。安全性应用场景激发了密码算法更多的在线运行需求。一方面, 实时信息服务如直播、交互视频、股票交易系统等需要密码算法的高处理性能和低延迟; 另一方面, 很多存储等资源有限制的嵌入式、物联网等设备一次只能处理和存储一小段数据。因此, 在未来 5G 等应用环境中, 在线数据处理的安全性和高效性变得越来越重要。

1.1 在线密码

在线密码(online cipher)[2] 是由 Bellare 等于 2001 年首次提出的概念, 通常作为密码组件以在线的方式来处理数据。在线密码的构造是在线密码方案设计的关键, 大量在线认证加密(authenticated encryption, AE)方案如 McOE[3]、POET[4]、COPA[5]、ELmD[6]、COLM[7]等都基于在线密码, 它们的在线安全强度保证了超越普通机密性和完整性的在线安全和 Nonce 误用容忍等性质。

在线密码可以使用分组密码(block cipher, BC)、可调分组密码(tweakable block cipher, TBC)或置换(permutation)再结合泛哈希函数(universal hash functions, UHF)的方式来构造, 构造方法可分为串行的(sequential)和非串行的(non-sequential)两类。

早期的设计如 HCBC1/2[2]、HPCBC[8]、MHCBC[9]、MCBC[9]、TC1/2/3[10]等都属于串行的, 以一个分组接着一个的方式来处理数据, 若前一分组没处理完成, 不能进行当前分组的处理。近几年提出的超越生日界(beyond birthday bound, BBB)安全的在线密码如 POEx[11]、XTC[12]等也是串行的。

非串行在线密码的设计通常使用多层结构, 包括 4 轮 Feistel 结构如 OleF[13]等、EME(Encrypt-Mix-Encrypt)结构如 COPA[5]、ELmD[6]、COLM[7]等中使用的在线密码、HEH(Hash-ECB-Hash)结构如 POE[4]等。多层结构提供了潜在的并行计算的可能性。第一种结构使用 4 轮 Feistel 结构处理每个分组来输出密文分组, 可以达到选择密文攻击(chosen ciphertext attack, CCA)下的安全性[14], 但是其分组长度是轮函数分组长度的两倍。EME 结构首先使用分组密码加密明文分组, 再用线性变换混淆这些分组, 最后再次使用分组密码来加密分组得到密文分组。EME 结构在两层分组密码加密层可以并行计算, 但结构只达到选择明文攻击下的安全性。HEH 结构首先使用泛哈希函数处理分组数据, 再使用 ECB 模式加密这些分组, 最后再次使用泛哈希函数来处理生成密文分组。泛哈希函数[15]是密码方案中常用的重要组件, 由于其是满足某种组合性质的简单函数, 在效率上有极大的优势[16-19]。表 1 从分类和安全性两个方面列出了现有的在线密码。

表 1 现有在线密码的分类和安全性

Table 1 The category and security of Online Cipher

	串行	非串行
CPA 安全	HCBC1[2]、TC1[10]	COPA[5]中的 COPE、ELmD[6]和 COLM[7]中的在线密码
CCA 安全	HCBC2[2]、MCBC[9]、MHCBC[9]、TC2/3[10]、XTC[12]	OleF[13]、POE[4]

1.2 POE 的缺陷

POE 是目前唯一的基于 HEH 结构的在线密码, 是认证加密方案 POET[4]中的核心模块。不幸的是, Nandi[20]指出, 当哈希层使用某些特殊的泛哈希函数时, POE 存在一种区分攻击, 只需要做一次加密询问即可攻击成功。因此 POE 的安全假设是不合理的, 其证明存在缺陷。

1.3 动机

为了软硬件实现上的运算效率, 我们应尽量设计能够并行或者部分并行的在线密码方案。由于敌手可能对轻量级设备进行访问, 在一段时间内获得加密或者解密的权限, 在线密码方案应该在选择密文攻击下是安全的。因此, 一个理想的在线密码应该既是非串行的, 也是 CCA 安全的。HEH 结构提供了在线密码同时满足这两方面条件的可能性。作为这种结构目前唯一的实例, POE 是一个很好的研究对象, 我们首先需要研究清楚 POE 的设计到底存在什么缺陷。POE 的哈希层使用一个带有 AXU(almost XOR universal)性质的泛哈希函数 f 在 CFB 模式运行, 记作 $CFB[f]$, 但是根据 Nandi[20]的研究, f 函数的 AXU 性质不足以保证 HEH 结构的安全性。一个问题是什么样的 f 可以保证 HEH 结构的安全性? 除了 CFB 模式, 还有很多方式可以构造哈希层, 一个

更一般的问题是哈希层的什么性质可以保证 HEH 结构的安全性。

如果存在 HEH 结构的 CCA 安全的在线密码, 更进一步的问题是如何使用该在线密码来构造安全的认证加密方案。

1.4 我们的工作

(1) 分析了 POE 的结构, 其哈希层是使用泛哈希函数的 CFB 模式, 之前的研究表明泛哈希函数的 AXU 性质不能保证 POE 的安全性。为了防止这种攻击, 哈希层的分量函数的输出碰撞概率应该是可忽略的。

(2) 将经典的泛哈希函数概念扩展到在线的情况, 提出了在线泛哈希函数(online universal hash function, OUHF)的定义, 包括 OAU(online almost universal)函数和 OAXU(online almost XOR universal)函数。POE 的哈希层中使用均匀随机自反函数(uniform random involution function, URIF)或有限域上乘法函数的 CFB 模式不是 OAU 函数。

(3) 给出了基于均匀随机函数(uniform random function, URF)的在线泛哈希函数的构造, 包括 CFB 和 CBC 模式, 还给出了基于有限域上乘法函数的构造 MCFB 和使用输入输出异或链接方式的构造 XCH。

(4) 证明了当哈希层是 OAU 函数、底层分组密码是选择密文攻击下安全的伪随机置换(pseudorandom permutation, PRP)时, HEH 结构是 CCA 安全的。

(5) 有了基于 HEH 结构构造的在线密码 OC 作为底层组件, 我们给出了一种简单的在线认证加密方案 OAE 的构造, 然后定义了一种新的在线认证加密方案的安全性, 并给出在新的定义下我们设计的在线认证加密方案的安全性证明。

2 基础知识

2.1 符号

对于 $b\in\{0,1\}$, 令b^n是 n 比特 b 的字符串。令$\{0,1\}^n$是所有 n 比特字符串的集合, $\{0,1\}^{n*}$是所有比特长度为 n 的倍数的字符串的集合, 不包括空字符串ϵ。字符串 M 的比特长度写作$|M|$, 显然$|\epsilon|=0$。如果字符串 $X\in\{0,1\}^{n*}$可以分成 n 比特分组, 则将其分组长度写作$|X|_n$。如果$|X|_n=m$, 它的第 i 个分组表示为$X[i]$, 则$X=X[1]\,||\,X[2]\,||\cdots||\,X[m]$, 或者写作 $X=(X[1],X[2],...,X[m])$。如果 $i>|X|_n$, $X[i]=\epsilon$。X 的连续的从第 i 到第 j 个$(1\le i<j\le m)$分组表示为 $X[i..j]=(X[i],X[i+1],\cdots,X[j])$, 如果 $j<i$, 定义$X[i..j]=\epsilon$。

$LCP_n(X,Y)$表示 $X,Y\in\{0,1\}^{n*}$的最长公共前缀, 即最长的字符串 $Z\in\{0,1\}^{n*}$使得 $Z=X[1..i]=Y[1..i]$且 $X[i+1]\neq Y[i+1],i\le|X|_n,i\le|Y|_n$。最长公共前缀的长度记作 $LLCP_n(X,Y)$, 是最长公共前缀的分组长度, 即 $LLCP_n(X,Y)=|Z|_n$, 特别地, $LCP_n(X,X)=X$, $LLCP_n(X,X)=|X|_n$。

一般在不会引起混淆的情况下, 乘法运算符·可以省略, 例如 $KX=K\cdot X$。K^i表示 i 个 K 的乘积。$\mathcal{S}_1\times\mathcal{S}_2$表示两个集合$\mathcal{S}_1$和$\mathcal{S}_2$的笛卡尔积。$s\xleftarrow{\$}\mathcal{S}$ 表示从集合 $\mathcal{S}$ 中均匀随机选取一个元素 s。

令$A^{\mathcal{O}}\Rightarrow 1$ 表示敌手 A 询问一次或多次谕言机 $\mathcal{O}$, 然后输出一比特 1。不失一般性地, 假设敌手从不做那些知道答案的询问, 例如, 敌手不会对确定性的谕言机重复做相同的询问, 不会用可以从加密询问推断出来的答案去询问解密询问, 反之亦然。

2.2 选择明文攻击与选择密文攻击

敌手具有询问加密算法的能力, 可以选择一定数量的明文, 并获得相应的密文, 然后根据得到的信息进行攻击的过程, 称为选择明文攻击(chosen plaintext attack, CPA)。敌手同时具有询问加密算法和解密算法的能力, 可以选择一些明文, 获得相应的密文, 也可以选择一些密文, 获得相应的明文, 然后根据得到的信息进行攻击的过程, 称作选择密文攻击(CCA)。显然, 能够抵抗选择密文攻击的方案有更高的安全性。

2.3 不可区分性

在密码方案的安全性证明中, 我们通常用其与随机函数是不可区分的来阐述。

定义 1. 可忽略函数. N 和 R 分别为自然数集和实数集, 若函数$\varepsilon: N\to R$ 满足$\forall r\in R,\exists n_0\in N,\forall n\ge n_0$都有$\varepsilon(n)\le n^{-r}$, 则称函数$\varepsilon$是可忽略函数。

若所有的多项式时间敌手都不能区分两个概率分布 X 和 Y, 也即存在可忽略函数ε, 对于任意的多项式时间敌手 A 都有 $\Pr[A(X)\Rightarrow 1]-\Pr[A(Y)\Rightarrow 1]\le\varepsilon$, 则称这两个概率分布是计算上不可区分的。

2.4 在线函数和在线置换

G:$\{0,1\}^{n*}\to\{0,1\}^{n*}$是一个在线函数, 如果 G 是保长的, 且当前的输出只与当前输入与之前的输入有关, 即 m 分组输入 $X=(X[1],X[2],\cdots,X[m])$映射到 m 分组的输出 $Y=(Y[1],Y[2],\cdots,Y[m])$, 其中$|X[i]|=|Y[i]|=n,i=1,2,\cdots,m$, 并且每个输出分组 $Y[i]$只依赖于 $X[1..i]$。在线函数 G 的每个输出可以用分量函数G^c:$\{0,1\}^{n*}\to\{0,1\}^n$表示, G^c只输出最后一个分组的数据, 即$G^c(X[1..i])=Y[i]$, 则 $G(X)=(G^c(X[1]),\ G^c(X[1..2]),\cdots,G^c(X[1..m]))$, $i=1,2,\cdots$。例如由分量函数 $Xor^c(X[1..i])=$

$\oplus_{j=1}^{i} X[i], i=1,2,\cdots$定义的异或函数 Xor 是一个在线函数。任意实用的在线函数应该可以用一种在线的方式高效计算, 如 $Xor(X[1..m])=Y[1..m]:S=S\oplus X[i], Y[i]=S, i=1,2,\cdots,m$, 其中 $S\in\{0,1\}^n$是在线计算过程中维持的状态, 初始化为0^n。

