密码芯片基于聚类的模板攻击

吴震，杜之波，王敏，向春玲
（成都信息工程大学网络空间安全学院，四川 成都 610225）

摘 要：传统的模板攻击需要已知密钥建模等对实验设备完全控制的前置条件来实施攻击，该前置条件限制了模板攻击的应用场景，使模板攻击只能应用于可以控制密钥输入的设备。为了解决该问题，提出了基于聚类的模板攻击方法。该方法根据信息泄露模型的特征对聚类期望最大值（EM）算法进行改造，使改造后的聚类方法能够较为准确地拟合出泄露信息的概率模型，在未知密钥的情况下，即可确定信息泄露的位置。该方法通过建模进行模板匹配，消除了传统模板攻击对已知密钥建模等前置条件的依赖，从而扩大了模板攻击的应用范围。

关键词：侧信道攻击；模板攻击；聚类；EM 算法
中图分类号：TP309.1
文献标识码：A
doi: 10.11959/j.issn.1000−436x.2018130

Template attack of Crypto chip based on clustering

WU Zhen, DU Zhibo, WANG Min, XIANG Chunling
College of Information Security Engineering, Chengdu University of Information Technology, Chengdu 610225, China

Abstract: The known-key establishment template and others full control of experimental equipment preconditions are required to implement the traditional template attack. The preconditions restrict the application scenario of template attack. The template attack is only applied to the device that the key input can be controlled. In order to resolve the restrictive preconditions, a novel method of template attack based on clustering was proposed. The clustering EM algorithm was modified according to the characteristics of information leakage model in the method. The modified clustering methods accurately fitted the leaked information probability model in the case of unknown key, the location of information leakage could be determined. Then the attack established the templates in the location, and implemented template matching. The proposed method eliminates the dependence of traditional template attacks on per-conditions and expand the application scenario of template attack.

Key words: side channel attack, template attack, clustering, EM algorithm

1 引言

侧信道攻击是一种利用加密设备的能量泄露来攻击设备密钥的方法。Kocher 等最早提出 SPA（simple power analysis）[1]和 DPA（differential power analysis）[2]的能量攻击方法。这些攻击方法的原理是：假设加密的某个中间状态与泄露的能量具有相关关系，攻击者可以使用泄露的能量来检查、猜测密钥的正确性，从而达到攻击的目的。CPA（correlation power analysis）[3]是利用这种相关性进

收稿日期：2018−04−02；**修回日期**：2018−07−16
通信作者：杜之波，du139123456789@163.com
基金项目：国家科技重大专项基金资助项目（No.2014ZX01032401）；国家高技术研究发展计划（“863”计划）基金资助项目（No.2012AA01A403）；“十三五”国家密码发展基金资助项目（No.MMJJ20180244）；四川省科技支撑计划项目基金资助（No.2017GZ0313）；四川省教育厅重点科研基金资助项目（No.17ZB0082）
Foundation Items: The National Science and Technology Major Project of China (No.2014ZX01032401), The National High Technology Research and Development Program of China (863 Program) (No.2012AA01A403), The “13th Five-Years” National Cryptogram Development Fund (No.MMJJ20180244), Sichuan Science and Technology Support Programmer (No. 2017GZ0313), Sichuan Provincial Education Department Key Scientific Research Projects (No.17ZB0082)

行攻击的最有效方法。SPA、DPA、CPA 以及基于这些方法提出的其他方法都是直接对加密设备进行攻击的，并不需要一个训练阶段。

模板攻击[4]是侧信道攻击技术的另一个分支。模板攻击分为 2 个阶段：建模和攻击。在建模阶段，模板攻击利用与被攻击设备相同的、攻击者可完全控制的实验设备，通过设置密钥和明文实施多次加密操作，收集操作的大量能耗数据，并据此建立、刻画被泄露信息的能耗概率模型。精确的能耗概率模型可以使攻击者仅使用少量的攻击能迹就能够恢复设备的密钥。但相对于 SAP、DPA 和 CPA，模板攻击需要攻击者具有对实验设备的完全控制能力，这限制了其应用范围。为了克服这一现状，文献[5]提出一种仅需 3 个密钥的半监督建模方法；文献[6-7]则更近一层地提出 EIS（equal image under different sub-key）的概念，即不论使用固定密钥随机明文，还是使用固定明文随机密钥，加密中间值的分布都是相同的。这样，攻击者只需要知道实验设备的密钥，而不需要改变其密钥，采用固定密钥随机明文的方法收集能迹，同样可以建立泄露信息的模板，进而成功地获取被攻击设备的密钥。

本文的方法则希望能更进一步地使模板攻击完全不需要实验设备，即在未知密钥的情况下，直接对被攻击设备的能耗建模，进而对其实施模板攻击。普遍模板攻击的建模阶段，攻击者需要根据已知信息对能迹进行分组，并针对分组能迹的特征分布建立相应的模板。从这些模板的功能和使用原理上看，它们就是一个分类器。模板攻击实质上就是机器学习中分类算法的应用。因此，很多论文提出了使用不同的分类算法实施模板攻击[8-10]。从机器学习的角度看，当拥有标签的训练数据时，分类算法是自然选择。而在不具备有标签数据时，可以采用聚类算法在一定程度上还原数据的本质类别。文献[11]提出在 DPA 中使用簇质量为标准来判断猜测密钥是否正确的方法，该方法在本质上还不是使用聚类算法建模。文献[12]提出另一种使用无监督学习的攻击方法：用线性拟合计算猜测密钥对应中间值（文中称为 feature）的能耗转换系数（文中称为权重）。文献[6]在其 Stochastic Model 中已提出这种方法，并且有更详细的关于计算方法的说明，从而得到一个猜测密钥在一个时刻上的拟合高斯分布，然后运用正确密钥在正确时刻上的预测误差最小的原则进行攻击。这种方法在本质上与文献[11]没有太大区别，都是使用猜测 key 对能迹进行分组和建模，区别在于用预测误差代替文献[11]中的簇质量。这些方法都不是利用聚类算法直接得到攻击模板的。文献[13]提出采用 *k*-means 聚类算法来区分指数加密中单执行中的位泄露。该方法需要实现在能迹上人工确定每位的泄露区域，且仅具有二类识别的能力。同时，由于聚类模型的不准确，其攻击效果需要结合多个侧信道测量进行同时攻击。文献[14]提出使用高斯混合分布的 EM 聚类算法得到对加掩算法中掩码的攻击模板。其原理是通过聚类计算出每个可能的(x,k)对的不同掩码产生的混合高斯分布，其中，x 为明文，k 为密钥。如果掩码为 d 位，则每个(x,k)对的混合高斯分布包含 2^d 个成员分布。由于模型多、模型参数多，这种方法需要大量的能迹和迭代时间，因此在实际上的可行性不高。事实上，文献[14]的实验中仅对掩码的单个位进行攻击，因此其聚类算法也是二分类的。文中也提出了针对高斯混合分布的 EM 算法的几种变体，例如固定先验概率、单位协方差矩阵、共享协方差矩阵等，但并未给出采用这些变体理由。此外，上述方法均没有考虑 EM 算法的一个根本性问题：EM 算法得到的是局部最优，其聚类结果与初始参数设置密切相关。上述文献均采用随机设置初始参数的方法，无法保证聚类的有效性和稳定性。

