doi:10.11959/j.issn.1000-436x.2015186

# 基于 QA-NIZK 证明系统的高效简短可验证洗牌方案

程小刚[1,2]，王箭[1]，陈永红[2]
（1. 南京航空航天大学 计算机科学与技术学院，江苏 南京 210016；2. 华侨大学 计算机科学与技术学院，福建 厦门 361021）

**摘 要**：电子投票中为保护投票者的隐私，要由一系列混合服务器对选票进行洗牌操作，为保证洗牌操作的诚实性，需要可验证洗牌方案。现有简短可验证洗牌方案的证明大小要依赖于混合服务器的数量和投票者的数量，效率较低。基于近来的一个高效的 QA-NIZK 证明系统构建了一个高效的简短可验证洗牌方案，不管有多少混合器和投票者其证明大小都是 $O(1)$，即常量大小。具有其独立的意义是在构建中指出原 QA-NIZK 证明系统是可变的。
**关键词**：简短可验证洗牌；可变性；电子投票；QA-NIZK
**中图分类号**：TP309 **文献标识码**：A

# Highly efficient compact verifiable shuffle scheme based on QA-NIZK proof

CHENG Xiao-gang[1,2], WANG Jian[1], CHEN Yong-hong[2]
(1. College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China;
2. College of Computer Science and Technology, Huaqiao University, Xiamen 361021, China)

**Abstract:** To protect the privacy of voters in e-voting, votes should be shuffled by a series of mix servers. To guarantee the honesty of mix servers, verifiable shuffle scheme was needed. However the proof size of existed CVS (compact verifiable shuffle) scheme was dependent on the number of mix servers and the number of voters, which could be very inefficient when there were lots of mix servers and voters. A new CVS scheme was presented with the proof size of only $O(1)$, i.e. constant no matter how many mix servers and voters were involved. The construction is based on an efficient proof system QA-NIZK (quasi-adaptive non-interactive zero knowledge) presented recently. It also points out that the QA-NIZK proof system is malleable, which is of independent interest.
**Key words:** compact verifiable shuffle; malleability; e-voting; QA-NIZK

## 1 引言

在电子投票系统中，投票者把自己的选票加密发送给统计方，统计方对选票解密后统计选举结果，这样简单的处理可能会暴露投票者的隐私，即统计方能把某张选票同某个投票者联系起来，为保护投票者隐私，Chaum 提出了“洗牌（shuffle）”方案[1]，即加密的选票在解密之前先经过一系列混合服务器的洗牌操作（即对密文进行随机化和打乱处理），然后再由统计方进行解密，此时统计方就不再能把选票和投票人联系起来了，从而保护了投票人的隐私。

为防止恶意混合服务器的欺骗行为即防止恶意的混合服务器修改和替换选票，实践中需要的是可验证洗牌(VS, verifiable shuffle)方案[2]，即混合服务器要证明自己诚实地进行了洗牌操作没有欺骗。现有的很多 VS 方案是交互式[3~7]，交互性在实践中并不是一个好的特征，因为交互式的 VS 方案在验证洗过牌的投票时，要求所有的混合服务器都在线，并同验证者进行多个回合的对话，若有混合服务器不在线就无法完成验证。所以从实用的角度出发，非交互式的 VS 方案更好。但现有的非交互式的洗牌方案如文献[8, 9]效率较低，因其证明大小是 $O(nl)$，即依赖于参与的

**收稿日期**：2014-10-15；**修回日期**：2015-01-28
**基金项目**：国家自然科学基金资助项目(61370007)；福建省自然科学基金资助项目(2013J01241)
**Foundation Items:** The National Natural Science Foundation of China (61370007); The Natural Science Foundation of Fujian Province (2013J01241)

混合服务器的数量 $l$ 和投票者的数量 $n$ 的乘积，显然在有些应用中有很多的混合服务器或投票者时，这就是一个严重的效率瓶颈。

近来，Chase 等提出了一个高效的非交互式“简短可验证洗牌方案”CVS(compact verifiable shuffle)[10]，这里“简短”的含义是只用一个证明来证明所有的混合服务器都诚实地完成了工作。其证明的大小为 $O(n^2+l)$[10]（随后的改进工作又把证明大小进一步减小为 $O(n+l)$[11]），比传统的的证明大小 $O(nl)$有了很大提高。