因此我们可以用多分组输入多分组输出函数、分量函数或在线计算过程 3 种方式描述在线函数。

如果 G 是可逆的在线函数, 即对任意的 $X[1..i-1]\in\{0,1\}^{n(i-1)}, G^c(X[1..i-1],\cdot), i=1,2,\cdots$ 都是在$\{0,1\}^n$上的置换, 我们称 G 是一个在线置换, 其中定义$G^c(X[1..0],\cdot)=G^c(\cdot)$。$G$ 的逆表示为G^{-1}也是一个在线置换, 如果 $Y[1..i-1]=G(X[1..i-1]), i=2,3,\cdots$, $(G^{-1})^c(Y[1..i-1],\cdot)$是$G^c(Y[1..i-1],\cdot)$的逆, 比如异或函数 Xor 是一个在线置换。

如果 G 是一个在线置换, $X,X'\in\{0,1\}^{n*}$ 且 $LLCP_n(X,X')=l$, 则 $LLCP_n(G(X),G(X'))=LLCP_n(G^{-1}(X),G^{-1}(X'))=l$, 我们将这种性质称作最长公共前缀保持性(LCPP)。

2.5 分组密码和在线密码

$\mathcal{K}$ 是密钥空间, $E:\mathcal{K}\times\{0,1\}^n\to\{0,1\}^n$是分组密码, 如果对于任意 $K\in\mathcal{K}$, $E(K,\cdot)$是置换。一般我们将密钥写作下标的形式, 即$E_K(\cdot)=E(K,\cdot)$, 其逆表示为D_K。$OC:\mathcal{K}\times\{0,1\}^{n*}\to\{0,1\}^{n*}$是在线密码, 如果对于任意 $K\in\mathcal{K}$, $OC_K(\cdot)=OC(K,\cdot)$是在线置换。另外我们定义 $F:\mathcal{K}\times\{0,1\}^m\to\{0,1\}^n$是 m 比特输入到 n 比特输出的带密钥的函数。

当使用分组密码、在线密码或一般的带密钥的函数时, 其密钥都是从密钥空间随机选取的, 即 $K\xleftarrow{\$}\mathcal{K}$, 使得我们选择的分别是随机置换、在线随机置换或随机函数。令 $Perm(n)$是$\{0,1\}^n$上所有置换的集合, 一个均匀随机置换(uniform random permutation, URP), 是$\pi\xleftarrow{\$}Perm(n)$。令 $OPerm(n)$是$\{0,1\}^{n*}$上所有在线置换的集合, 一个均匀随机在线置换(online uniform random permutation, OURP)是$\rho\xleftarrow{\$}OPerm(n)$。令 $Func(m,n)$是$\{0,1\}^m$到$\{0,1\}^n$上所有函数的集合, 若 $m=n$ 记作 $Func(n)$, 一个均匀随机函数是 $f\xleftarrow{\$}Func(m,n)$。

分组密码、在线密码或带密钥的函数的安全性一般定义为其与相应均匀随机对象的不可区分性。

定义 2. PRP-CPA、PRP-CCA、OPRP-CPA、OPRP-CCA、PRF。$E:\mathcal{K}\times\{0,1\}^n\to\{0,1\}^n$是一个分组密码, $OC:\mathcal{K}\times\{0,1\}^{n*}\to\{0,1\}^{n*}$是一个在线密码, $F:\mathcal{K}\times\{0,1\}^m\to\{0,1\}^n$是一个带密钥的函数。敌手 A 的区分优势定义如下:

$$Adv_E^{prp-cpa}(A)=\Pr[A^{E_K}\Rightarrow 1]-\Pr[A^{\pi}\Rightarrow 1],$$

$$Adv_E^{prp-cca}(A)=\Pr[A^{E_K,D_K}\Rightarrow 1]-\Pr\left[A^{\pi,\pi^{-1}}\Rightarrow 1\right],$$

$$Adv_{OC}^{oprp-cpa}(A)=\Pr[A^{OC_K}\Rightarrow 1]-\Pr[A^{\rho}\Rightarrow 1],$$

$$Adv_{OC}^{oprp-cca}(A)=\Pr\left[A^{OC_K,OC_K^{-1}}\Rightarrow 1\right]-\Pr\left[A^{\rho,\rho^{-1}}\Rightarrow 1\right],$$

$$Adv_F^{prf}(A)=\Pr[A^{F_K}\Rightarrow 1]-\Pr\left[A^{f}\Rightarrow 1\right]。$$

这里 PRP-CPA、PRP-CCA、OPRP-CPA、OPRP-CCA 和 PRF 分别表示抗选择明文攻击的伪随机置换(pseudorandom permutation against chosen plaintext attack)、抗选择密文攻击的伪随机置换(pseudorandom permutation against chosen ciphertext attack)、抗选择明文攻击的在线伪随机置换(online pseudorandom permutation against chosen plaintext attack)、抗选择密文攻击的在线伪随机置换(online pseudorandom permutation against chosen ciphertext attack)和伪随机函数(pseudorandom function)。后面, 我们将拥有至多资源 R 的敌手 A 的最大优势写作 $Adv_{\Pi}^{xxx}(R)=max_A\{Adv_{\Pi}^{xxx}(A)\}$, 资源包括总的询问次数 q、总询问分组数σ和运行时间 t 等。当 $Adv_{\Pi}^{xxx}(R)$可忽略时, 我们认为Π是 xxx。

2.6 两个转换引理

均匀随机置换(URP), 即理想的分组密码, 与均匀随机函数(URF)是不可区分的, 即如下的 PRP/PRF 转换引理[21]。

引理 1. PRP/PRF 转换引理[21]。敌手 A 做至多 q 次询问, 则有

$$\Pr\left[A^{\pi,\pi^{-1}}\Rightarrow 1\right]-\Pr\left[A^{f,f'}\Rightarrow 1\right]\le q(q-1)/2^{n+1},$$

其中, $\pi\xleftarrow{\$}Perm(n)$, $f,f'\xleftarrow{\$}Func(n)$。

类似地, 还有 OPRP/OPRF 转换引理[2]: 一个在线均匀随机置换(OURP)几乎用随机分组值来回答询问, 除了那些满足 LCPP 性质的分组。换言之, 在做加密和解密询问时, 维护一个明密文对的记录, 对于任何对加密/解密的询问 Z, 找到 Z 的最长公共前缀和相应明密文对记录, 输出相对应的密文/明文前缀, 之后的分组输出均匀随机分组值。

引理 2. OPRP/OPRF 转换引理[2]。敌手 A 做至多 σ个分组的询问, 则有

$$\Pr\left[A^{\rho,\rho^{-1}}\Rightarrow 1\right]-\Pr\left[A^{g,g'}\Rightarrow 1\right]\le\sigma(\sigma-1)/2^{n+1},$$

其中, $\rho\xleftarrow{\$}OPerm(n)$, g,g'用均匀随机的分组值来回答询问, 除了那些被 LCPP 性质约束的分组外。

2.7 泛哈希函数

泛哈希函数广泛应用于密码方案的设计, 包括

消息鉴别码(message authentication code, MAC)[16-19]、可调加密方案[22-23]和认证加密方案[24-25]等。两个常用的泛哈希函数是 AU 函数和 AXU 函数[26]。

对于 AU 函数[26], 任意两个不同输入的输出碰撞概率是可忽略的。

定义 3. AU 函数[26]。H:$\mathcal{K}\times\mathcal{D}\to\mathcal{R}$ 是δ-AU 函数, 如果对于任意 $X,X'\in\mathcal{D},X\neq X'$,

$$\Pr\left[K\xleftarrow{\$}\mathcal{K}:H_K(X)=H_K(X')\right]\leq\delta,$$

当δ可忽略时, 我们称 H 是 AU 函数。

对于 AXU 函数[26], 任意两个不同输入的输出差分概率是可忽略的。

定义 4. AXU 函数[26]。H:$\mathcal{K}\times\mathcal{D}\to\mathcal{R}$ 是一个δ-AXU 函数, 如果对于任意 $X,X'\in\mathcal{D},X\neq X',Y\in\mathcal{R}$,

$$\Pr\left[K\xleftarrow{\$}\mathcal{K}:H_K(X)\oplus H_K(X')=Y\right]\leq\delta,$$

当δ可忽略时, 我们称 H 是 AXU 函数。

显然, H 如果是δ-AXU 函数, 它也是δ-AU 函数, 因为 AU 函数是 $Y=0$ 时的 AXU 函数的特例。如果 H:$\mathcal{K}\times\mathcal{D}\to\mathcal{R}$ 是δ-AXU 函数, 则H':$\mathcal{K}\times\mathcal{D}\times\mathcal{R}\to\mathcal{R}$, $H'_K(X,X')=H_K(X)\oplus X'$是$\delta$-AU 函数。

AU 函数常被用作扩展伪随机函数 PRF 的输入长度, 这是因为 PRF 和 AU 函数的复合仍然是 PRF。

引理 3. PRF(AU) = PRF[27]。F:$\mathcal{K}\times\{0,1\}^n\to\{0,1\}^m$ 是一个带密钥的函数, H:$\mathcal{K}'\times\mathcal{D}\to\{0,1\}^n$ 是δ-AU 函数, F 和 H 的复合为函数 $FH_{K_1,K_2}(X)=F_{K_1}(H_{K_2}(X))$。对任意的 PRF 敌手 A 询问 q 次来攻击 FH, 存在一个 PRF 敌手 B 询问 q 次, 用与 A 相近的时间来攻击 F, 使得

$$Adv_{FH}^{prf}(A)\leq Adv_F^{prf}(B)+q^2\delta/2。$$

2.8 认证加密方案

一个传统的认证加密方案 AE:$\mathcal{K}\times\mathcal{N}\times\mathcal{A}\times\mathcal{M}\to\mathcal{C}$, 其中 $\mathcal{K},\mathcal{N},\mathcal{A},\mathcal{M},\mathcal{C}$ 分别为密钥空间、Nonce 空间、关联数据(associated data)空间、消息空间、密文空间, 加密过程中 Nonce、关联数据、明文消息三元组$(N,A,M)\in\mathcal{N}\times\mathcal{A}\times\mathcal{M}$ 经过加密部分得到密文 $C\in\mathcal{C}$, 即 $C=AE(K,N,A,M)=AE_K(N,A,M)$, 然后将$(N,A,C)$发送给解密方。认证加密方案得解密过程定义为 AE^{-1}:$\mathcal{K}\times\mathcal{N}\times\mathcal{A}\times\mathcal{C}\to\mathcal{M}\cup\{\perp\}$。解密方收到三元组$(N',A',C')$, 解密时一般先解出明文消息$M'$, 明文消息$M'$暂时不输出, 然后校验收到的$(N',A',C')$是否有效, 是则输出明文消息 M', 否则输出表示无效的字符$\perp$。一类认证加密方案要求加密时, N 的取值不重复, 我们将这种方案称为基于 Nonce 的认证加密方案。若 Nonce 可重复, 则称为抵抗 Nonce 误用的认证加密方案。

认证加密方案的安全性包含机密性和完整性两个方面, 其机密性定义为, 密文不会泄露明文任意一比特的信息, 即对于任意一个敌手 A 在选择明文攻击下, 密文和相同长度的随机字符串是不可区分的, 也即敌手 A 的优势为

$$Adv_{AE}^{ind-cpa}(A)=\Pr[A^{AE_K}\Rightarrow 1]-\Pr[A^{\rho}\Rightarrow 1],$$

其中, ρ是一个理想化的加密过程, 对每次询问均输出与 AE_K输出长度相同的随机值。

完整性定义为密文的不可伪造性, 即敌手 A 通过加密和解密询问, 来伪造一个新的没有询问过的三元组(N^*,A^*,C^*)通过验证的概率是可忽略的, 敌手 A 的优势为

$$Adv_{AE}^{int-ctxt}(A)=Pr\left[A^{AE_K,\,AE_K^{-1}}\ forges\right]。$$

我们也可以用解密算法与函数$\perp$的不可区分性来定义完整性, 函数$\perp$的输入与解密算法的输入相同, 输出永远为无效字符, 则敌手 A 可以询问加密和解密算法, 其优势可以定义为

$$Adv_{AE}^{int-ctxt}(A)=Pr\left[A^{AE_K,AE_K^{-1}}\Rightarrow 1\right]-Pr[A^{AE_K,\perp}\Rightarrow 1]。$$