本文对高斯混合分布的 EM 算法在模板攻击中的应用方法进行了详细的讨论。根据理论上和实际中的经验，提出了对 EM 算法的几种针对性的约束，以便真正提高聚类的质量。这些方法运用在 9 个簇的汉明重量模型的聚类中，可以相当准确地还原出真实成员分布。针对 EM 算法得到局部最优的问题，本文给出了切实可行的算法初始参数设置方法。此外，上述文献并未讨论如何在密钥或掩码未知的情况下，找到确切的泄露位置，针对泄露位置的问题，要么采用类似暴力攻击的方法，要么直接假设已知泄露位置。对此，本文提出了一种在未知密钥的情况下，准确找到信息泄露位置的方法。

2 基于聚类的模板攻击模型

2.1 基本攻击模型

假设攻击者没有标准模板攻击所需要的实验设备，即不能设置加密设备的密钥，只能设置任意随机明文使用加密设备进行加密操作，并收集设备能耗。在这种场景下，由于密钥未知，无法计算目

标操作的中间值，不能使用有监督的学习方法获得模板。

在这种无法预知中间值、无法明确分类的情况下，则可以采用无监督的学习发现数据外在的划分或内在的模型。无监督的学习算法可以分为 2 个类别：簇合算法和基于潜变量模型的学习。簇合算法根据数据本身的特征（相似性、距离等），将特征相似或距离相近的数据聚集在一起形成簇。常见的簇合算法包括 *k*-means、DBSCAN、高斯混合模型等。簇合算法的质量取决于数据特征的显著程度。基于潜变量模型的学习算法认为数据中可观察的变量（显变量）是由内在的、不可观察的隐变量决定的。

在侧信道攻击中，广为接受的能耗模型是汉明重量或汉明距离模型。该模型认为，加密操作中间值的汉明重量或汉明距离与相应的能耗呈线性关系，且叠加高斯噪声。这样，各汉明重量或汉明距离的能耗是一个高斯分布，所有汉明重量或汉明距离的能耗则是一个混合高斯分布。这种成员分布就是模板攻击中所需要的模板。针对混合高斯分布的无监督学习算法是 EM 算法。如果攻击者可以使用 EM 算法还原出真实的内在分布，就可以以此为模板进行模板攻击。这里的关键问题是，对于高噪声、低信噪比的能耗数据，EM 算法能够在多大程度上还原数据的真实分布。

2.2　混合高斯模型及其 EM 聚类算法

2.2.1　汉明重量能耗模型与混合高斯分布

根据汉明重量模型，设备操作的能耗 w 包括操作能耗 w_i、数据相关能耗 w_d 和高斯白噪声 n。其中，数据相关能耗 w_d 与操作数据 x 的汉明重量（HW，Hamming weight）h 线性相关[15]。

$$w = w_i + w_d + n \tag{1}$$

$$w_d = \alpha h \tag{2}$$

$$n \sim \mathrm{N}(0,\sigma^2) \tag{3}$$

由此，对给定的汉明重量为 h、操作能耗为 w 的高斯分布为

$$f_h(w|\theta) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{(w-\mu_h)^2}{2\pi{\sigma_h}^2}\right) \tag{4}$$

其中，θ 是高斯分布的参数，θ=$\{\mu_h,\sigma_h\}$。μ_h 为汉明重量 h 的均值能耗，${\sigma_h}^2$ 为其方差。设中间值有 b 位，则能迹集中包含 h 的 $b+1$ 种取值，其能耗为混合高斯分布，即

$$f(w|\theta) = \sum_{h=0}^{b}\tau_h\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{(w-\mu_h)^2}{2\pi{\sigma_h}^2}\right) \tag{5}$$

其中，θ 是混合高斯分布的参数，θ=$\{\tau_h,\mu_h,\sigma_h\}$。$\tau_h$ 为汉明重量 h 所对应的成员高斯分布的先验概率。

如果一个操作存在 d 个能耗泄露位置，则这些泄露位置上的能耗向量 $\boldsymbol{w}=[w_1,\cdots,w_d]^{\mathrm{T}}$ 符合混合多元高斯分布，即

$$f(\boldsymbol{w}|\theta) = \sum_{h=0}^{b}\tau_h\frac{1}{\sqrt{(2\pi)^d|\boldsymbol{\Sigma}_h|}}\exp\left(-\frac{1}{2}(\boldsymbol{w}-\boldsymbol{u}_h)^{\mathrm{T}}\boldsymbol{\Sigma}_h^{-1}(\boldsymbol{w}-\boldsymbol{u}_h)\right) \tag{6}$$

其中，$\theta=\{\tau_h,\boldsymbol{\mu}_h,\boldsymbol{\Sigma}_h\}$ 是多元混合高斯分布的参数。τ_h、$\boldsymbol{\mu}_h$、$\boldsymbol{\Sigma}_h$ 分别为汉明重量为 h 所对应的成员高斯分布的先验概率、均值能耗向量和协方差矩阵。

以 8 位 SBOX 输出为例，其汉明重量为 0~8。图 1 给出了实际能耗数据的混合一元高斯分布。

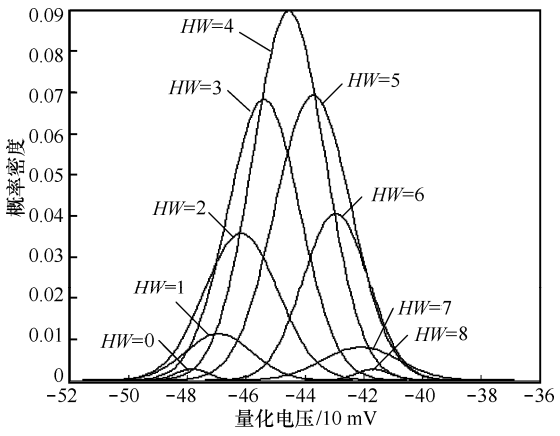


图 1　实际能耗数据的混合一元高斯分布

从图 1 可以看出，实际分布与汉明能耗模型的预测是非常相符的。

2.2.2　使用 EM 算法估算能耗的混合高斯分布

在密钥未知的前提下，需要使用聚类算法还原出数据本身的真实混合高斯分布。EM 算法的训练目标是：在给定数据集的条件下，模型参数的对数似然率的数学期望最大。给定能耗数据 $\boldsymbol{w}$，成员分布 k 的参数 θ_k 的对数似然率为

$$\begin{aligned}\ell_h(\theta_h;\boldsymbol{w}) &= \ln\tau_h f(\boldsymbol{w}|\theta_h)\\ &= \ln\tau_h + \ln\frac{1}{\sqrt{(2\pi)^d|\boldsymbol{\Sigma}_h|}} - \frac{1}{2}(\boldsymbol{w}-\boldsymbol{u}_h)^{\mathrm{T}}\boldsymbol{\Sigma}_h^{-1}(\boldsymbol{w}-\boldsymbol{u}_h)\end{aligned} \tag{7}$$