传统的非交互 VS 方案证明大小为 $O(nl)$，是因为每个服务器要证明自己诚实地完成了洗牌操作，$l$ 个服务器就要有 $l$ 个证明,每个证明大小为 $O(n)$（因有 $n$ 张加密的选票），所以最终的大小为 $O(nl)$。而 Chase 等的方案采用了“受控可变”（controlled malleable）证明系统的概念，即每个混合服务器不是单独生成一个证明，而是对上一个混合服务器提供的证明进行修改来继续证明自己的诚实性，使最终只用一个证明来证明所有的混合服务器都诚实地完成了工作，而不是 $l$ 个证明，但这个证明的大小不是常量，而是 $O(n+l)$，其中 $O(n)$用来证明全部的洗牌操作是正确的，$O(l)$用来证明了 $l$ 个服务器都参与到了洗牌操作之中。利用多重签名的技术，文献[12]中把 Chase 等的 CVS 方案中的 $l$ 个服务器的参与证明从 $O(l)$提高到了 $O(1)$；在文献[13]中，Chase 等基于一般的密码学原语进一步扩展了证明系统所支持的可变操作，但所得到的 CVS 证明大小仍然是 $O(n+l)$。

本文中，利用最近提出的高效 QA-NIZK 证明系统[14]进一步提高了 CVS 方案的效率，新的 CVS 方案中用来证明全部的洗牌操作是正确的数据大小仅为 $O(1)$常量级别而不是原来的 $O(n)$，这样结合文献[12]中的方案，可得到 $O(1)$大小的 CVS 方案；本文方案构建思想类似 Chase 等的方案，用一个可变的证明系统生成一个证明来证明所有的洗牌操作都是诚实的。这里的证明系统就是基于文献[14]中的高效 QA-NIZK 证明系统，为此首先注意到 QA-NIZK 是可变的，此外也引入了所谓“支持消息空间”的概念，其实就是满足特定格式的消息，然后结合 QA-NIZK 的可变性与“支持消息空间”来构建高效的 CVS 方案。

## 2　预备知识

### 2.1　原 CVS 方案

Chase 等的 CVS 方案构建是基于其提出的一个新概念 cm-NIZK(controlled malleable NIZK)。第 $i$ 个混合服务器从第 $i-1$ 个混合服务器得到$\{C,C^{(i-1)},\pi_{i-1},\{PK_{i-1}\}\}$，这里 $C$ 是原始的加密投票，$C^{(i-1)}$是到第 $i-1$ 个混合服务器为止的被随机化和重排后的加密投票，$\pi_{i-1}$ 是 cm-NIZK 来证明 $C^{(i-1)}$是原始投票 $C$ 的正确随机化和重排（即洗牌）后的结果，且洗牌操作是由$\{PK_{i-1}\}$集合（此集合包含前 $i-1$ 个混合服务器的公钥）中的前 $i-1$ 个混合服务器完成的。

接收到$\{C, C^{(i-1)}, \pi_{i-1},\{PK_{i-1}\}\}$后，第 $i$ 个混合服务器继续对 $C^{(i-1)}$进行随机化和重排操作后得到 $C^{(i)}$，由于 $\pi_{i-1}$ 是可变的，他可以对 $\pi_{i-1}$ 进行更新得到 $\pi_i$ 来直接证明 $C^{(i)}$是原始投票 $C$ 被正确洗牌后的结果；此外第 $i$ 个混合服务器也把他的公钥 $PK_i$ 加入到集合$\{PK_1, \cdots, PK_{i-1}\}$中去，并且证明自己知道跟 $PK_i$ 相对应的私钥（从而证明自己的合法性），这也是通过更新 $\pi_{i-1}$ 来实现的而不是提供另外一个单独的证明。在文献[10]中，作者也指出著名的 Groth-Sahai 证明系统[15]是可变的，并详述了各种所支持的可变操作，然后用之来构建 CVS 方案。然而虽然在其 CVS 方案中证明（证明所有的混合服务器都诚实地进行了洗牌操作）只有一个，但此证明的大小却是 $O(n^2+l)$（后提高至 $O(n+l)$），即依赖于投票的数量和混合服务器的数量，证明 $C^{(l)}$是 $C$ 的正确洗牌结果的证明大小为 $O(n^2)$（提高后为 $O(n)$），证明混合服务器知道集合$\{PK_1, \cdots, PK_l\}$中的公钥所对应的私钥的证明大小为 $O(l)$。因而总的证明大小为 $O(n^2+l)$（或 $O(n+l)$），当有大量的混合服务器或投票者的时候，则证明尺寸较大，显然这是一个效率瓶颈。