在线认证加密方案(online authenticated encryption scheme, OAE)在语法的定义与传统的一致, 在安全性定义上稍有不同, 我们将在第 6 节给出在线认证加密方案的安全性定义。

3 对 POE 的分析

3.1 POE 的结构

POE[4]由 Abed 等在 2014 年作为认证加密方案 POET 的核心组件提出, 是一个使用 HEH 结构的在线密码。POET 发表于 FSE 会议, 并提交为 CAESAR 竞赛的候选算法, 不过由于 Nandi[20]的攻击而终止于第二轮。

POE 的三层结构均为带密钥的在线置换, 每层使用一个独立随机的密钥。第一层为哈希层, 是基于泛哈希函数 F:$\mathcal{K}\times\{0,1\}^n\to\{0,1\}^n$的 CFB 模式, 第二层为基于分组密码 E 的 ECB 模式, 第三层也是哈希层, 是使用函数 F 的 CFB 模式的逆, 见图 1。

POE 的加密过程具体如下:

输入: $P=(P[1],P[2],\cdots,P[m])$

1) 第一层为 $CFB[F_{K_1}]$:$X_i=F_{K_1}(X_{i-1})\oplus P[i]$, $i=1,2,\cdots,m$, X_0为常数。

2) 第二层为 $ECB[E_{K_2}]$:$Y_i=E_{K_2}(X_i),i=1,2,\cdots,m$。

3) 第三层为 $CFB^{-1}[F_{K_3}]$:$C[i]=F_{K_3}(Y_{i-1})\oplus Y_i,i=1,2,\cdots,m$, Y_0为常数。

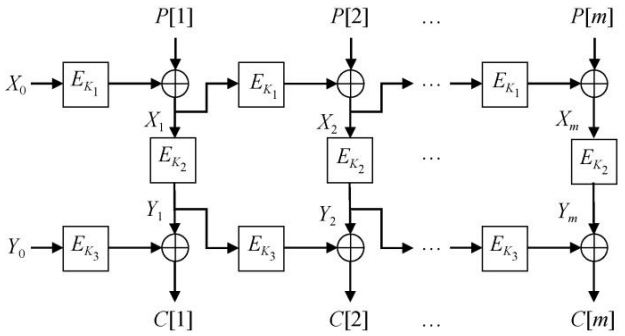


图 1　在线密码 POE, 其中X_0和Y_0为常数

Figure 1　POE, where X_0 and Y_0 are two constants

输出: $C=(C[1],C[2],\cdots,C[m])$。

由此我们可以将 POE 定义为 $CFB[F_{K_1}]-ECB[E_{K_2}]-CFB^{-1}[F_{K_3}]$。

3.2　AXU 性质并不充分

POE 声称是满足 OPRP-CCA 安全性的, 如果哈希层的 F 函数是 AXU 函数且底层分组密码 E 是 PRP-CCA 安全的。但实际上, 如果哈希层的 F 函数实例化为某些特殊的 AXU 函数, POE 甚至不是 OPRP-CPA 安全的。

如果我们找到两个不同的明文$(P[1],P[2],\cdots,P[i])$和$(P'[1],P'[2],\cdots,P'[j])$使得$CFB[F_{K_1}]^c(P[1],P[2],\cdots,P[i])=CFB[F_{K_1}]^c(P'[1],P'[2],\cdots,P'[j])$, 则对任意 $Y\in\{0,1\}^n$, 有 $CFB[F_{K_1}]^c(P[1],P[2],\cdots,P[i]),Y=CFB[F_{K_1}]^c(P'[1],P'[2],\cdots,P'[j],Y$。此时, 我们会发现第三层中$CFB^{-1}[F_{K_3}]$的每个输出分组只依赖于当前分组和前一分组这两个输入分组, 因此对于任意 $Y\in\{0,1\}^n$, $POE^c(P[1],P[2],\cdots,P[i],Y)=POE^c(P'[1],P'[2],\cdots,P'[j],Y)$。

敌手可以对加密谕言机询问$(P[1],P[2],\cdots,P[i],Y)$和$(P'[1],P'[2],\cdots,P'[j],Y)$以区分询问的是真实的加密算法还是理想的随机函数, 如果输出的密文最后两个分组是相同的, 则敌手输出 1, 否则输出 0。易知当询问的是真实的加密算法时, 敌手输出 1 的概率为 1, 当询问的是理想的随机函数时, 敌手输出 1 的概率为2^{-n}, 因此敌手对这两者的区分优势为 $1-2^{-n}$。所以攻击 POE 的关键为找到哈希层分量函数的一个输出碰撞。

接着我们将构造一些特殊的 AXU 函数使得上述攻击可行, 这些 AXU 函数是从 Nandi 的攻击中得出, 然后用哈希层的输出碰撞重新描述。

第一个实例为均匀随机自反(uniform random inverse, URI)函数$\theta:\{0,1\}^n\to\{0,1\}^n$, 容易验证其为 $2/(2^n-2)$-AXU 函数。由于$\theta(\theta(X))=X$, 则对任意 $X\in\{0,1\}^n$, 都有 $CFB[\theta]^c(X)=CFB[\theta]^c(X,0,0)$, 即两个不同的输入$(X)$和$(X,0,0)$会导致哈希层的输出碰撞。

第二个实例为有限域上乘法函数$F_K(X)=KX$, 其中 $K,X\in\{0,1\}^n$且 KX 是 K 和 X 的有限域乘法, 容易验证其为 $1/2^n$-AXU 函数。假设$X_0=0$, 则对任意 $X\in\{0,1\}^n$, 有 $CFB[F_K]^c(X)=CFB[F_K]^c(0,X)$, 即两个不同的输入$(X)$和$(0,X)$可得到哈希层的输出碰撞。

从以上分析可知, POE 中 F 函数的 AXU 的性质不能保证 $CFB[F_{K_1}]-ECB[E_{K_2}]-CFB^{-1}[F_{K_3}]$结构的安全性。一个问题是 F 函数的什么性质可以保证 $CFB-ECB-CFB^{-1}$结构的安全性? 另外 CFB 模式只是构造哈希层的一种方式, 更一般的问题是哈希层的什么性质可以保证 HEH 结构的安全性? 为此我们提出了新的概念——在线泛哈希函数(OUHF), 满足其定义的函数可以作为哈希层的组件来使得 HEH 结构达到安全性要求。

4　在线泛哈希函数

4.1　OAU 和 OAXU 的定义

为了弥补 POE 实例化时可能会有的安全性缺陷, 我们需要改进 POE 以避免出现第 3 节中的攻击, 这就要求其哈希层的分量函数的输出碰撞概率应该是可忽略的。参照 AU 函数和 AXU 函数的定义, 我们针对以上要求提出了 OAU 函数和 OAXU 函数的概念来定义哈希层函数的性质。

对于 OAU 函数来说, 要求其分量函数的任意两个不同的输入, 输出发生碰撞的概率是可忽略的。

定义 5. OAU 函数。$G:\mathcal{K}\times\{0,1\}^{n*}\to\{0,1\}^{n*}$是带密钥的在线函数, 则 G 是一个δ-OAU 函数, 如果它的分量函数G^c是δ-AU 函数, 也即对于任意 $X,X'\in\{0,1\}^{n*},X\neq X'$,

$$\Pr\left[K\xleftarrow{\$}\mathcal{K}:G_K^c(X)=G_K^c(X')\right]\leq\delta,$$

当δ可忽略时, 我们称 G 是 OAU 函数。

对于 OAXU 函数来说, 要求其分量函数的任意两个不同的输入, 输出的差分概率分布是近乎均匀的。

定义 6. OAXU 函数。$G:\mathcal{K}\times\{0,1\}^{n*}\to\{0,1\}^{n*}$是带密钥的在线函数, 则$G$是一个$\delta$-OAXU 函数, 如果它的分量函数$G^c$是$\delta$-AXU 函数, 也即对于任意 $X,X'\in\{0,1\}^{n*},X\neq X'$和任意

$$\Pr\left[K\xleftarrow{\$}\mathcal{K}:G_K^c(X)\oplus G_K^c(X')=Y\right]\leq\delta,$$

当δ可忽略时, 我们称 G 是 OAXU 函数。

显然, 如果 G 是δ-OAXU 函数, 它也是δ-OAU 函数, 因为δ-OAU 函数是 $Y=0$ 时δ-OAXU 函数的特殊例子。

在第 3 节我们说明了使用特殊的 AXU 函数(包括均匀随机自反函数和有限域乘法函数)的 CFB 模式对不同长度的消息, 其输出不是抗碰撞的, 因此用这些函数在CFB模式下构造的哈希层不满足OAU函数的定义。

在线泛哈希函数的概念可以看作逐分组泛哈希函数(block-wise universal hash function)的在线版本, 逐分组泛哈希函数在构造可调加密方案 TET[23]时提出, 但因为 TET 哈希层的每个输出分组都依赖于输入的每个分组, 所以它不是在线的, 与我们的定义不一致。

由第 3 节中分析 POE 的过程可以知道, 如果找到 $X,X^{'}\in\{0,1\}^{n*}, X\neq X^{'}$使得 $CFB^{c}(X)=CFB^{c}(X^{'})$以高概率 p 成立, 则可以 $p-2^{-n}$的优势将 POE 和在线均匀随机置换区分开来, 而哈希层的 OAU 性质正好可以避免这种攻击, 由此弥补 POE 的安全缺陷。

在第 5 节我们证明了当哈希层是 OAU 函数、底层分组密码是 PRP-CCA 时, HEH 结构是 OPRP-CCA 安全的。

因此哈希层的OAU性质就是我们需要用来保证 HEH 结构安全性的充分条件, 接下来我们将讨论 OAU 函数的构造。

4.2 OAU 函数的构造

POE 的哈希层是 CFB 模式, 尽管当 f 是均匀随机自反函数或有限域上乘法函数时, $CFB[f]$不是 OAU 函数, 但当 f 是均匀随机函数时, 其是 OAU 函数。所以哈希层是否满足 OAU 函数的性质取决于其使用的底层函数的安全强度。

CFB 模式和 CBC 模式如图 2 所示, 其具体计算过程可以表示如下:

CFB 模式: $CFB[f](X[1..m])=Y[1..m]{:}Y[i]=f(Y[i-1])\oplus X[i], i=1,2,\cdots,m$, 其中 $Y[0]$是常数, $f\xleftarrow{\$}Func(n)$。

CBC 模式: $CBC[f](X[1..m])=\mathrm{Y}[1..m]{:}\mathrm{Y}[i]=\mathrm{f}(Y[i-1]\oplus X[i]), i=1,2,\cdots,m$, 其中 $Y[0]=0$, $f\xleftarrow{\$}Func(n)$。

我们证明当$f\xleftarrow{\$}Func(n)$时, $CFB[f]$和$CBC[f]$均为 OAU 函数, 根据引理 3, 我们有以下结果。

引理 4. CBC-MAC 是 AU 函数[28]。对不同的 $X,X^{'}\in\{0,1\}^{n*},|X|_n=m,|X^{'}|_n=m^{'},f\xleftarrow{\$}Func(n)$, 有 $\Pr[CBC-MAC[f](X)=CBC-MAC[f](X^{'})]\leq\frac{mm^{'}}{2^n}+\frac{\max\{m,m^{'}\}}{2^n}$。