给定能耗数据 $\boldsymbol{W}=\{\boldsymbol{w}_1,\cdots,\boldsymbol{w}_n\}$，混合高斯分布的参数 $\theta=\{\tau_h,\boldsymbol{\mu}_h,\boldsymbol{\Sigma}_h\},h=0,\cdots,b$ 的对数似然率的数学期望为

$$\begin{aligned}\boldsymbol{Q}(\theta\left|\theta^{(t)}\right.)&=\mathrm{E}[\ell(\theta;\boldsymbol{W})]\\&=\sum_{i=1}^{n}\sum_{h=1}^{b}\boldsymbol{T}_{i,h}^{(t)}\ell_h(\theta_h;\boldsymbol{w}_i)\end{aligned}\tag{8}$$

其中，$\boldsymbol{T}_{h,i}^{(t)}$ 是第 t 轮迭代时的数据 $\boldsymbol{w}_i$ 对分布 h 的成员率，为

$$\boldsymbol{T}_{h,i}^{(t)}=\frac{\tau_k^{(t)}f(\boldsymbol{w}_i;\boldsymbol{\mu}_h^{(t)},\boldsymbol{\Sigma}_h^{(t)})}{\sum_{k=1}^{b}f(\boldsymbol{w}_i;\boldsymbol{\mu}_h^{(t)},\boldsymbol{\Sigma}_h^{(t)})}\tag{9}$$

若式(9)对各参数的偏导等于0，就可以得到EM算法迭代中参数的更新方法，为

$$\tau_h^{(t+1)}=\frac{1}{n}\sum_{i=1}^{n}\boldsymbol{T}_{h,i}^{(t)}\tag{10}$$

$$\boldsymbol{\mu}_h^{(t+1)}=\frac{\sum_{i=1}^{n}\boldsymbol{T}_{h,i}^{(t)}\boldsymbol{w}_i}{\sum_{i=1}^{n}\boldsymbol{T}_{h,i}^{(t)}}\tag{11}$$

$$\boldsymbol{\Sigma}_h^{(t+1)}=\frac{\sum_{i=1}^{n}\boldsymbol{T}_{h,i}^{(t)}(\boldsymbol{w}_i-\boldsymbol{\mu}_i^{(t+1)})(\boldsymbol{w}_i-\boldsymbol{\mu}_i^{(t+1)})^{\mathrm{T}}}{\sum_{i=1}^{n}\boldsymbol{T}_{h,i}^{(t)}}\tag{12}$$

EM 算法通过反复迭代来优化对数似然率的数学期望，直到对数似然率的变化小于一个阈值为止。每次迭代分为 E-Step 和 M-Step。

E-Step：根据当前的模型参数，使用式(9)计算成员率矩阵。

M-Step：根据当前的成员率，使用式(10)~式(12)更新模型参数。

然而，对于低信噪比的能耗数据，标准 EM 算法的聚类并不能恢复真实的成员分布。图 2 是在混合二元高斯模型下，EM 算法得到的成员分布与真实成员分布的对比情况。其中，椭圆是各成员分布的 95%最大概率密度的等高线。黑色为真实数据的成员分布，灰色是 EM 算法得到的混合高斯分布的聚类成员分布（下同）。可以看出，由于各汉明重量的能耗数据高度混淆在一起，标准 EM 算法并不能准确恢复真实成员分布的中心位置和方差。

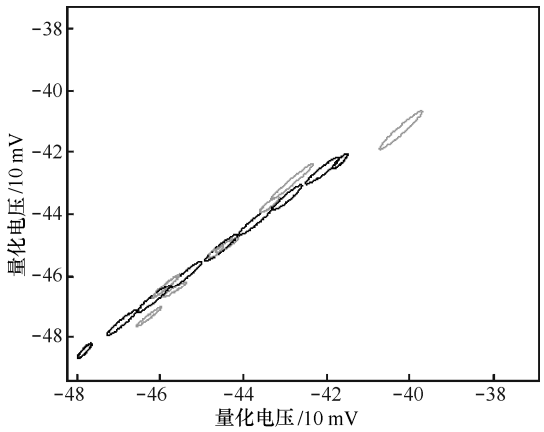


图 2 真实成员分布（黑色）与 EM 聚类成员分布（灰色）比较

2.3 汉明能耗模型及其 EM 聚类算法

汉明能耗模型是一种特殊的高斯混合分布模型，有以下特征。

1) 成员分布的先验概率固定

对于本文设定的攻击场景：密钥固定和采用随机明文，目标操作的中间值是均匀分布的。这样，中间值的汉明重量呈二项分布。假设中间值为 b 位，汉明重量为 h 时对应的成员分布的先验概率为

$$\tau_h=\frac{1}{2^b}\binom{b}{h}\tag{13}$$

2) 相邻成员分布中心的距离相等

在汉明能耗模型中，由于均值能耗与汉明重量呈线性关系，即

$$\mu_h=\alpha h\tag{14}$$

则任意 2 个相邻的汉明重量的均值能耗之差为

$$\Delta\mu=\alpha(h_i-h_{i+1})=\alpha\tag{15}$$

据此，汉明重量为 h 的均值能耗为

$$\mu_h=\alpha m_h+\bar{\mu},h=0,\cdots,b\tag{16}$$

其中，$\bar{\mu}$ 是全部成员分布的均值能耗，即整个数据的均值能耗，为

$$\bar{\mu}=\frac{1}{n}\sum_{i=1}^{n}w_i\tag{17}$$

m_h 是汉明重量 h 的成员分布的均值能耗相对于整体均值能耗的偏移系数，为

$$m_h\in\left\{-\frac{b}{2},\cdots,0,\cdots,\frac{b}{2}\right\}\tag{18}$$

在特征 1)和 2)成立的情况下，高斯多元混合模型的参数 θ=$\{\alpha,\boldsymbol{\Sigma}_h\}$。相应地，EM 算法 E-Step 的参

数更新方法变为

$$\alpha^{(t+1)}=\frac{\sum_{i=1}^{n}\sum_{h=0}^{b}(\boldsymbol{w}_i-\bar{\boldsymbol{\mu}})m_h\boldsymbol{T}_{h,i}^{(t)}}{\sum_{i=1}^{n}\sum_{h=0}^{b}\boldsymbol{T}_{h,i}^{(t)}} \tag{19}$$

$$\boldsymbol{\mu}_h^{(t+1)}=\bar{\boldsymbol{\mu}}+m_h\alpha^{(t+1)} \tag{20}$$

$$\boldsymbol{\Sigma}_h^{(t+1)}=\frac{\sum_{i=1}^{n}\boldsymbol{T}_{h,i}^{(t)}(\boldsymbol{w}_i-\boldsymbol{\mu}_i^{(t+1)})(\boldsymbol{w}_i-\boldsymbol{\mu}_i^{(t+1)})^{\mathrm{T}}}{\sum_{i=1}^{n}\boldsymbol{T}_{h,i}^{(t)}} \tag{21}$$