### 2.2　QA-NIZK 证明系统及其可变性

近来在 Asiacrypt 2013 会议上，Jutla 等提出一种非常高效的 NIZK 证明系统 QA-NIZK[14]，如利用此种证明系统来证明（$g,h,rg,rh$）是一 DDH 元组的话，证明只需要一个群元素。下面来简介如何在 SXDH 假设下实现此证明系统。

设置此证明系统，要先生成阶为 $p$ 的双线性群 $G_1$、$G_2$、$G_T$，具有双线性映射 $e:G_1\times G_2\rightarrow G_T$。而 SXDH 假设的含义就是在群 $G_1$ 和 $G_2$ 中 DDH 问题都是难解的。从 $G_1$ 中选 2 个随机元素$(g, f)$，从 $G_2$ 中选一个随机元素 $g_2$，并从群 $Z_p$ 中 2 个随机数 $b$、$d$，生成公共参考字符串（common reference string），如下

$$CRS_p = dg + b^{-1}f \tag{1}$$

$$CRS_v=(CRS_{v1},CRS_{v2},CRS_{v3}) = (bd\cdot g_2, g_2, -bg_2) \quad (2)$$

其中，$CRS_p$ 表示是由证明方（prover）所使用的 $CRS$，而 $CRS_v$ 是由验证方（verifier）所使用的 $CRS$。

要证明$(U, U') = (rg, rh) \in G_1^2$是一 DDH 元组，证明方只要利用其自己知道的秘密值 $r$ 计算证明为

$$\pi = rCRS_p = r(dg + b^{-1}f) \quad (3)$$

要验证此证明，验证方只要检查下面的 PPE 方程是否成立。

$$E((U, U', \pi), CRS_v) = e(U, CRS_{v1})e(U', CRS_{v2})\cdot e(\pi, CRS_{v3})=0_T \quad (4)$$

对于正确生成的证明，显然上述方程成立，因为

$$E((U, U', \pi), CRS_v) = e(rg, bdg_2)e(rh,g_2)\cdot e(r(dg + b^{-1}f), -bg_2)=0_T \quad (5)$$

下面来看 QA-NIZK 证明系统的可变性，假设一证明方已经生成了 $\pi$ 来证明$(g,h,U,U')$是一 DDH 元组，那么任何人可对$(U, U')$进行重随机化得到另一 DDH 元组$(U_1, U_1')$，即$(U+sg,U'+sh) =((r+s)g, (r+s)h)=(U_1,U_1')$。然后可对 $\pi$ 进行更新来证明$(U_1,U_1')$是一 DDH 元组，即令 $\pi_1=\pi+sCRS_p=(r+s)CRS_p$，显然 $\pi_1$ 是$(U_1, U_1')$是 DDH 元组的合法证明。

## 3 构建

高效 CVS 方案构建如下（注意上文中用加法表示群运算，下面为叙述方便，群中的运算用乘法表示，且加密是在双线性群中的 $G_1$ 中进行的）。

1) 初始投票：第 $i$ 个投票者作出投票 $V_i$，并用 ElGamal 加密方案进行加密得到$C_i = (g^{r_i}, h^{r_i}V_i)$，每个投票者并对其加密的选票 $C_i$ 进行签名来认证其合法性；然后所有投票者把加过密的选票$\{C_1,\cdots,C_n\}$及其签名传输给第一个混合服务器。

为防止恶意的混合服务器篡改选票，在此要求投票消息 $V_i$ 满足一定的格式（如限定消息的前一半二进制位为 0），称之为“支持消息空间”，其中的安全性考虑将在下节讨论。