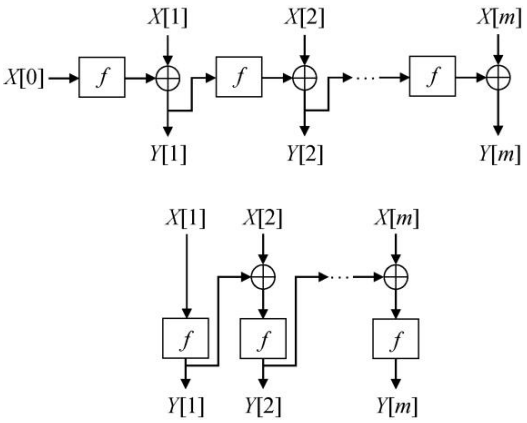


图 2 $CFB[f]$(上)和 $CBC[f]$(下)

Figure 2 $CFB[f]$ (above) and $CBC[f]$ (below)

我们容易知道使用 f 函数的 CBC 模式的每个输出的分量函数 $CBC[f]^{c}$就是使用 f 函数在 CBC 模式下构造的 MAC 方案 $CBC-MAC[f]$, 由引理 4 可知其对不同输入, 输出发生碰撞的概率是可忽略的, 因此 $CBC[f]$是 OAU 函数。

使用相同 f 函数的 CBC 模式 $CBC[f]$和 CFB 模式 $CFB[f]$ 的之间关系是 $f(CFB[f]^{c}(X[1..m]))=CBC[f]^{c}(X[0],X[1..m])$。对于不同的两个输入 $X,X^{'}\in\{0,1\}^{n*}$, $CFB[f]^{c}(X)=CFB[f]^{c}(X^{'})$ 可以推出 $f(CFB[f]^{c}(X))=f(CFB[f]^{c}(X^{'}))$, 比如$\Pr[CFB[f]^{c}(X)=CFB[f]^{c}(X^{'})]\leq\Pr[(X[0],X)=$ 则 $\Pr[CFB[f]^{c}(X)=CFB[f]^{c}(X^{'})]\leq\Pr[(X[0],X)=CBC[f]^{c}(X[0],X^{'})]$, 也即使用 f 函数构造的 CFB 模式分量函数的输出碰撞概率不超过使用相同 f 函数构造的 CBC 模式分量函数的输出碰撞概率, 因此 $CFB[f]$也是 OAU 函数。

根据 PRP/PRF 转换引理, 安全的分组密码定义为伪随机置换, 在一定的安全边界下可以看作伪随机函数, 所以 f 函数可以使用分组密码来实例化, 但这也意味着 HEH 结构处理每个分组数据需要调用 3 次分组密码, 如图 3 所示, 对一些应用来说负担过重, 我们需要一些更轻量的方法。一种方法是使用约减轮的分组密码(比如 6 轮 AES 等), 而另一种方法是使用泛哈希函数, 泛哈希函数比分组密码更轻量、运算效率更高, 可以用其来实例化 f 函数而不会带来过多的复杂运算。尽管第 3 节已经表明, 对于有限域上的乘法函数$F_K(X)=KX$, $CFB[F_K]$不是 OAU 函数, 但用其作为泛哈希函数来构造OAU函数仍然具有研究意义。

我们知道 $CFB[F_K]^{c}(X[1..m])=X[1]K^{m-1}\oplus X[2]K^{m-2}\oplus\cdots\oplus X[m]$是 GCM[25]、HCTR[22]等认证

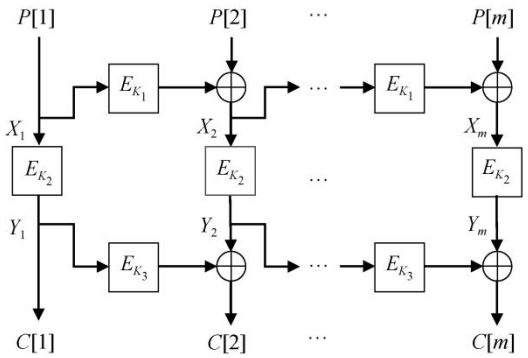


图 3 用分组密码实例化的 HEH 结构

Figure 3 The structure of HEH instantiated by block cipher

加密方案中用到的经典多项式赋值函数, 尽管 $CFB[F_K]$在$\{0,1\}^{n*}$上, 即对变长的输入来说不是 OAU 函数, 但对于固定的正整数 d, 其分量函数的输出碰撞概率是可忽略的, 即它是$\{0,1\}^{nd}$上的 OAU 函数。

为了使得处理可变长的输入数据仍能保持 OAU 性质, 我们对 $CFB[F_K]$的每个输出分组都异或上2^iL, 其中 $2=0^{n-2}10, i=1,2,\cdots, L \xleftarrow{\$} \{0,1\}^n$是另一个随机独立的密钥, 其结构如图 4 所示。我们将这种模式称作 MCFB(掩码 CFB)模式, 其分量函数可以表示为 $MCFB[F_K]^c(X[1..m]) = X[1]K^{m-1} \oplus X[2]K^{m-2} \oplus \cdots \oplus X[m] \oplus 2^mL$。

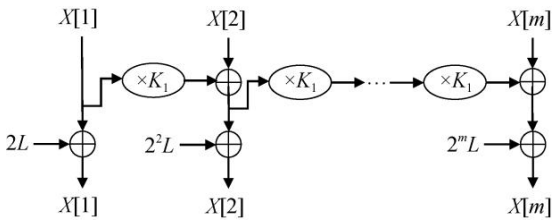


图 4 $MCFB[F_K]$

Figure 4 $MCFB[F_K]$

假设 $X,X' \in \{0,1\}^{n*}, X \neq X', |X|_n = m, |X'|_n = m'$, 如果 $m = m'$, $MCFB^c(X) \oplus MCFB^c(X')$对于 K 是非零多项式, 其次数至多为$(m-1)$, 所以在有限域中至多有$(m-1)$个根, 则 $MCFB[F_K]^c$的输出碰撞概率的界为 $(m-1)/2^n$。如果 $m \neq m'$, $MCFB^c(X) \oplus MCFB^c(X')$有一个单项式$\left(2^m \oplus 2^{m'}\right)L$, 所以输出碰撞概率为 $1/2^n$。因此 $MCFB[F_K]$是 $1/2^n$-OAU 函数。

对比 $CFB[F_K]$, $MCFB[F_K]$中每个分组异或上的 2^iL 带来的额外计算是很少量的, 对每个分组只有一个移位运算和一个异或运算, 不会带来运算性能的显著降低。

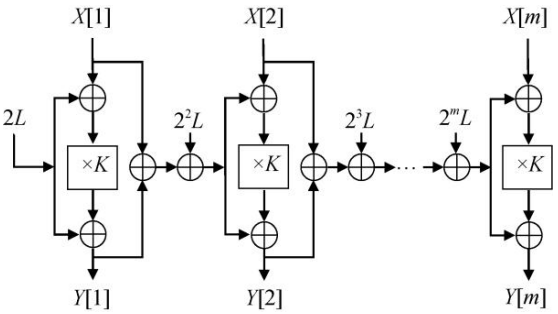


图 5 $XCH[F_K]$

Figure 5 $XCH[F_K]$

我们还可以使用 AXU 函数及其输入输出异或链接的方式来构造 OAU 函数, 如图 5 所示。这种构造思想来自 TC3[10]等工作模式的设计, 主要组件是一个小规模的带密钥的泛哈希函数F_K, 通过输入输出异或链接的方式, 形成一个在线函数 $XCH[F_K]$, 其描述如表 2 所示。

表 2 $XCH[F_K]$的计算过程

Table 2 The process of $XCH[F_K]$

$XCH[F_K](X)$
$(X[1],X[2],\cdots,X[m]) \leftarrow X$
$S[1] = 2L$
for $i = 1$ to m
$Y[i] = F_K(X[i] \oplus S[i]) \oplus S[i]$
$S[i+1] = X[i] \oplus Y[i] \oplus 2^{i+1}L$
return $Y = (Y[1],Y[2],\cdots,Y[m])$

定理 1. $XCH[F_K]$是 OAU 函数. 在 $XCH[F_K]$中, 如果$F_K(X) = KX, K \neq 0$ 且 $K \neq 1$, 则 $XCH[F_K]$是 OAU 函数, 其逆也是 OAU 函数。

证明. 先证明 $XCH[F_K]$是 OAU 函数。

我们知道

$$S[1] = 2L,\ Y[1] = X[1]K \oplus 2LK \oplus 2L,$$

当 $i \geq 2$ 时,

$$S[i] = X[i-1] \oplus Y[i-1] \oplus 2^{i+1}L,$$

$$Y[i] = (X[i] \oplus S[i])K \oplus S[i],$$

所以

$$S[i] = \bigoplus_{j=1}^{i-1} (1 \oplus K)^{i-j}(X[j] \oplus 2^jL) \oplus 2^iL,$$

$$Y[i] = X[i]K \oplus \bigoplus_{j=1}^{i-1} (1 \oplus K)^{i-j+1}(X[j] \oplus 2^jL) \oplus (1 \oplus K)2^iL$$

也即 $XCH[F_K]$的分量函数为

$$XCH^{(1)}(X[1]) = X[1]K \oplus 2L,$$

$XCH^{(i)}(X[1],X[2],\cdots,X[i]) == X[i]K \oplus \bigoplus_{j=1}^{i-1}(1 \oplus K)^{i-j+1}(X[j] \oplus 2^jL) \oplus (1 \oplus K)2^iL, i \geq 2$。

对于 $XCH^{(1)}$，当 $X[1] \neq X'[1]$ 时，若 $XCH^{(1)}(X[1]) = XCH^{(1)}(X'[1])$，则 $(X[1] \oplus X'[1])K = 0$，该方程关于 K 只有一个根，所以成立概率不超过 $1/(2^n-1)$。

对于 $(X[1],X[2],\cdots,X[i]) \neq (X'[1]), i \geq 2$，$XCH^{(i)}(X[1],X[2],\cdots,X[i]) = XCH^{(1)}(X'[1])$，则

$$X[i]K \oplus \bigoplus_{j=1}^{i-1}(1 \oplus K)^{i-j+1}(X[j] \oplus 2^jL) \oplus (1 \oplus K)2^iL \oplus X[1]K \oplus 2L = 0$$

以上次数不为零的多项式至多有 i 个根，故其成立概率不超过 $i/(2^n-1)$。

当 $(X[1],X[2],\cdots,X[i]) \neq (X'[1],X'[2],\cdots,X'[i])$ 时，若 $XCH^{(i)}(X[1],X[2],\cdots,X[i]) = XCH^{(i)}(X'[1],X'[2],\cdots,X'[i])$，则

$$(X[i] \oplus X'[i])K \oplus \bigoplus_{j=1}^{i-1}(1 \oplus K)^{i-j+1}(X[j] \oplus X'[j]) = 0,$$

由 $(X[1],X[2],\cdots,X[i]) \neq (X'[1],X'[2],\cdots,X'[i])$ 可知以上关于 K 的多项式的次数不为 0，所以方程至多有 i 个根，故其成立概率不超过 $i/(2^n-1)$。

当 $(X[1],X[2],\cdots,X[i]) \neq (X'[1],X'[2],\cdots,X'[i']), 2 \leqslant i' < i$ 时，若 $XCH^{(i)}(X[1],X[2],\cdots,X[i]) = XCH^{(i)}(X'[1],X'[2],\cdots,X'[i'])$，则

$$(X[i] \oplus X'[i])K \oplus \bigoplus_{j=1}^{i'-1}(1 \oplus K)^{i-j+1}(X[j] \oplus X'[j]) \oplus \bigoplus_{j=i'}^{i-1}(1 \oplus K)^{i-j+1}(X[j] \oplus 2^jL) \oplus (1 \oplus K)2^{i-i'}L = 0,$$