其中，协方差矩阵的更新方式并未发生变化。

汉明能耗模型还有一个弱特征，即各成员分布的协方差矩阵相同或接近。这是因为，与同一个操作相关的噪声主要是由操作指令的硬件实现决定的，而与操作数的关系不大。假设该特征成立，则各成员的协方差矩阵应该是相同的。共享协方差矩阵的更新式为

$$\boldsymbol{\Sigma}_h^{(t+1)}=\frac{\sum_{i=1}^{n}\sum_{h=0}^{b}\boldsymbol{T}_{h,i}^{(t)}(\boldsymbol{w}_i-\boldsymbol{\mu}_i^{(t+1)})(\boldsymbol{w}_i-\boldsymbol{\mu}_i^{(t+1)})^{\mathrm{T}}}{\sum_{i=1}^{n}\sum_{h=0}^{b}\boldsymbol{T}_{h,i}^{(t)}} \tag{22}$$

图 3 和图 4 为采用上述特征改造 EM 算法后，数据的聚类成员分布与真实成员分布的对比情况。从图 3 可以看出，聚类得到的成员分布与真实成员分布仍然有一定差异。不仅是各成员分布中心与真实中心仍有较大的差异，成员分布的方差与真实方差相比也有较大差异。但总体来看，其比采用标准 EM 算法得到的结果要好得多。而图 4 是采用共享协方差矩阵得到的结果。其特点是：共享协方差矩阵的 EM 算法趋于将成员分布重叠在整体能耗均值中心，这一点可以从理论上得到证明（略）。因此，共享协方差矩阵在这里无法得到应用。

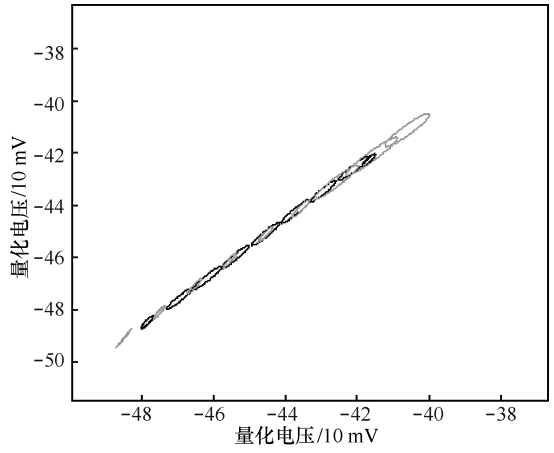


图 3　非共享协方差矩阵时，聚类成员分布与真实成员分布的比较

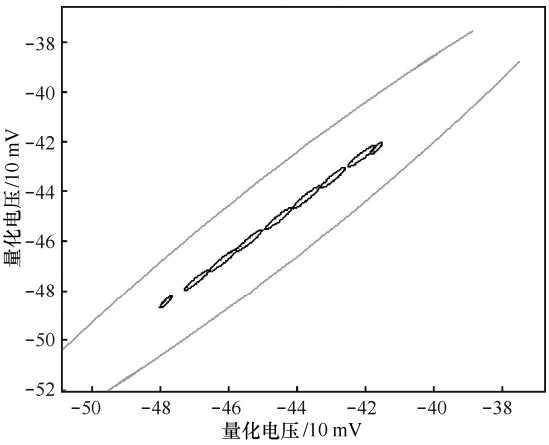


图 4　共享协方差矩阵时，聚类成员分布与真实成员分布

3　EM 算法改造

由于能耗数据的低信噪比的特点，标准的 EM 算法的聚类结果与真实数据分布相差巨大。而模板攻击所依赖的是对数据噪声的准确了解和表达。如果不对算法进行适应性改造，使用聚类算法进行模板攻击是不可行的。汉明重量模型提供了一些特征，可供对混合高斯模型以及相应的 EM 算法进行改造。

3.1　对聚类的优化

上述根据汉明能耗模型特征对 EM 算法的改造仍然无法得到所需的聚类质量。本节根据其他一些隐藏的特征，为 EM 算法附加一些额外约束，以期提高聚类质量。

3.1.1　成员分布的相关系数约束（pdc）

传统模板攻击采用 SOD、SOSD、SOST 等方法，根据目标中间值对训练能迹进行分组，通过计算组间差找到与目标操作泄露有关的能耗样本位置，该位置称为兴趣点（POI），用能耗向量 $\boldsymbol{w}$ 的各维表示能迹上各兴趣点位置上的能耗。假设任意 2 个给定 POI 上的能耗的相关性来源于电路的物理相关性（例如电容充放电的不同阶段），则不同中间值对应的能耗向量分组内，POI 位置上能耗的相关性应该是相同的，即各成员分布的相关系数矩阵与整体数据的相关系数矩阵相等。设能耗向量的任意二维分别为 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$，则二维之间的整体相关系数为

$$\rho_{xy}=\frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\bar{x})^2\sum_{i=1}^{n}(y_i-\bar{y})^2}} \tag{23}$$

成员分布的协方差矩阵的非对角元素为

$$\sigma_{xy}=\rho_{xy}\sigma_x\sigma_y \tag{24}$$

相应地修改 EM 算法的 M-Step：首先按对角协方差矩阵更新对角元素为

$$\boldsymbol{\Sigma}_h^{(t+1)}=\frac{\sum_{i=1}^{n}((\boldsymbol{w}_i-\boldsymbol{\mu}_i^{(t+1)})^{\mathrm{T}})^2\boldsymbol{T}_{h,i}^{(t)}}{\sum_{i=1}^{n}\boldsymbol{T}_{h,i}^{(t)}} \tag{25}$$

然后根据式(23)计算协方差矩阵的非对角元素。

图 5 显示了采用相同相关系数后，聚类得到的成员分布与真实成员分布的对比情况。与第 2 节的聚类结果相比，采用相同的相关系数后，成员分布的中心和方差与真实分布更加接近。

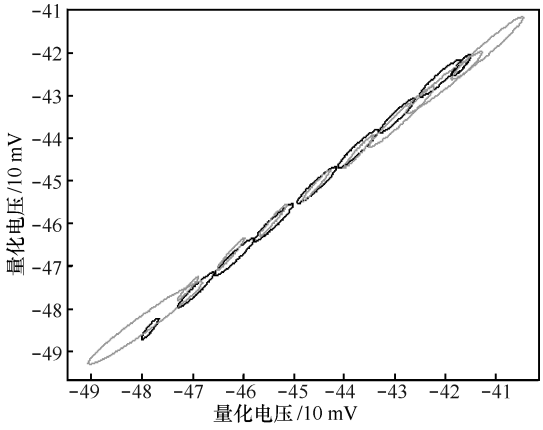


图 5 pdc 条件下聚类成员分布与真实成员分布对比

从图 5 可以看出，采用 pdc 相关系数约束后，成员分布中心和方差与真实分布较为接近。

3.1.2 成员分布的形状约束（pds）

根据真实分布的特点可以看出，所有成员分布等高线的“形状”大致相似。这并不是偶然的。“形状相似”实际上反映了不同汉明重量下，能耗的概率密度分布是相似的。不同的汉明重量可能影响方差的大小，但不会影响各维数据方差的比例关系和相关性。因此，成员分布形状相似的假设比共享协方差矩阵的假设稍弱。