2) 第一个混合服务器：收到初始加密选票$C=\{C_1,\cdots,C_n\}$后，先验证签名是否合法，不合法则退出，否则对 $C$ 进行随机化和重排操作，随机化就是选一随机数 $r_i$'把原来的加密选票

$$C_i = (C_{i1}, C_{i2}) = (g^{r_i}, h^{r_i}V_i) \quad (6)$$

更新为

$$(g^{r_i'}C_{i1}, h^{r_i'}C_{i2}) \quad (7)$$

然后打乱重排得到$C'=\{C_1', \cdots, C_n'\}$。接下来他要证明 $C'$的确是 $C$ 正确随机化和重排后的结果。

利用上述的 QA-NIZK 证明系统来生成此证明 $\pi_1$，如果是正确生成的 $C$ 和 $C'$，那么有

$$\prod_{i=1}^{n} C_i' = (g^{\sum r_i'}g^{\sum r_i}, h^{\sum r_i'}h^{\sum r_i}V_1V_2\cdots V_n) \quad (8)$$

$$\prod_{i=1}^{n} C_i = (g^{\sum r_i}, h^{\sum r_i}V_1V_2\cdots V_n) \quad (9)$$

所以

$$\frac{\prod_{i=1}^{n} C_i'}{\prod_{i=1}^{n} C_i} = (g^{\sum r_i'}, h^{\sum r_i'}) \quad (10)$$

就是一 DDH 元组，按上述 QA-NIZK 证明系统，证明这一事实的 $\pi_1$ 只要一个群元素，因而十分高效；如上所述，$\pi_1$ 是可变的，第二个和后续的混合服务器可不断对其更新来分别证明其各自的洗牌操作是诚实的。

3) 第 2 个混合服务器：当第 2 个混合服务器从第 1 个混合服务器收到$\{C, C', \pi_1\}$后，先验证证明 $\pi_1$ 是否正确，然后要随机化和随机重排加密选票 $C'$ 得到 $C^{(2)}$（方法同上）。为证明他做的工作是诚实的，他要更新 $\pi_1$ 为 $\pi_2$ 来证明 $\frac{\prod_{i=1}^{n} C_i^{(2)}}{\prod_{i=1}^{n} C_i}$ 是一 DDH 元组，方法也同上，最后他把 $\{C, C^{(2)}, \pi_2\}$ 传递给下一个混合服务器。

4) 第 $i$ 个混合服务器：类似于第 2 个混合服务器，当第 $i$ 个混合服务器从第 $i-1$ 个混合服务器收到$\{C, C^{(i-1)}, \pi_{i-1}\}$后，他要先验证证明 $\pi_{i-1}$ 是否正确，然后随机化和随机重排加密选票 $C^{(i-1)}$得到 $C^{(i)}$，再把证明 $\pi_{i-1}$ 更新为 $\pi_i$，来证明 $\frac{\prod_{i=1}^{n} C_i^{(i)}}{\prod_{i=1}^{n} C_i}$ 是一 DDH 元组。

5) 选票验证与解密：当所有的 $l$ 个混合服务器都完成了洗牌工作后，$\{C, C^{(l)}, \pi_l\}$ 就由最后一个混合服务器送到一个权威机构，来解密和统计选票。首先此权威机构要验证证明 $\pi_l$ 是否合法，不合法就退出，否则就解密被洗牌过的加密选票 $C^{(l)}$ 并统计选举结果。显然一个不诚实的权威机构可以选择解密原来的选票 $C$ 而不是 $C^{(l)}$，这样就可能暴露选举人的隐私，为防止此种恶意行为，更好的是采用阈值解密过程，即必须有几个权威机构合作才能解密选票，来保证机密被洗牌过后的选票 $C^{(l)}$而不是原来的 $C$。