易知以上关于 K 的多项式的次数不为 0，所以方程至多有 i 个根，故其成立概率不超过 $i/(2^n-1)$。

故而 $XCH[F_K]$是 OAU 函数。

再证 $XCH[F_K]$（$K \neq 0$）的逆 $XCH^{-1}[F_K]$也是 OAU 函数。

$XCH^{-1}[F_K]$的分量函数为

$XCH^{-1,(1)}(Y[1]) = Y[1]K^{-1} \oplus 2L$,

$$XCH^{-1,(i)}(Y[1],Y[2],\cdots,Y[i]) = Y[i]K^{-1} \oplus \bigoplus_{j=1}^{i-1}(1 \oplus K^{-1})^{i-j+1}(Y[j] \oplus 2^jL) \oplus (1 \oplus K^{-1})2^iL, i \geq 2。$$

令 $K' = K^{-1}$，则 $XCH^{-1}[F_K]$的分量函数与 $XCH[F_K]$的分量函数相同，故而 $XCH[F_K]$的逆也是 OAU 函数。

定义了 OAU 函数及提出可行的构造方法后，我们可以用其实例化 HEH 结构的哈希层，并证明使用 OAU 函数的 HEH 结构的安全性。

5 HEH 结构及其安全性

5.1 HEH 的结构

HEH 结构的一般形式如图 6 所示，第一层为带密钥的在线置换 G:$\mathcal{K} \times \{0,1\}^{n*} \to \{0,1\}^{n*}$；第二层为 ECB 模式，底层分组密码为 E:$\mathcal{K}' \times \{0,1\}^n \to \{0,1\}^n$；第三层为第一层在线置换 G 的逆G^{-1}。每一层均使用随机独立选取的密钥，这种结构记作 $HEH[G,E]$，其加解密过程分别记作 $\mathcal{E}[G,E]$和 $\mathcal{D}[G,D]$，如表 3 所示。

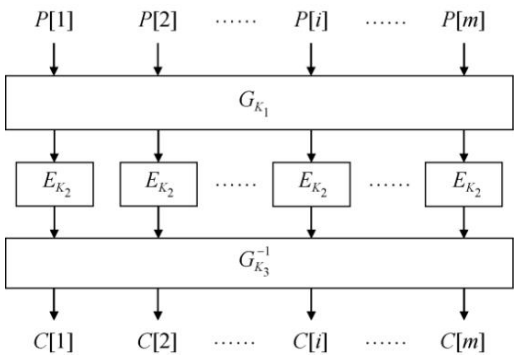


图 6 HEH 结构

Figure 6 The structure of HEH

表 3 $HEH[G,E]$的加解密过程

Table 3 The encryption and decryption of $HEH[G,E]$

$\mathcal{E}[G,E](P)$	$\mathcal{D}[G,D](C)$
$X = G_{K_1}(P)$	$Y = G_{K_3}(C)$
$Y = ECB[E_{K_2}](X)$	$X = ECB[D_{K_2}](Y)$
$C = G_{K_3}^{-1}(Y)$	$P = G_{K_1}^{-1}(X)$
return C	return P

我们证明当哈希层的在线置换 G 是 OAU 函数、中间 ECB 层使用的 E 是 PRP-CCA 安全的分组密码时，$HEH[G,E]$是 OPRP-CCA 安全的。

5.2 HEH 的安全性

定理 2. $HEH[G,E]$是 OPRP-CCA 安全的。在 $HEH[G,E]$中，如果 G 是δ-OAU，则对于任意的 OPRP-CCA 敌手 A 攻击 $HEH[G,E]$总共询问σ个分组，存在一个 PRP-CCA 敌手 B 攻击 E 询问σ次且运行时间与敌手 A 相近，使得

$$Adv_{HEH[G,E]}^{oprp-cca}(A) \leq Adv_E^{prp-cca}(B) + \sigma^2/2^n + \sigma^2\delta。$$

证明. 我们的目标是证明$(\mathcal{E}[G,E],\mathcal{D}[G,D])$和$(\rho,\rho^{-1})$之间的不可区分性，其中$\rho \xleftarrow{\$} OPerm(n)$为随机选取的在线置换。我们逐步将 $HEH[G,E]$理想化，通过相邻谕言机对之间的不可区分性证明其安全性。

具体地, 我们在这两个谕言机对之间添加了 3 个逐步理想化的谕言机对, 如图 7 所示。我们只需要证明相邻两个谕言机对间的不可区分性, 再利用混杂(hybrid)技术即可完成证明。

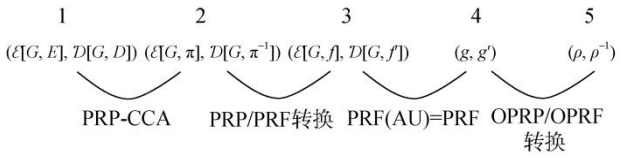


图 7 证明路线

Figure 7 Roadmap of the proof

1-2: $(\mathcal{E}[G,E],\mathcal{D}[G,D])$--$(\mathcal{E}[G,\pi],\mathcal{D}[G,\pi^{-1}])$, 其中$\pi \xleftarrow{\$} Perm(n)$。如果将第一个谕言机对中的$(E_{K_2},D_{K_2})$替换为$(\pi,\pi^{-1})$, 便得到第二个谕言机对, 所以它们之间的差别只是底层分组密码在选择密文攻击下的安全边界。敌手 B 模拟 A 的询问过程, 并返回 A 的返回值, 则有

$$\Pr\left[A^{\mathcal{E}[G,E],\mathcal{D}[G,D]} \Rightarrow 1\right] - \Pr\left[A^{\mathcal{E}[G,\pi],\mathcal{D}[G,\pi^{-1}]} \Rightarrow 1\right] = Adv_E^{prp-cca}(B)。$$

2-3: $(\mathcal{E}[G,\pi],\mathcal{D}[G,\pi^{-1}])$与$(\mathcal{E}[G,f],\mathcal{D}[G,f'])$, 其中$f,f' \xleftarrow{\$} Func(n)$, f 和 f'是两个均匀随机函数。这两个谕言机对之间的差别只是将第一个谕言机对中的随机置换及其逆(π,π^{-1})替换为两个随机函数对(f,f')。根据 PRP/PRF 转换引理[21], 有

$$\Pr\left[A^{\mathcal{E}[G,\pi],\mathcal{D}[G,\pi^{-1}]} \Rightarrow 1\right] - \Pr\left[A^{\mathcal{E}[G,f],\mathcal{D}[G,f']} \Rightarrow 1\right] \leq \sigma(\sigma-1)/2^{n+1}。$$

3-4: $(\mathcal{E}[G,f],\mathcal{D}[G,f'])$与$(g,g')$, 其中$(g,g')$用随机分组值回答询问中那些除了被 LCPP 性质限制的分组, 而那些满足 LCPP 性质的分组则对应输出与其有相同前缀的明文分组得到的密文。这两对谕言机的具体过程如表 4, 其中左边为$(\mathcal{E}[G,f],\mathcal{D}[G,f'])$的过程, 右边为$(g,g')$的过程。

对于$(\mathcal{E}[G,f],\mathcal{D}[G,f'])$过程, f 函数维护一个输入输出记录(X,Y), 若输入在记录中则给出对应的输出, 如果输入不在记录中, 则取随机串输出, 并将该输入输出加入记录。f'函数与 f 函数共享记录, 若 f'函数的输出有相同的 Y 在记录中, 则输出对应的 X, 否则输出随机的 X, 并将(X,Y)加入记录中。对于(g,g')过程, P^*/C'^*为之前询问过的且与当前询问有最大的最长公共前缀的明文/密文, C^*/P'^*为询问P^*/C'^*得到的密文/明文。

现我们将$(\mathcal{E}[G,f],\mathcal{D}[G,f'])$逐分组理想化为$(\mathcal{E}^{(a)}[G,f],\mathcal{D}^{(a)}[G,f'])$, 两对谕言机的定义如表 5, 其中右半部分中的 a 为第 a 个分组, F 和 F':$\{0,1\}^{n*} \rightarrow \{0,1\}^n$为随机函数, 与 f 和 f'类似, 对已经出现过的输入给出对应的输出, 其他给出随机输出。我们只需

表 4 $(\mathcal{E}[G,f],\mathcal{D}[G,f'])$与$(g,g')$

Table 4 $(\mathcal{E}[G,f],\mathcal{D}[G,f'])$ and (g,g')

$(\mathcal{E}[G,f],\mathcal{D}[G,f'])$	(g,g')
$\mathcal{E}[G,f](P)$	$g(P)$
$(P[1],P[2],\ldots,P[m]) \leftarrow P$	$(P[1],P[2],\ldots,P[m]) \leftarrow P$
for $i = 1$ to m	$l = LLCP(P,P^*)$
$X[i] = G^c_{K_1}(P[1..i])$	for $i = 1$ to l
$Y[i] = f(X[i])$	$C[i] = C^*[i]$
$C[i] = G^{c,-1}_{K_3}(Y[1..i])$	for $i = l+1$ to m
return $C = (C[1],C[2],\ldots,C[m])$	$C[i] \xleftarrow{\$} \{0,1\}^n$
	return $C = (C[1],C[2],\ldots,C[m])$
$\mathcal{D}[G,f'](C')$	$g'(C')$
$(C'[1],C'[2],\ldots,C'[m]) \leftarrow C'$	$(C'[1],C'[2],\ldots,C'[m]) \leftarrow C'$
for $i = 1$ to m	$l = LLCP(C',C'^*)$
$Y'[i] = G^c_{K_3}(C'[1..i])$	for $i = 1$ to l
$X'[i] = f'(Y'[i])$	$P'[i] = C'^*[i]$
$P'[i] = G^{c,-1}_{K_1}(X'[1..i])$	for $i = l+1$ to m
return $P' = (P'[1],\ldots,P'[m])$	$P'[i] \xleftarrow{\$} \{0,1\}^n$
	return $P' = (P'[1],\ldots,P'[m])$

表 5 $(\mathcal{E}[G,f],\mathcal{D}[G,f'])$与$(\mathcal{E}^{(a)}[G,f],\mathcal{D}^{(a)}[G,f'])$

Table 5 $(\mathcal{E}[G,f],\mathcal{D}[G,f'])$ and $(\mathcal{E}^{(a)}[G,f],\mathcal{D}^{(a)}[G,f'])$

$(\mathcal{E}[G,f],\mathcal{D}[G,f'])$	$(\mathcal{E}^{(a)}[G,f],\mathcal{D}^{(a)}[G,f'])$
$\mathcal{E}[G,f](P)$	$\mathcal{E}^{(a)}[G,f](P)$
$(P[1],P[2],\ldots,P[m]) \leftarrow P$	$(P[1],P[2],\ldots,P[m]) \leftarrow P$
for $i = 1$ to m	for $i = 1$ to a
$X[i] = G^c_{K_1}(P[1..i])$	$Y[i] = F(P[1..i])$
$Y[i] = f(X[i])$	$C[i] = G^{c,-1}_{K_3}(Y[1..i])$
$C[i] = G^{c,-1}_{K_3}(Y[1..i])$	for $i = a+1$ to m
return $C = (C[1],C[2],\ldots,C[m])$	$X[i] = G^c_{K_1}(P[1..i])$
	$Y[i] = f(X[i])$
	$C[i] = G^{c,-1}_{K_3}(Y[1..i])$
	return $C = (C[1],C[2],\ldots,C[m])$
$\mathcal{D}[G,f'](C')$	$\mathcal{D}^{(a)}[G,f'](C')$
$(C'[1],C'[2],\ldots,C'[m]) \leftarrow C'$	$(C'[1],C'[2],\ldots,C'[m]) \leftarrow C'$
for $i = 1$ to m	for $i = 1$ to a
$Y'[i] = G^c_{K_3}(C'[1..i])$	$X'[i] = F'(C'[1..i])$
$X'[i] = f'(Y'[i])$	$P'[i] = G^c_{K_1}(X[1..i])$
$P'[i] = G^{c,-1}_{K_1}(X'[1..i])$	for $i = a+1$ to m
return $P' = (P'[1],\ldots,P'[m])$	$Y'[i] = G^c_{K_3}(C'[1..i])$
	$X'[i] = f'(Y'[i])$
	$P'[i] = G^{c,-1}_{K_1}(X'[1..i])$
	return $P' = (P'[1],\ldots,P'[m])$