从几何上看，等高线形状相似包括 2 个因素：等高线椭圆的方向一致，且椭圆的长短轴比例相等。首先推导二元高斯分布等高线的方向，然后扩展到多元高斯分布。

二元高斯分布等高线为椭圆。椭圆轴的方向就是二元高斯分布等高线的方向。设分布中心 $\boldsymbol{\mu}=0$，多元高斯分布为

$$p(\boldsymbol{w})=\frac{1}{2\pi|\boldsymbol{\Sigma}|}\exp\left(-\frac{1}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}\boldsymbol{w}\right) \tag{26}$$

设 $p(\boldsymbol{w})=p_0$，方程两端取对数，等高线方程为

$$-\frac{1}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}\boldsymbol{w}=\ln(2\pi|\boldsymbol{\Sigma}|\,p_0) \tag{27}$$

对二元高斯分布，设 $\boldsymbol{w}=[x,y]^{\mathrm{T}}$，有

$$\boldsymbol{\Sigma}^{-1}=\begin{bmatrix}\dfrac{\sigma_y^2}{\sigma_x^2\sigma_y^2-\sigma_{xy}^2} & -\dfrac{\sigma_{xy}}{\sigma_x^2\sigma_y^2-\sigma_{xy}^2}\\ -\dfrac{\sigma_{xy}}{\sigma_x^2\sigma_y^2-\sigma_{xy}^2} & \dfrac{\sigma_x^2}{\sigma_x^2\sigma_y^2-\sigma_{xy}^2}\end{bmatrix}\overset{\text{set}}{=}\begin{bmatrix}\alpha & -\gamma\\ -\gamma & \beta\end{bmatrix} \tag{28}$$

将式(28)代入式(27)，且设式(27)右端等于 1，则有

$$\alpha x^2-2\gamma xy+\beta y^2=1 \tag{29}$$

当 $\sigma_{xy}=0$，即 x 和 y 无关时，$\alpha=\dfrac{1}{\sigma_x^2}$，$\beta=\dfrac{1}{\sigma_y^2}$，$\gamma=0$，方程(28)转换为

$$\frac{x^2}{\sigma_x^2}+\frac{y^2}{\sigma_y^2}=1 \tag{30}$$

式(30)是标准椭圆方程。即在 $\sigma_{xy}=0$ 时，二元高斯分布的等高线是一个正椭圆。对于这里特定概率密度的等高线，椭圆 2 个轴的长度分布为 σ_x 和 σ_y。

但当 $\sigma_{xy}\neq 0$ 时，等高线是一个旋转一定角度的椭圆。设等高线椭圆逆时针方向旋转角度为 θ，则旋转方程为

$$\begin{bmatrix}x'\\ y'\end{bmatrix}=\begin{bmatrix}\cos\theta & -\sin\theta\\ -\sin\theta & \cos\theta\end{bmatrix}\begin{bmatrix}x\\ y\end{bmatrix} \tag{31}$$

式(30)旋转后得

$$\left(\frac{\cos^2\theta}{\sigma_x'^2}+\frac{\sin^2\theta}{\sigma_y'^2}\right)x'^2+\left(\frac{\sin 2\theta}{\sigma_x'^2}-\frac{\sin 2\theta}{\sigma_y'^2}\right)x'y'+\left(\frac{\sin^2\theta}{\sigma_x'^2}+\frac{\cos^2\theta}{\sigma_y'^2}\right)y'^2=1 \tag{32}$$

由于旋转后的σ_x和σ_y不再是二元高斯分布在x轴和y轴上的方差，因此式(32)中使用σ'_x和σ'_y代替σ_x和σ_y。

比较式(30)和式(28)，有以下方程组成立。

$$\begin{cases}\dfrac{\sigma_y^2}{\sigma_x^2\sigma_y^2-\sigma_{xy}^2}=\dfrac{\cos^2\theta}{\sigma_x'^2}+\dfrac{\sin^2\theta}{\sigma_y'^2}\\ \dfrac{\sigma_x^2}{\sigma_x^2\sigma_y^2-\sigma_{xy}^2}=\dfrac{\sin^2\theta}{\sigma_x'^2}+\dfrac{\cos^2\theta}{\sigma_y'^2}\\ \dfrac{\sigma_{xy}}{\sigma_x^2\sigma_y^2-\sigma_{xy}^2}=\dfrac{1}{2}\left(\dfrac{\sin 2\theta}{\sigma_x'^2}-\dfrac{\sin 2\theta}{\sigma_y'^2}\right)\end{cases} \tag{33}$$

解式(32)可得

$$\sin 2\theta=\sqrt{\frac{4\sigma_{xy}^2}{4\sigma_{xy}^2-4\sigma_x^2\sigma_y^2+(\sigma_x^2+\sigma_y^2)^2}}$$

$$\begin{aligned}\sigma_x'&=\frac{2(\sigma_x\sigma_y-\sigma_{xy}^2)}{\sigma_x^2+\sigma_y^2-\dfrac{2\sigma_{xy}}{\sin 2\theta}}\\ \sigma_y'&=\frac{2(\sigma_x\sigma_y-\sigma_{xy}^2)}{\sigma_x^2+\sigma_y^2+\dfrac{2\sigma_{xy}}{\sin 2\theta}}\end{aligned} \tag{34}$$

其中，σ_x'和σ_y'是数据在椭圆 2 个轴上的方差。使用式(34)，可以根据二元高斯分布的协方差矩阵，计算出其等高线椭圆的方向以及在椭圆2个轴上的方差（即数据反向旋转θ后，在x轴和y轴上的方差）。

假设二元高斯分布方向θ已知，则可以根据 x和y轴上的方差，计算协方差σ_{xy}。将式(34)代入方程组式(32)，求解后可得

$$\sigma_{xy}=\frac{(\sigma_x^2-\sigma_y^2)\sin 2\theta}{2(\cos^2\theta-\sin^2\theta)} \tag{35}$$