## 4　安全性

**定理 1**　基于 ElGmal 加密方案的语义安全性（基于DDH假设，在双线性群中则为SXDH假设），在上述 CVS 方案中，一个恶意的混合服务器不能更改任何一个选票且使得证明仍然成立。

**证明**　众所周知，ElGamal 加密方案是可变，因为对 $m$ 的 ElGamal 的加密为 $(g^r, h^r m)$，任何人可以把之改为对 $mm'$ 的加密，只要用 $m$’ 乘以 $h^r m$ 即可，得到 $h^r mm$’ 但如上所说，要求选票明文的格式要满足一定的条件（如要求前 $\frac{n}{2}$ 比特为 0，所谓“支持的消息空间”）。所以基于 ElGamal 加密方案的语义安全性，敌手只能知道密文 $(g^r, h^r m)$ 是对一个随机消息的加密，虽然 $m$ 满足那种严格的格式限制。敌手当然能更改密文为 $h^r mm$’，但既然 $m$ 是随机的，那么对敌手来说，$M=mm'$也是随机的，所以 $M$ 满足那种严格的格式的概率是可忽略的。

所以已证明了一个恶意的混合服务器不能更改一个合法的选票为另一个，而只能对已有的加密选票随机化。因为如果他做了这种修改，那么在解密选票时，很可能这种恶意修改会被抓住（因不符合限定的格式）。

接下来证明一个恶意混合服务器不能用一个新的选票来代替任何已有的选票，且使得证明 $\pi$ 仍是合法的。

假设一个恶意混合服务器把 $C_l$ 替换为 $C_1$’=($g^r$,$h^r V_1$’)（是对新选票 $V_1$’的加密），因为要构建证明，他要证明下列是 DDH 元组

$$\prod_{i=1}^{n}\frac{C_i'}{C_i}=\left(\frac{g^{r+(r_2+r_2')+\cdots+(r_n+r_n')}}{g^{r_1+r_2+\cdots+r_n}},\frac{h^{r+(r_2+r_2')+\cdots+(r_n+r_n')}V_1'V_2\cdots V_n}{h^{r_1+r_2+\cdots+r_n}V_1V_2\cdots V_n}\right)$$
$$=\left(\frac{g^{r+r_2'+\cdots+r_n'}}{g^{r_1}},\frac{h^{r+r_2'+\cdots+r_n'}V_1'}{h^{r_1}V_1}\right)$$
$$=\left(g^{r+r_2'+\cdots+r_n'-r_1},h^{r+r_2'+\cdots+r_n'-r_1}\frac{V_1'}{V_1}\right)\tag{11}$$

为此，$\frac{V_1'}{V_1}$ 要为 1。而此事件发生的概率是可忽略的，因为对敌手来说 $V_1$ 是随机的（基于 ElGamal 加密方案的语义安全性）。

## 5　效率比较

1) 与非交互式 VS 方案效率对比

从表 1 中可见，同已有的非交互式 VS 方案比较，本文方案有较大的效率优势，即实现了常量级的证明大小，即通信复杂度是 $O(1)$的，而已有的方案通信复杂方式都依赖于投票者数量和参与的混合服务器数量。

2) 与交互式 VS 方案对比

当前大量的洗牌方案的证明是交互式的，即要

**表 1**　本文方案与已有非交互式 VS 方案效率对比

| 方案 | 诚实洗牌的证明大小 | 服务器参与的证明大小 | 总大小 | 备注 |
|---|---|---|---|---|
| GL 方案 | 每个服务器生成一个 $O(n)$大小的证明证明其诚实的洗牌操作 | | $O(nl)$ | $n$ 是投票者数量<br>$l$ 是混合服务器数量 |
| LZ 方案 | 每个服务器生成一个 $O(n)$大小的证明证明其诚实的洗牌操作 | | $O(nl)$ | 效率只比 GL 方案有少许提高，渐近复杂度是一致的 |
| 原 CVS 方案 | $O(l)$ | $O(n)$ | $O(n+l)$ | |
| 本文方案 | $O(1)$ | $O(1)$ | $O(1)$ | |

**表 2**　本文方案与交互式 VS 方案效率对比

| 方案 | 单个服务器证明回合数 | $l$ 个服务器证明回合数 | 单服务器通信复杂度 | $l$ 个服务器通信复杂度 |
|---|---|---|---|---|
| BG 方案 | 9 | $9l$ | $\Theta\left(n^{\frac{1}{2}}\right)$ | $l\Theta\left(n^{\frac{1}{2}}\right)$ |
| GI 方案 | 7 | $7l$ | $\Theta\left(n^{\frac{2}{3}}\right)$ | $l\Theta\left(n^{\frac{2}{3}}\right)$ |
| Groth 方案 | 7 | $7l$ | $\Theta(n)$ | $l\Theta(n)$ |
| TW 方案 | 5 | $5l$ | $\Theta(n)$ | $l\Theta(n)$ |
| Peng 方案 | 3 | $3l$ | $\Theta(n)$ | $l\Theta(n)$ |
| 本文方案 | 1 | 1 | $O(1)$ | $O(1)$ |