要说明 $(\mathcal{E}[G,f],\mathcal{D}[G,f'])$ 与 $(\mathcal{E}^{(1)}[G,f],\mathcal{D}^{(1)}[G,f'])$, $(\mathcal{E}^{(2)}[G,f],\mathcal{D}^{(2)}[G,f'])$, …, $(\mathcal{E}^{(m)}[G,f],\mathcal{D}^{(m)}[G,f'])$相邻两个之间的不可区分性即可, 而每相邻两个之间只有1个分组的差别, 前一个是使用 G 函数和 f 函数生成第三层的输入, 后一个使用随机函数产生第三层输入, 根据引理 3(PRF(AU)=PRF)[27]可知, 相邻两个谕言机对是不可区分的, 敌手对$(\mathcal{E}[G,f],\mathcal{D}[G,f'])$与$(\mathcal{E}^{(m)}[G,f],\mathcal{D}^{(m)}[G,f'])$的区分优势为$\sigma^2\delta/2$。

$(\mathcal{E}^{(m)}[G,f],\mathcal{D}^{(m)}[G,f'])$与$(g,g')$两者对于最长公共前缀的处理是一致的, 其区分点在之后的部分。$(\mathcal{E}^{(m)}[G,f],\mathcal{D}^{(m)}[G,f'])$是将随机输入通过 G 函数来处理得到输出, (g,g')是取随机字符串作为输出, 对于一个分组其区分概率为δ, 所以敌手的区分优势为$\sigma^2\delta/2$。因此有

$$\Pr\left[A^{\mathcal{E}[G,f],\mathcal{D}[G,f']}\Rightarrow 1\right]-\Pr\left[A^{g,g'}\Rightarrow 1\right]\le\sigma^2\delta。$$

4-5: (g,g')与(ρ,ρ^{-1})。这两个谕言机对之间只是在线伪随机函数和在线伪随机置换的距离, 根据OPRP/OPRF 转换引理[2]可得

$$\Pr\left[A^{g,g'}\Rightarrow 1\right]-\Pr\left[A^{\rho,\rho^{-1}}\Rightarrow 1\right]\le\sigma(\sigma-1)/2^{n+1}。$$

综合以上, 有

$$\begin{aligned}Adv^{oprp-cca}_{HEH[G,E]}(A)&=\Pr\left[A^{\mathcal{E}[G,E],\mathcal{D}[G,D]}\Rightarrow 1\right]\\&\quad-\Pr\left[A^{\rho,\rho^{-1}}\Rightarrow 1\right]\\&\le Adv^{prp-cca}_{E}(B)+\sigma(\sigma-1)/2^{n+1}\\&\quad+\sigma^2\delta+\sigma(\sigma-1)/2^{n+1}\\&\le Adv^{prp-cca}_{E}(B)+\sigma^2/2^{n}+\sigma^2\delta。\end{aligned}$$

定理得证。

5.3 在线密码的实例化

我们可以使用之前构造的 OAU 函数来实例化 HEH 结构。类似 POE 的构造, 我们可以实例化 HEH 结构的在线密码 OC 为 $MCFB[F_{K_1}]-ECB[E_{K_2}]-MCFB^{-1}[F_{K_3}]$, 具体结构如图 8 所示。

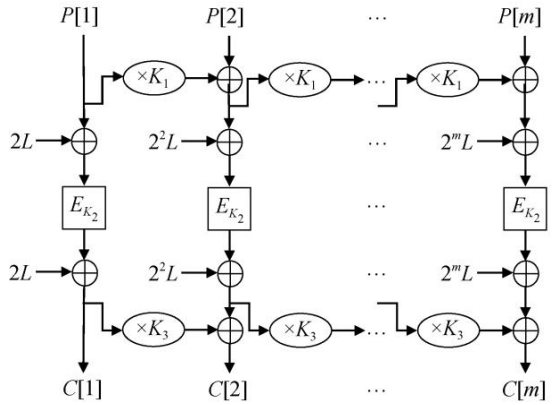


图 8 $MCFB[F_{K_1}]-ECB[E_{K_2}]-MCFB^{-1}[F_{K_3}]$

Figure 8 $MCFB[F_{K_1}]-ECB[E_{K_2}]-MCFB^{-1}[F_{K_3}]$

因为 $MCFB[F_{K_1}]$是 OAU 函数, $ECB[E_{K_2}]$使用 PRP-CCA 安全的分组密码, 所以根据 5.2 节的证明, 在线密码的实例 $MCFB[F_{K_1}]-ECB[E_{K_2}]-MCFB^{-1}[F_{K_3}]$是 OPRP-CCA 安全的。

如果我们将 ECB 层前后的异或运算从哈希层剥离出来, 将其与分组密码放在一起作为整体来看, 它们构成 XEX 结构的可调分组密码(TBC)[24], 即第 i 个分组使用的可调分组密码为 $XEX^{(i)}[E_{K_2}](X)=E_{K_2}(2^iL\oplus X)\oplus 2^iL, i\in\{1,2,\cdots,m\}$。由于不同的调柄(tweak)会使得可调分组密码得到一个新的独立随机的置换, 所以即使对相同的输入, 不同的调柄决定的可调分组密码也会输出随机的字符串, 这样哈希层不同位置(这些位置决定了可调分组密码的调柄值)的碰撞也不会对安全性产生影响, 而哈希层本身对不同明文相同位置的输出是抗碰撞的, 因此我们将 POE 中的分组密码更换为可调分组密码可以解决其安全性问题。

以上只是 OAU 实例化后得到的 HEH 结构的一个实例, OAU 函数还可以有很多其他可能的构造方式, 这些构造实例化后并不一定能经过结构重组将中间层看作可调分组密码, 所以 $OAU-ECB-OAU^{-1}$结构更具有一般性, 这也是我们提出 OAU 函数概念的原因和意义。

5.4 效率

我们对比了在线密码 $HEH[G,E]$与已有的 CCA 安全的在线密码的性能, 以 m 个分组的明文加密所需要的有限域乘法次数和调用分组密码的次数来评估, 如表 6 所示。其中乘法次数计算为将哈希函数使用有限域乘法实例化时乘法运算的使用次数。对于每一分组, 我们的构造 $HEH[G,E]$只比 HCBC 等串行的在线密码多一到两次乘法运算, 在具体实现时可以通过第二层和第三层的并行计算提高效率; 相比非串行的在线密码 POE 只多了掩码部分的一次移位运算(乘以 2); 非串行的 OleF 需要两次分组密码调用, 而乘法是比分组密码更轻量的运算, 在相同并行度的情况下, $HEH[G,E]$可以具有更高的效率。

6 在线认证加密方案 OAE

6.1 OAE 的构造

从一个 OPRP-CCA 安全的在线密码 OC, 很容易构造安全的在线认证加密方案 OAE, 我们只需要仔细地设计其认证部分和对加密部分稍作调整, 包括关联数据的处理、Nonce 的处理和认证方式的设计。

如果我们有使用 HEH 结构构造的在线密码为 $OC=HEH[G,E]$, 其逆为 OC^{-1}, OC 和 OC^{-1}的第 i 个

分量函数分别为 $OC^{(i)}$和 $OC^{-1,(i)}$, 则可以构造如图 9~图 11 的在线认证加密方案 $OAE[OC]$。

表 6 $HEH[G,E]$与现有在线密码的效率对比

Table 6 The Comparation of performance between $HEH[G,E]$ and existing online cipher

	乘法	分组密码	类型
HCBC2[2]	$2m$	m	串行
MCBC[9]	m	m	串行
MHCBC[9]	0	$2m$	串行
TC2/3[10]	m	m	串行
XTC[12]	$2m$	m	串行
POE[4]	$2m$	m	非串行
OleF[13]	0	$2m$	非串行
$HEH[G,E]$	$2m-2$	m	非串行

我们将 $OAE[OC]$的加解密函数分别表示为 OAE^e:$\mathcal{N}\times\mathcal{A}\times\mathcal{M}\to\mathcal{C}$ 和 OAE^d:$\mathcal{N}\times\mathcal{A}\times\mathcal{C}\to\mathcal{M}\cup\{\perp\}$。

$OAE^e(N,A,M)$的过程如下:

1) 处理关联数据和 Nonce, 如图 9 所示。

$Auth=A[1]K_4^{a+2}\oplus A[2]K_4^{a+1}\oplus\cdots\oplus A[a]K_4^3\oplus aK_4^2\oplus NK_4$, 其中 $A\in\mathcal{A}, A=(A[1],A[2],\cdots,A[a])$, $a=|A|_n, N\in\mathcal{N}$。

2) 加密消息 $M\in\mathcal{M}$, 如图 10 所示。

加密 $M=(M[1],M[2],\cdots,M[m]), m=|M|_n$得到密文 $C=(C[1],C[2],\cdots,C[m],C[m+1])$。

$C[1]=OC^{(1)}(M[1]\oplus Auth)\oplus Auth$,

$C[i]=OC^{(i)}(M[1]\oplus Auth,M[2],\cdots,M[i]), 2\le i\le m$,

$C[m+1]=OC^{(m+1)}(M[1]\oplus Auth,M[2],\cdots, M[m],0^n\oplus 2^mL)\oplus 2^mL$。

$OAE^d(N,A,C)$的过程描述如下:

1) 与 OAE^e同样的方式处理关联数据和 Nonce 得到 $Auth$。

2) 解密消息 $C\in\mathcal{C}$, 如图 11 所示。

解密 $m+1$ 个分组的密文 $C=(C[1],C[2],\cdots,C[m],C[m+1])$ 得到 $m+1$ 个分组的明文 $(M[1],M[2],\cdots,M[m],M[m+1])$。

$M[1]=OC^{-1,(1)}(C[1]\oplus Auth)\oplus Auth$,

$M[i]=OC^{-1,(i)}(C[1]\oplus Auth,C[2],\cdots,C[i]), 2\le i\le m$,

$M[m+1]=OC^{-1,(m+1)}(C[1]\oplus Auth,C[2],\cdots, C[m], C[m+1])\oplus 2^mL$。

3) 验证并输出。

若解密得到的明文分组 $M[m+1]=0^n$, 则输出 $M=(M[1],M[2],\cdots,M[m])$作为明文消息, 否则输出无效字符 $\perp$, 解密得到的 $(M[1],M[2],\cdots,M[m], M[m+1])$不能输出。