推理　当不同成员分布的σ_x和σ_y的比例相等，且等高线方向相同时，成员分布等高线形状相似。

证明　设$\dfrac{\sigma_x}{\sigma_y}=r$，而$\sigma_{xy}=\rho_{xy}\sigma_x\sigma_y$，由式(34)可得

$$\begin{aligned}\frac{\sigma_x'}{\sigma_y'}&=\frac{\sigma_x^2+\sigma_y^2+\dfrac{2\rho_{xy}\sigma_x\sigma_y}{\sin 2\theta}}{\sigma_x^2+\sigma_y^2+\dfrac{2\rho_{xy}\sigma_x\sigma_y}{\sin 2\theta}}\\ &=\frac{r^2+1+\dfrac{2r\rho_{xy}}{\sin 2\theta}}{r^2+1-\dfrac{2r\rho_{xy}}{\sin 2\theta}}\end{aligned} \tag{36}$$

式(36)表明，等高线椭圆 2 个轴长的比例由r、ρ_{xy}和θ决定。如果σ_{xy}按式(35)计算，则

$$\begin{aligned}\rho_{xy}&=\frac{\sigma_{xy}}{\sigma_x\sigma_y}\\ &=\frac{(\sigma_x^2-\sigma_y^2)\sin 2\theta}{2(\cos^2\theta-\sin^2\theta)}\cdot\frac{1}{\sigma_x\sigma_y}\\ &=f(\theta)\frac{r-1}{r}\end{aligned} \tag{37}$$

式(37)表明，当r和θ确定时，ρ_{xy}是确定的。因此可证明二元高斯分布等高线椭圆的形状仅由$r=\dfrac{\sigma_x}{\sigma_y}$和$\theta$决定。

根据这个结论，在 EM 算法中的 M-Step 中，只要保证所有成员分布的$\dfrac{\sigma_x}{\sigma_y}$相等，且$\sigma_{xy}$按式(35)计算，就能够保证它们的等高线形状相似。

对于多元高斯分布，对其中任意二维使用式(34)，可计算等高线在该二维子平面上的旋转角度。然后利用式(35)，根据成员分布在该二维上的方差，计算子平面上的协方差。

在实际应用中，假设成员分布方向与数据的整体高斯分布方向一致，则可以根据整体数据的协方差矩阵计算成员分布等高线的角度θ。在 EM 算法的 M-Step 中，成员分布协方差矩阵首先按对角矩阵更新，即首先计算各成员分布的σ_x和σ_y，再用中间汉明重量（$h=\dfrac{b}{2}$）的成员分布的$\dfrac{\sigma_x}{\sigma_y}$为标准，调整其余成员分布的$\dfrac{\sigma_x}{\sigma_y}$。然后根据式(35)更新成员分布的$\sigma_{xy}$。

根据这种方法，聚类的结果如图 6 所示。除汉明重量为 0 和 8 的成员分布与真实分布有偏差外，其余成员分布的中心和方差几乎完全与真实分布相同，非常好地还原了真实成员分布。汉明重量为 0 和 8 的拟合成员分布与真实分布的偏差主要来源于真实分布自身的偏差：在固定密钥、随机明文的训练能迹中，汉明重量为 0 和 8 的能迹数与总训练能迹数的比例近似等于其理想先验概率，即 0.39%。实验中使用 8 000 条能迹，而汉明重量为 0 和 8 的能迹数仅为 31 条左右。由于能迹不足，造成了其真实分布的统计偏差。

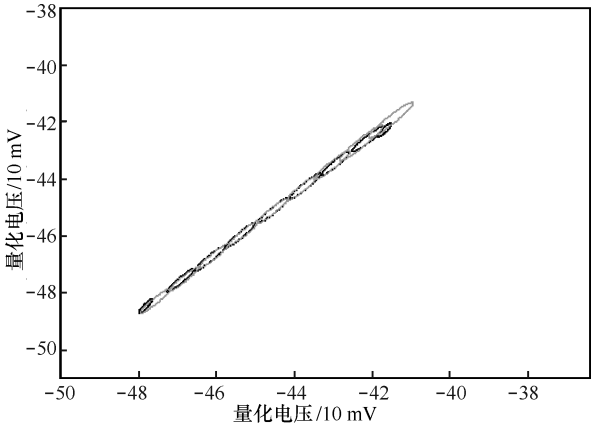


图 6 psd 条件下成员分布与真实成员分布对比

从图 6 可以看出，使用成员分布等高线形状相似约束后，EM 算法可以很好地恢复真实分布。

3.2 聚类的初始参数设置

3.2.1 未知密钥时发现信息泄露的位置

由于能迹上的样本数非常多，大多数样本与目标操作的信息泄露无关。直接使用能迹的全部样本作为能迹向量的元素，在效率上不可行，聚类的结果也不可能作为有效的模板。在本文设置的攻击场景下，设备密钥位置，无法计算真实的目标中间值，因此无法使用这些方法得到 POI。

在密钥固定的情况下，可以使用基于明文或线性变换后的明文（根据加密算法不同而有所不同）对能迹分组，通过计算组间差发现 SBOX 输入或输出的泄露位置。例如，SM4 算法，第一轮 SBOX 输入为 $X_1 \oplus X_2 \oplus X_3 \oplus RK$ 。在 RK 固定的情况下，如果多个能迹的 $Y = X_1 \oplus X_2 \oplus X_3$ 的值相同，则它们的 SBOX 输入相同。而 SBOX 变换的输入和输出是一一对应的，因此 SBOX 输出也相同。这样，如果使用 Y 对能迹分组，分组内 SBOX 输入和输出值（或其汉明重量）是相同的，分组均值的组间差同样可以反映 SBOX 输入或输出的泄露位置。

3.2.2 根据设备信噪比设置聚类初始参数

由于 EM 算法得到的是局部最优解，算法初始参数的设置非常关键。根据混合分布的整体方差与成员方差的关系以及理想分布中关于成员分布中心均匀线性分布和方差相等的假设，可以根据 POI 上的信噪比估算初始参数。

混合分布与其成员分布的方差关系为

$$\sigma^2 = \sum_{i=1}^{K} \tau_i (\mu_i^2 + \sigma_i^2) - \mu^2 \tag{38}$$

其中，σ_i 和 μ_i 为成员分布的方差和中心，σ 和 μ 为混合分布的方差和中心。

根据理想分布的假设，有

$$\begin{cases} \tau_i = \dfrac{1}{2^K}\binom{K}{i} \\ \mu_i = \alpha m_i + \mu, m_i = -\dfrac{K}{2}, \cdots, 0, \cdots, \dfrac{K}{2} \\ \sigma_i^2 = \sigma_0^2 \end{cases} \tag{39}$$

代入式(38)可得

$$\sigma_0^2 = \sigma^2 - 2\alpha^2 \tag{40}$$

根据能耗上信息泄露的特点，将信噪比定义为

$$SNR = \frac{\alpha}{\sigma} \tag{41}$$

代入式（40）后得到

$$\begin{aligned} \sigma_0^2 &= \sigma^2 \left(1 - 2SNR^2\right) \\ \alpha &= \sqrt{\frac{\sigma^2 - \sigma_0^2}{2}} \end{aligned} \tag{42}$$