来回数个回合才能证明服务器洗牌的诚实性；在表 2 中比较了本文方案与现有交互式 VS 方案的效率，可见本文方案除了在回合数上远优于交互式的方案之外，在通信复杂度上也有较大优势。

## 6　结束语

本文把 CVS 方案的证明大小减小为常数大小，大大提高了效率。主要所用的技术是最近提出的 QA-NIZK 证明系统，并指出其是可变的。

然而该方案仍有一些缺点，值得进一步的改进研究，如用到了所谓“支持的消息空间”来防止恶意的混合服务器篡改加密过的选票，因而选票明文中有一部分比特就被浪费了（为符合限定的格式条件），效率不是很高。

## 参考文献：


[1] CHAUM D. Untraceable electronic mail, return addresses, and digital pseudonyms[J]. Communications of ACM, 1981, 24(2): 84-88.

[2] SAKO K, KILIAN J. Receipt-free mix-type voting scheme[A]. EUROCRYPT 1995[C]. Springer, 1995. 393-403.

[3] BAYER S, GROTH J. Efficient zero-knowledge argument for correctness of a shuffle[A]. EUROCRYPT 2012[C]. Springer, 2012. 281-300.

[4] GROTH J, ISHAI Y. Sub-linear zero-knowledge argument for correctness of a shuffle[A]. EUROCRYPT 2008[C]. Springer, 2008. 379-396.

[5] GROTH J. A verifiable secret shuffle of homomorphic encryptions[J]. Journal of Cryptology, 2010, 23(4): 546-579.

[6] TERELIUS B,WIKSTROM D. Proofs of restricted shuffles[A]. AFRICACRYPT 2010[C]. Springer, 2010. 100-113.

[7] PENG K. A shuffle to achieve high efficiency through pre-computation and batch verification[J]. International Journal of Information Security, 2013, 12(4): 337-345.

[8] GROTH J, LU S. A non-interactive shuffle with pairing based verifiability[A]. ASIACRYPT 2007[C]. Springer, 2007. 51-67.

[9] LIPMAA H, ZHANG B. A more efficient computationally sound non-interactive zero-knowledge shuffle argument[A]. SCN 2012[C]. Springer, 2012. 477-502.

[10] CHASE M, KOHLWEISS M, LYSYANSKAYA A, *et al*. Malleable proof systems and applications[A]. EUROCRYPT 2012[C]. Springer, 2012. 281-300.

[11] CHASE M, KOHLWEISS M, LYSYANSKAYA A, *et al*. Verifiable elections that scale for free[A]. PKC 2013[C]. Springer, 2013. 479-496.

[12] CHENG X, WANG J, CHEN Y. Improvement of a compact verifiable shuffle scheme[J]. ICIC Express Letters, Part B: Applications, 2014, 5(4): 1115-1119.

[13] CHASE M, KOHLWEISS M, LYSYANSKAYA A, *et al*. Succinct malleable NIZKs and an application to compact shuffles[A]. TCC 2013[C]. Springer, 2013. 100-119.

[14] JUTLA C, ROY A. Shorter quasi-adaptive NIZK proofs for linear subspaces[A]. ASIACRYPT 2013 Part I [C]. Springer, 2013. 1-20.

[15] GROTH J, SAHAI A. Efficient non-interactive proof systems for bilinear groups[A]. EUROCRYPT 2008[C]. Springer, 2008. 415-432.



**作者简介：**

**程小刚**（1973-），男，安徽六安人，南京航空航天大学博士生，华侨大学讲师，主要研究方向为应用密码学。

**王箭**（1968-），男，江苏南京人，南京航空航天大学教授、博士生导师，主要研究方向为信息安全。

**陈永红**（1974-），男，湖北巴东人，华侨大学教授，主要研究方向为图像处理、计算机控制、信息安全技术等。