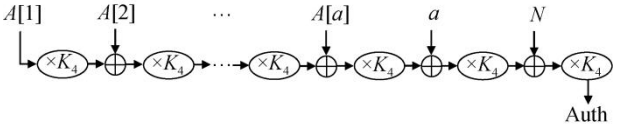


图 9 关联数据和 Nonce 的处理过程

Figure 9 The process dealing with Associated Data and Nonce

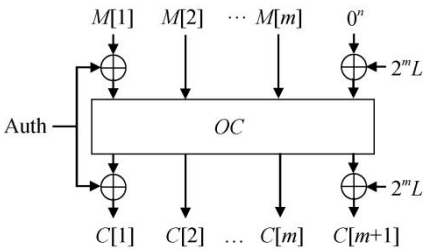


图 10 加密过程中对消息的处理

Figure 10 The encryption dealing with plaintext message

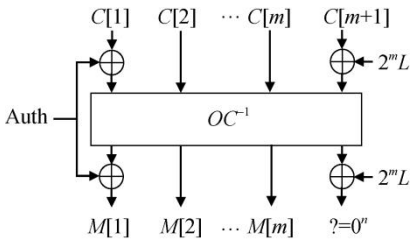


图 11 解密过程中对密文的处理

Figure 11 The decryption dealing with ciphertext

对于关联数据和 Nonce 的处理与在线密码的第一层相同, 算法实现时可以复用, 节约存储成本。在认证上我们采用在消息末尾添加冗余分组然后解密时校验冗余分组的方式。相对于在线密码 OC, 除了关联数据和 Nonce 的处理部分, 只有消息的第一个分组和冗余消息的分组稍有不同。若关联数据的分组数为 a, 消息的分组数为 m, 则在线认证加密 OAE 做乘法运算的次数为 $a+3+2m$, 调用分组密码次数为 $m+1$。当关联数据较短、消息长度足够长时, 在线认证加密 OAE 处理每个分组的平均运行效率在理论上与在线密码的效率相当。

6.2 安全性定义

在线认证加密方案与传统的认证加密方案在语法定义上一致, 只是要求可以在线处理, 这意味着密文分组的生成只与明文当前分组和之前的分组有关。

对于在线认证加密方案中的加密函数

$OAE^e(N,A,M)$, 我们也可以讨论其最长前缀保持(LCPP)性质。

对于不同的输入(N,A,M)和(N',A',M'), 若$(N,A)\neq(N',A')$, 则其最长公共前缀为空字符串, 即$LCP_n((N,A,M),(N',A',M'))=\epsilon$; 若$(N,A)=(N',A')$, 其最长公共前缀定义为两个消息的最长公共前缀, 即$LCP_n((N,A,M),(N',A',M'))=LCP_n(M,M')$, 简言之, 我们可以将在线认证加密方案中的最长公共前缀按之前的定义表示出, 即$LCP_n((N,A,M),(N',A',M'))=LCP_n(N||A||M,N'||A'||M')$, 最长公共前缀的长度表示为$LLCP_n((N,A,M),(N',A',M'))=\left|LCP_n((N,A,M),(N',A',M'))\right|_n$。

设函数$OF:N\times A\times\{0,1\}^{n*}\to\{0,1\}^{n*}$是$\{0,1\}^{n*}$上的在线置换, 其逆为$OF^{-1}$, 若对于两个不同的$X,X'\in\{0,1\}^{n*}$且$LLCP_n((N,A,X),(N',A',X'))=l$有$LLCP_n(OF(N,A,X),OF(N,A,X'))=LLCP_n(OF^{-1}(N,A,X),OF^{-1}(N,A,X'))=l$, 则在线置换$OF$具有最长公共前缀保持性(LCPP)。当$OF$函数处理不同的$(N,A)$二元组时, 不能保持其最长前缀。

根据在线认证加密方案的最长公共前缀保持性质, 我们可以定义在线认证加密方案的安全性。

机密性定义为除了相同最长公共前缀得到的密文分组受到LCPP性质的约束外, 其他密文分组要求与随机比特串是不可区分的, 也即如果(N,A)不重复, 每个密文分组都是随机的, 即使(N,A)重复, 在最长公共前缀之后, 输出的密文分组仍然是随机值。设函数of是从上述定义的所有的保持最长公共前缀的函数OF中随机选取的函数, 其输出长度与使用认证加密方案加密OAE^e相同输入的明文得到的密文长度相同。敌手A在选择明文攻击下区分OAE^e和of的优势为

$$Adv_{OAE}^{ind-cpa}(A)=Adv_{OAE^e[OC]}^{of}(A)=\Pr\left[A^{OAE^e}\Rightarrow 1\right]-\Pr\left[A^{of}\Rightarrow 1\right],$$

若敌手至多询问q_e次, 区分OAE^e与函数of的最大优势为可忽略的, 则在线认证加密方案能够保证选择明文攻击下的机密性。之后敌手A对两个对象Π与xxx的区分优势均采用类似定义和记法, 即$Adv_{\Pi}^{xxx}(A)=\Pr[A^{\Pi}\Rightarrow 1]-\Pr[A^{xxx}\Rightarrow 1]$。

完整性与传统的认证加密方案一致, 敌手不能通过篡改和伪造使得新的三元组(N,A,C)通过解密验证, 我们也可以使用不可区分性来定义, 即解密过程中用于验证的字符串的生成是随机的, 在我们的OAE方案中就是由三元组(N,A,C)中最后一个密文分组解密得到的明文(即$M[m+1]$)应该是随机的。

设$OAE^t:\mathcal{N}\times\mathcal{A}\times\mathcal{C}\to\{0,1\}^n$为在线认证加密方案解密过程$OAE^d(N,A,C)$中生成用于认证的字符串的函数(即$OC^{-1,(m+1)}$)。令$of'$是随机函数, 其输入与在线认证加密方案解密时的输入相同, 其输出为随机字符串, 长度与OAE^t相同。若敌手可以询问在线认证加密方案的加密部分OAE^e和解密部分中的OAE^t, 分别询问q_e次和q_d次, 但OAE^t和of'不能区分, 则方案具有密文完整性, 敌手B的优势为

$$\begin{aligned}Adv_{OAE}^{int-ctxt}(B)&=Adv_{OAE^t[OC^{-1,(m+1)}]}^{of'}(B)\\&=\Pr\left[A^{OAE^e,OAE^t}\Rightarrow 1\right]\\&\quad-\Pr\left[A^{OAE^e,of'}\Rightarrow 1\right].\end{aligned}$$

6.3 安全性证明

以下我们证明使用第5节中OPRP-CCA安全的$HEH[G,E]$结构的在线密码OC构造的在线认证加密方案$OAE[OC]$在6.2节的定义下的安全性。

定理 3. $OAE[OC]$是安全的。在$OAE[OC]$中, 如果使用三层结构$HEH[G,E]$构造的OC是OPRP-CCA安全的, 则对于任意的选择明文攻击的敌手A攻击$OAE[OC]$的机密性, 询问加密部分q_e次, 总共询问σ个分组, 询问解密部分q_d次, 存在一个PRP-CCA敌手C攻击分组密码E询问σ次且运行时间与A相近, 使得

$$Adv_{OAE}^{ind-cpa}(A)\le Adv_{E}^{prp-cca}(C)+\sigma^2/2^n+\sigma^2\delta+q_e^2/2^n;$$

对于任意的敌手B攻击$OAE[OC]$的完整性, 询问加密部分q_e次, 总共询问σ个分组, 询问解密部分q_d次, 成功的优势

$$Adv_{OAE}^{int-ctxt}(B)\le q_d^2\delta+(q_e^2+q_d^2)/2^n。$$

证明. 我们先证明$OAE[OC]$在选择明文攻击下的机密性, 再证明其完整性。

1)首先我们计算敌手区分OAE的加密部分$OAE^e[OC]$和of的优势, q_e次加密询问的三元组(N,A,M)是两两不同的。

由OAE加密过程OAE^e中的$Auth=A[1]K_4^{a+2}\oplus A[2]K_4^{a+1}\oplus\cdots\oplus A[a]K_4^3\oplus aK_4^2\oplus NK_4$, $C[1]=OC^{(1)}(M[1]\oplus Auth)\oplus Auth$和$C[i]=OC^{(i)}(M[1]\oplus Auth,M[2],\cdots,M[i]),2\le i\le m$可知, 当$(N,A)$相同时, $Auth$值是相同的, 相同的明文前缀得到相同的密文前缀, 敌手A对OAE^e与of的区分优势与敌手D对底层在线密码OC询问加密部分q_e次, 总共询问σ个分组的OPRP-CPA攻击优势相同, 即$Adv_{HEH[G,E]}^{oprp-cpa}(D)$。

当$(N,A)\neq(N',A')$时, 对应得到的$Auth=Auth'$的概率为$1/2^n$。若$Auth\neq Auth'$, 则由在线密码的性质, OAE^e与of的区分优势也为$Adv_{HEH[G,E]}^{oprp-cpa}(D)$,

若$(N,A) \neq (N',A'), Auth = Auth'$, 则 OAE^e对相同的明文前缀仍然输出相同密文前缀, 敌手可以概率 1 将 OAE^e与 of 区分开来, 则q_e次加密询问敌手的优势为 $Adv^{oprp-cpa}_{HEH[G,E]}(D) + q_e^2/2^n$。

所以敌手 A 区分 $OAE^e[OC]$和 of 的优势为

$$\begin{aligned} Adv^{of}_{OAE^e[OC]}(A) &\leq Adv^{oprp-cpa}_{HEH[G,E]}(D) + q_e^2/2^n \\ &\leq Adv^{oprp-cca}_{HEH[G,E]}(D) + q_e^2/2^n \\ &\leq Adv^{prp-cca}_{E}(C) + \sigma^2/2^n + \sigma^2\delta \\ &\quad + q_e^2/2^n。 \end{aligned}$$

也即;

2) 然后, 我们讨论 $OAE^t\left[OC^{-1,(m+1)}\right]$与 of'的区分概率。用于验证完整性的校验字符串 $M[m+1] = OAE^t(N,A,C) = OC^{-1,(m+1)}(Auth \oplus C[1],C[2],\cdots,(C[m],C[m+1] \oplus 2^mL) \oplus 2^mL$。设敌手通过$q_e$次加密询问得到$(N_i,A_i,M_i,C_i), i = 1,2,\cdots,q_e$, 显然若存在 $i \neq j$ 使得 $C_i[m+1] = C_j[m+1]$, $C_i[m+1] \oplus 2^mL = C_j[m+1] \oplus 2^mL$, 底层在线密码在 $m+1$ 分组后得到相同的中间状态, 我们使用$(N_i,A_i,C_i||C')$与$(N_j,A_j,C_j||C')$询问 OAE^t, 其中 $C' \in \{0,1\}^{n*}$, 将得到相同的校验字符串, 从而使得 $OAE^t\left[OC^{-1,(m+1)}\right]$与 of'区分开来, 敌手的优势不超过 $\Pr(C_i[m+1] = C_j[m+1]) \leq q_e^2/2^n$。以下计算敌手使用新的三元组$(N^*,A^*,C^*)$来作区分的攻击成功优势。

若对于某个 $i \in \{1,2,\cdots,q_e\}$有$(N^*,A^*) = (N_i,A_i)$, 则对应的 $Auth^* = Auth_i$, 则必有某个 j 使得$C^*[j] \neq C_i[j]$, 因为 OAE 使用的 OC 具有 OPRP-CCA 安全性, 所以其解密部分 OC^{-1}与加密部分具有相同强度, 通过与上面 OAE^e和 of 类似情况的讨论, 我们知道 $OAE^t\left[OC^{-1,(m+1)}\right]$和 of'的区分概率由 $OC^{-1,(m+1)}$输出的随机性决定。

若对于所有的 $i = 1,2,\cdots,q_e$ 都有 $(N^*,A^*) \neq (N_i,A_i)$, 则只有当对应得到的 $Auth^* = Auth_i$且$C^* = C_i$时, $OAE^t\left[OC^{-1,(m+1)}\right]$和 of'才能区分开来, 其他情况下 $OAE^t\left[OC^{-1,(m+1)}\right]$的运行方式都与 of'一致, 此情况下敌手的区分优势为 $(N^*,A^*) \neq (N_i,A_i)$ 下 $Auth^* = Auth_i$的概率, 即 $1/2^n$, 所以q_d次解密询问两者的区分概率为$q_d^2/2^n$, 仍需要考虑 $OC^{-1,(m+1)}$输出的随机性。