使用式(42)，并根据全局方差和信噪比，可以估算出成员分布的初始参数。其中，混合分布的全局方差 σ^2 可以根据数据直接计算得到。信噪比则可以根据明文能迹分组中心进行估算。设明文分组中心为 $\mu_j, j = 1, \cdots, 256$ ，则信噪比估算为

$$\widehat{SNR} = \frac{\max(\mu_j) - \min(\mu_j)}{K - 1} \tag{43}$$

4 实验分析

4.1 实验设置

针对某密码芯片的 SM4 算法软实现进行多种方式的攻击，共收集能迹 10 000 条，其中，8 000 条用于训练，2 000 条用于攻击测试。芯片主频为 2.8 MHz，示波器采样频率为 100 MHz。采样能迹进行了低通滤波和对齐处理。

4.2 确定信息泄露位置

根据上文所述方法，确定被攻击 SM4 芯片的 POI，图 7 是根据 SOST 确定的第一个 SBOX 输入和输出的 POI，图中 2 个高尖峰对应的位置即是 SBOX 输入和输出的 POI，SBOX 输入的位置为 509，SBOX 输出的位置为 554，根据分组内 SBOX

输入和输出值以及输入和输出汉明重量确定的 POI 如表 1 所示。

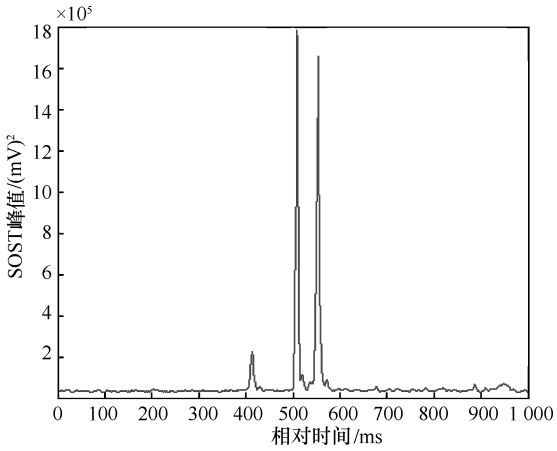


图 7　能耗迹的 SOST 峰值

表 1　　POI 点

SBOX 输入值	SBOX 输出值	SBOX 输入 *HW*	SBOX 输出 *HW*
509	554	509	554

4.3　与标准模板攻击的对比实验

实验中分别采用以下方法获得模板，以便进行对比。

1) 标准的模板攻击。即使用已知的固定密钥对训练能迹集进行分组，从而得到模板，该方法用 STD 表示。

2) 采用 EM 算法对训练能迹进行聚类得到的模板，该方法用 EM 表示。

3) 采用根据汉明能耗模型改造的 EM 算法聚类得到的模板，该方法用 EMPD 表示。

4) 在汉明能耗模型的基础上，采用增加相关性约束改造 EM 算法后聚类得到的模板，该方法用 EMPDC 表示。

5) 在汉明能耗模型基础上，采用增加“形状相似”约束改造 EM 算法后聚类得到的模板，该方法用 EMPDS 表示。

攻击方法 2)~5)选择 POI 的方法见 3.2 节。实验中统一选择 2 个兴趣点。攻击质量的指标用 1 阶成功率到 4 阶成功率表示。*n* 阶成功率表示正确密钥在攻击后得到的、按概率降序排序的候选密钥集的前 *n* 个的比例。猜测熵[16]表示在多次实验攻击中，正确密钥的平均位置。根据文献[16]给出的猜测熵计算方法，计算每个攻击方法的猜测熵。

为了与标准模板攻击进行对比，本文实验采用独立的攻击能迹集，其中包含 2 000 条能迹。实验中采用多能迹攻击。设攻击能迹数为 m，攻击次数为 $\frac{2\,000}{m}$。

从图 8 可以更明确地观察 5 种模板生成方法之间的比较关系。从表 2 可以看出，EMPDS 方法的攻击成功率基本与模板攻击相当，而 EM 方法最差，EMPD 与 EMPDC 没有本质的区别。

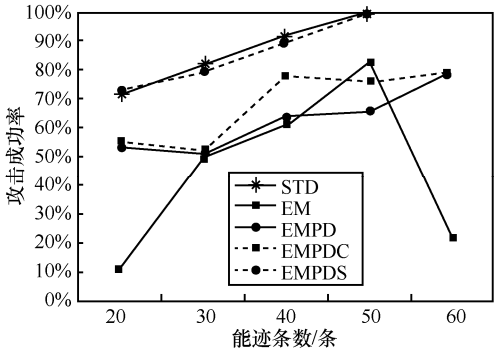


图 8　5 种模板生成方法的 1 阶成功率比较

4.4　对一个能迹集的攻击实验

在本文设定的攻击场景下，由于训练时密钥未知，因此在实际攻击时，不存在训练能迹集与攻击能迹集的区别。本节实验采用一个能迹集，同时用于训练和攻击。本实验的目的是，测试能迹集的大小与攻击成功率的关系。

攻击时并不采用全部能迹，而仅选择在能耗均值向量附近的 30 条能迹作为攻击能迹。由于计算成功率需要多次实验，且每次实验需要采用不同的能迹进行攻击，因此首先选择与能耗均值向量最接近的 600 条能迹作为攻击能迹集，每次实验随机从其中选择 30 条进行攻击，共进行 20 次攻击。此外，根据 4.2 节实验的结果，这里仅对标准模板攻击和采用 EMPDS 的攻击进行比较，如表 3 所示。

从表 3 可以看出，训练能迹集达到 3 000 条后，4 阶成功率可达到 100%；训练能迹集达到 5 000 条后，1 阶成功率可达到 100%，该成功率已优于标准模板攻击。当能迹数在 4 000 条以下时，由于数据稀缺问题，聚类质量出现下降，该问题在标准模板攻击中同样存在。从猜测熵的对比来看，EMPDS 在所有训练能迹数下均小于标准模板攻击。

5　结束语

从实验结果可以看出，改造后的 EM 算法可以相当好地还原出不同汉明重量能耗的成员分布，从而成功

表 2　5 种模板训练方法的攻击成功率和猜测熵对比

攻击方法	攻击能迹数 m	1 阶成功率	2 阶成功率	3 阶成功率	4 阶成功率	猜测熵
STD	20	71.23%	83.56%	84.93%	86.30%	4.96
	30	81.63%	91.84%	97.96%	97.96%	1.73
	40	91.67%	91.67%	97.22%	97.22%	1.33
	50	100.00%	100.00%	100.00%	100.00%	1.00
	60	100.00%	100.00%	100.00%	100.00%	1.00
EM	20	10.96%	17.81%	21.92%	24.66%	49.93
	30	48.98%	59.18%	71.43%	73.47%	8.04
	40	61.11%	69.44%	72.22%	72.22%	29.53
	50	82.76%	86.21%	86.21%	89.66%	3.45
	60	20.83%	29.17%	37.50%	37.50%	14.13
EMPD	20	53.42%	58.90%	61.64%	64.38%	19.74
	30	51.02%	61.22%	63.27%	73.47%	12.84
	40	63.89%	75.00%	77.78%	77.78%	9.89
	50	65.52%	72.41%	72.41%	75.86%	4.55
	60	79.17%	87.50%	87.50%	91.67%	3.33
EMPDC	20	54.79%	58.90%	60.27%	63.01%	15.97
	30	51.02%	57.14%	75.51%	77.55%	9.49
	40	77.78%	77.78%	77.78%	83.33%	8.53
	50	75.86%	79.31%	79.31%	86.21%	3.14
	60	79.17%	87.50%	91.67%	91.67%	2.75
EMPDS	20	72.60%	76.71%	78.08%	82.19%	6.18
	30	79.59%	83.67%	89.80%	91.84%	2.29
	40	88.89%	91.67%	91.67%	91.67%	1.58
	50	100.00%	100.00%	100.00%	100.00%	1.00
	60	100.00%	100.00%	100.00%	100.00%	1.00