又 $OC^{-1,(m+1)}$输出的随机性由其哈希层的 OAU 碰撞概率决定, 即δ, 所以q_d次加密询问两者的区分概率为$q_d^2\delta$。

所以敌手 B 区分 OAE^t 和 of' 的优势为 $Adv^{of'}_{OAE^t\left[OC^{-1,(m+1)}\right]}(B) \leq q_d^2\delta + (q_e^2 + q_d^2)/2^n$。

也即 $Adv^{int-ctxt}_{OAE}(B) \leq q_d^2\delta + (q_e^2 + q_d^2)/2^n$。

6.4 设计理念

1) 在线认证加密方案的设计有很多方式, 比如基于分组密码、基于可调分组密码、基于置换等, 几乎所有可用来设计认证加密方案的方法都可以经过改造用来设计在线认证加密方案。而使用在线密码来设计在线认证加密方案的优点是: 在结构上很简洁, 只添加了对关联数据和 Nonce 处理的部分、认证方式和一些简单运算; 由于底层用的了 OPRP-CCA 安全的在线密码, 认证部分是直接通过对明文加冗余信息(0^n)实现的, 这种方式比 COLM 等对明文分组异或值加密生成认证码的方式更简洁和节省资源; 同时对安全性证明来说, 也很容易证明由此构造的在线认证加密方案的安全性。

2) 对于关联数据和 Nonce 的处理, 我们采用有限域乘法逐分组来处理。这种处理方式与 HEH 结构中的哈希层相似的处理方式, 只是没有异或上2^iL, 因为 HEH 结构的哈希层需要消息的相同位置和不同位置的输出都应该是抗碰撞的, 而对关联数据和 Nonce 的处理只需要其生成的 $Auth$ 值是抗碰撞的即可, 由有限域乘法函数迭代得到的多项式函数为 AXU 函数(最后多做一次乘以密钥的运算, 否则只是 AU 函数), 与第一个消息分组异或后满足除公共最长前缀外消息分组的抗碰撞性。

3) 关联数据处理后得到的中间值 $Auth$ 是通过在在线密码处理的前后异或到消息的第一个分组处理中的。这样从加密过程来看, 关联数据和 Nonce 会对所有的密文的生成产生影响, 包括冗余信息分组产生的密文; 从解密验证过程来看, 关联数据和 Nonce 会对所有解密得到的明文产生影响, 包括冗余认证消息分组的再生成。这将关联数据和 Nonce 对认证加密方案安全性的影响达到了最大。

4) 对于在线认证加密方案 OAE 的认证部分, 我们采用在消息最后添加冗余的方式来验证其完整性。由于将冗余分组紧接在消息之后处理, 需要使用不同的方式处理冗余分组, 以将二者的处理区别开来, 否则询问$(N,A,M[1..m])$, 其中 $M[m] = 0^n$, 得到密文$(C[1..m+1])$, 我们用截取密文的方式伪造密文消息$(N,A,C[1..m])$, 可以通过解密验证。所以我们对冗余信息加掩码2^mL, 在生成密文前再次加上该掩码。实际上在冗余信息分组处理的哈希层中异或了$2^{m+1}L$, 则有$2^mL \oplus 2^{m+1}L = 3 \cdot 2^mL$, 因此将冗余消息分组的处理和消息的处理区别开来。

7 结论

通过分析目前文献中 Hash-ECB-Hash 结构的唯一实例化在线密码 POE, 我们扩展了经典泛哈希函数的概念到在线的形式, 即在线泛哈希函数, 包括 OAU 函数和 OAXU 函数。证明了当哈希层是 OAU 函数、底层分组密码是 CCA 安全时, HEH 结构是 CCA 安全的。我们还给出了 OAU 函数的具体构造, 包括使用均匀随机函数的 CFB 模式和 CBC 模式, 还有基于有限域上乘法函数的 MCFB 模式和使用输入输出异或链接方式的构造 XCH。HEH 结构提供了一般的方法来构造并行且 CCA 安全的在线密码, 由此我们继续用其来构造安全的在线认证加密方案。

由于在线密码 OC 的三层结构中密钥都假设为随机独立的, 还有使得哈希层成为 OAU 函数而异或上的 L, 共需要 4 个密钥, 认证加密方案 OAE 也有类似情况, 在处理关联数据和 Nonce 时需要额外的一个随机独立的密钥, 这在具体应用中会带来密钥管理的麻烦, 因此在具体实例化时, 可以由一个主密钥 K 派生而来, 比如 $L=E_K(0),K_1=E_K(1),K_2=E_K(2),K_3=E_K(3),K_4=E_K(4)$。

由于在线密码中 HEH 结构的 OAU 哈希层只能串行计算, 我们设计的在线密码和在线认证加密方案在运行时至少需要维护 $2n$ 长度的状态(n 为分组长度), 这在软件实现上对内存提高了要求, 如何改进设计使得状态长度小于 $2n$ 甚至降至 n 是我们未来工作需要考虑的重要方面。

致谢 本课题得到国家自然科学基金(No.61732021, No.61472415)和国家重点研发计划(No.2018YFA0704704, No.2018YFB0803801)资助。

参考文献

[1] Liu G, Wang P, Wei R, et al. Revisiting Construction of Online Cipher in Hash-ECB-Hash Structure[M]. Information Security and Cryptology. Cham: Springer International Publishing, 2021: 491-503.

[2] Bellare M, Boldyreva A, Knudsen L, et al. On-line ciphers and the hash-CBC constructions[J]. *Journal of Cryptology*, 2012, 25(4): 640-679.

[3] Fleischmann E, Forler C, Lucks S. McOE: A Family of Almost Foolproof On-Line Authenticated Encryption Schemes[C]. *International Workshop on Fast Software Encryption*. Berlin, Heidelberg: Springer, 2012: 196-215.

[4] Abed F, Fluhrer S, Forler C, et al. Pipelineable On-line Encryption[C]. *International Workshop on Fast Software Encryption*. Berlin, Heidelberg: Springer, 2015: 205-223.

[5] Andreeva E, Bogdanov A, Luykx A, et al. Parallelizable and Authenticated Online Ciphers[C]. *The 19th International Conference on Advances in Cryptology - ASIACRYPT 2013 - Volume 8269*, 2013: 424-443.

[6] Datta N, Nandi M. ELmD v2.0. *Submission to the CAESAR Competition*, https://competitions.cr.yp.to/round2/elmdv20.pdf, 2015.

[7] Andreeva E, Bogdanov A, Datta N, et al. COLM v1. *Submission to the CAESAR Competition*, https://competitions.cr.yp.to/round3/colm v1.pdf, 2016.

[8] Boldyreva A, Taesombut N. Online Encryption Schemes: New Security Notions and Constructions[C]. *Cryptographers' Track at the RSA Conference*. Berlin, Heidelberg: Springer, 2004: 1-14.

[9] Nandi M. Two New Efficient CCA-Secure Online Ciphers: MHCBC and MCBC[C]. *The 9th International Conference on Cryptology in India*: *Progress in Cryptology*, 2008: 350-362.

[10] Rogaway P, Zhang H B. Online Ciphers from Tweakable Blockciphers[C]. *Cryptographers' Track at the RSA Conference*. Berlin, Heidelberg: Springer, 2011: 237-249.

[11] Forler C, List E, Lucks S, et al. *Cryptography and Communications*, 2018, 10(1): 177-193.

[12] Jha A, Nandi M. On rate-1 and beyond-the-birthday bound secure online ciphers using tweakable block ciphers[J]. *Cryptography and Communications*, 2018, 10(5): 731-753.

[13] Bhaumik R, Nandi M. OleF: An Inverse-Free Online Cipher. an Online SPRP with an Optimal Inverse-Free Construction[J]. *IACR Transactions on Symmetric Cryptology*, 2017: 30-51.

[14] Datta N, Luykx A, Mennink B, et al. Understanding RUP Integrity of COLM[J]. *IACR Transactions on Symmetric Cryptology*, 2017: 143-161.

[15] Carter J L, Wegman M N. Universal classes of hash functions[J]. *Journal of Computer and System Sciences*, 1979, 18(2): 143-154.

[16] Wegman M N, Carter J L. New hash functions and their use in authentication and set equality[J]. *Journal of Computer and System Sciences*, 1981, 22(3): 265-279.

[17] Black J, Halevi S, Krawczyk H, et al. UMAC: Fast and Secure Message Authentication[M]. Advances in Cryptology — CRYPTO' 99. Berlin, Heidelberg: Springer Berlin Heidelberg, 1999: 216-233.

[18] Etzel M, Patel S, Ramzan Z. Square Hash: Fast Message Authentication *via* Optimized Universal Hash Functions[M]. Advances in Cryptology — CRYPTO' 99. Berlin, Heidelberg: Springer Berlin Heidelberg, 1999: 234-251.

[19] Halevi S, Krawczyk H. MMH: Software Message Authentication in the Gbit/Second Rates[M]. Fast Software Encryption. Berlin, Heidelberg: Springer Berlin Heidelberg, 1997: 172-189.

[20] Nandi M. Forging Attacks on Two Authenticated Encryption Schemes COBRA and POET[C]. *International Conference on the Theory and Application of Cryptology and Information Security*. Berlin, Heidelberg: Springer, 2014: 126-140.

[21] Bellare M, Rogaway P. The Security of Triple Encryption and a Framework for Code-Based Game-Playing Proofs[C]. *The 24th annual international conference on The Theory and Applications of Cryptographic Techniques*, 2006: 409-426.

[22] Wang P, Feng D G, Wu W L. HCTR: A Variable-Input-Length Enciphering Mode[M]. Information Security and Cryptology. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005: 175-188.

[23] Halevi S. Invertible Universal Hashing and the TET Encryption Mode[C]. Annual International Cryptology Conference. Berlin, Heidelberg: Springer, 2007: 412-429.

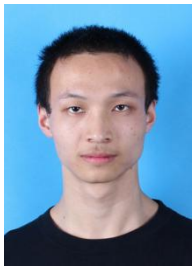
[24] Rogaway P. Efficient Instantiations of Tweakable Blockciphers and Refinements to Modes OCB and PMAC[C]. *International Conference on the Theory and Application of Cryptology and Information Security*. Berlin, Heidelberg: Springer, 2004: 16-31.

[25] McGrew D A, Viega J. The Security and Performance of the Galois/Counter Mode (GCM) of Operation[C]. *The 5th International Conference on Cryptology in India*, 2004: 343-355.

[26] Stinson D R. Universal hashing and authentication codes[J]. *Designs*, *Codes and Cryptography*, 1994, 4(3): 369-380.

[27] Shoup V. Sequences of games: A tool for taming complexity in security proofs[J]. *IACR Cryptol. ePrint Arch*, https://ia.cr/2004/332, 2004.

[28] Black J, Rogaway P. CBC MACs for arbitrary-length messages: The three-key constructions[J]. *Journal of Cryptology*, 2005, 18(2): 111-131.



刘刚 于 2015 年在湖南大学计算机科学与技术专业获得学士学位。现在中国科学院大学网络空间安全专业攻读博士学位。研究领域为对称密码算法的设计与分析。Email: liugang@iie.ac.cn

王鹏 于 2006 年在中国科学院大学通信与系统专业获得博士学位。现在中国科学院信息工程研究所, 副研究员。研究领域为对称密码方案的设计与分析。Email: wpeng@iie.ac.cn

魏荣 于 2019 年在中国科学院大学计算机技术专业获得硕士学位。现在北京卫星信息工程研究中心, 助理研究员。研究领域为对称密码算法的设计与分析。Email: weirong1127@163.com

叶顶锋 于 1996 年在中国科学院研究生院应用数学专业获得博士学位。现在中国科学院信息工程研究所, 研究员。研究领域为密码算法的设计与分析、理论密码学。Email: yedingfeng@iie.ac.cn