表 3　STD 与 EMPDS 训练能迹数与攻击成功率的关系

攻击方法	训练能迹数/条	1 阶成功率	2 阶成功率	3 阶成功率	4 阶成功率	猜测熵
STD	8 000	81.63%	91.84%	97.96%	97.96%	1.73
	7 000	81.63%	91.84%	97.96%	97.96%	1.73
	6 000	81.63%	97.96%	97.96%	97.96%	1.73
	5 000	79.59%	93.88%	93.88%	95.92%	2.59
	4 000	83.67%	91.84%	93.88%	95.92%	4.12
	3 000	69.39%	87.76%	89.80%	89.80%	6.00
	2 000	65.31%	85.71%	87.76%	87.76%	7.55
	1 000	67.35%	81.63%	85.71%	87.76%	25.10
EMPDS	8 000	100.00%	100.00%	100.00%	100.00%	1.15
	7 000	100.00%	100.00%	100.00%	100.00%	1.00
	6 000	92.31%	100.00%	100.00%	100.00%	1.08
	5 000	100.00%	100.00%	100.00%	100.00%	1.00
	4 000	76.92%	100.00%	100.00%	100.00%	1.23
	3 000	69.23%	92.31%	100.00%	100.00%	1.38
	2 000	61.54%	76.92%	76.92%	76.92%	4.92
	1 000	61.54%	84.62%	92.31%	92.31%	1.77

实现模板攻击。这种攻击方式并不需要攻击者掌握对被攻击设备的控制权，甚至也不需要了解任何密钥信息，就能以相当高的成功率攻击出密钥。从效率来看，由于本文方法可以准确地找到 POI，同时合理地设置了 EM 算法的初始参数，聚类效率也比较高（一般 10 次迭代就可以达到阈值）。实验中为了证明本文方法的有效性，与标准模板攻击进行对比，验证了两者的应用场景不同。标准模板攻击通过实验设备得到模板后，只需要少量能迹即可实施攻击。而本文方法虽然不需要实验设备，但事实上需要更多的能迹来以聚类方式建模和实施攻击。本文的主要贡献在于：突破了已有对实验设备完全可控的局限性，详细讨论了 EM 聚类算法在能耗混合高斯分布中的适应性改造，并在实践中证明了这种改造的有效性。

参考文献：

[1] KOCHER P C. Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems[C]//Annual International Cryptology Conference. 1996: 104-113.

[2] KOCHER P, JAFFE J, JUN B. Differential power analysis[C]//Annual International Cryptology Conference. 1999: 388-397.

[3] BRIER E, CLAVIER C, OLIVIER F. Correlation power analysis with a leakage model[C]//International Workshop on Cryptographic Hardware and Embedded Systems. 2004: 16-29.

[4] CHARI S, RAO J R, ROHATGI P. Template attacks[C]//International Workshop on Cryptographic Hardware and Embedded Systems. 2002: 13-28.

[5] LERMAN L, MEDEIROS S F, VESHCHIKOV N, et al. Semi-supervised template attack[C]//International Workshop on Constructive Side-Channel Analysis and Secure Design. 2013: 184-199.

[6] SCHINDLER W, LEMKE K, PAAR C. A stochastic model for differential side channel cryptanalysis[C]//International Workshop on Cryptographic Hardware and Embedded Systems. 2005: 30-46.

[7] GIERLICHS B, LEMKE-RUST K, PAAR C. Templates vs. stochastic methods[C]//International Workshop on Cryptographic Hardware and Embedded Systems. 2006: 15-29.

[8] KARSMAKERS P, GIERLICHS B, PELCKMANS K, et al. Side channel attacks on cryptographic devices as a classification problem[J]. Esat Kuleuven Be, 2009, 7: 36.

[9] LERMAN L, POUSSIER R, BONTEMPI G, et al. Template attacks vs. machine learning revisited (and the curse of dimensionality in side-channel analysis)[C]//International Workshop on Constructive Side-Channel Analysis and Secure Design. 2015: 20-33.

[10] LERMAN L, BONTEMPI G, MARKOWITCH O. Side channel attack: an approach based on machine learning[J]. Center for Advanced Security Research Darmstadt, 2011: 29-41.

[11] BATINA L, GIERLICHS B, LEMKE-RUST K. Differential cluster analysis[M]//Cryptographic Hardware and Embedded Systems-CHES. 2009: 112-127.

[12] CHOU J W, CHU M H, TSAI Y L, et al. An unsupervised learning model to perform side channel attack[C]//Pacific-Asia Conference on Knowledge Discovery and Data Mining. 2013: 414-425.

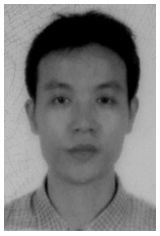
[13] HEYSZL J, IBING A, MANGARD S, et al. Clustering algorithms for non-profiled single-Execution attacks on exponentiations[C]// International Conference on Smart Card Research and Advanced Applications. 2013: 79-93.

[14] LEMKE-RUST K, PAAR C. Gaussian mixture models for higher-order side channel analysis[C]//International Workshop on Cryptographic Hardware and Embedded Systems. 2007: 14-27.

[15] MANGARD S, OSWALD E, POPP T. Power analysis attacks: revealing the secrets of smart card[M]. New York: Springer. 2007.

[16] STANDAERT F X, MALKIN T G, YUNG M. A unified framework for the analysis of side-channel key recovery attacks[C]//Annual International Conference on the Theory and Applications of Cryptographic Techniques. 2009: 443-461.

[作者简介]



吴震（1975−），男，江苏苏州人，成都信息工程大学副教授，主要研究方向为信息安全、密码学、侧信道攻击与防御、信息安全设备设计与检测。

杜之波（1982−），男，山东冠县人，成都信息工程大学副教授，主要研究方向为信息安全、侧信道攻击与防御、天线应用和物联网安全。

王敏（1977−），女，四川资阳人，成都信息工程大学副教授，主要研究方向为网络攻防、侧信道攻击与防御。

向春玲（1990−），女，湖北宜昌人，成都信息工程大学助教，主要研究方向为信息安全、嵌入式系统安全、侧信道攻击与防御。