NTNU | Norwegian University of Science and Technology

LATTICE-BASED PROOF OF SHUFFLE AND APPLICATIONS TO ELECTRONIC VOTING

eprint.iacr.org/2021/338.pdf

Diego F. Aranha, Carsten Baum, Kristan Gjøsteen,
Tjerand Silde and Thor Tunge

October 15, 2021

Introduction

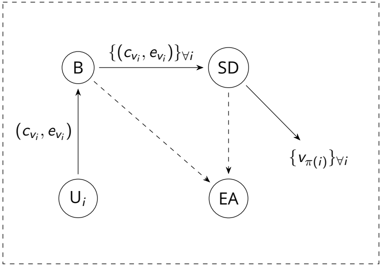
Preliminaries

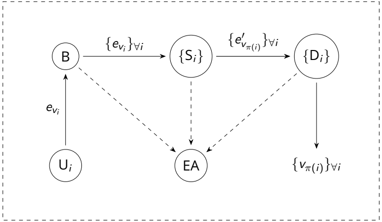
Proof of Shuffle

Mixing Network

Verifiable Decryption

Electronic Voting





Introduction - Goals

1. Build a zero-knowledge protocol to prove correct shuffle of messages
2. Extend the shuffle to handle ciphertexts instead of messages
3. Build a mixing network from the extended shuffle
4. Extend the encryption scheme to support verifiable distributed decryption
5. Combine everything to construct systems for electronic voting
6. Use primitives based on lattices to achieve post-quantum security

Note: The proof of security is in ROM, not QROM.

Preliminaries - Commitment

Algorithms:

$\texttt{Com}$: samples randomness $\boldsymbol{r}_m$ and commits to m as $[m] = \texttt{Com}(m; \boldsymbol{r}_m)$.

$\texttt{Open}$: takes as input $([m], m, \boldsymbol{r}_m)$ and verifies that $[m] \stackrel{?}{=} \texttt{Com}(m; \boldsymbol{r}_m)$.

Properties:

$\texttt{Binding}$: it is hard to find $m \neq \hat{m}$ and $\boldsymbol{r}_m \neq \hat{\boldsymbol{r}}_{\hat{m}}$ s.t. $\texttt{Com}(m; \boldsymbol{r}_m) = \texttt{Com}(\hat{m}; \hat{\boldsymbol{r}}_{\hat{m}})$.

$\texttt{Hiding}$: it is hard to distinguish $\texttt{Com}(m; \boldsymbol{r}_m)$ from $\texttt{Com}(0; \boldsymbol{r}_0)$ when given m.

For more details about the commitment scheme see Baum et al. [BDL+18].

Preliminaries - Proof of Linearity

Let

$$[x] = \texttt{Com}(x; \boldsymbol{r}) \quad \text{and} \quad \left[x'\right] = [\alpha x + \beta] = \texttt{Com}(x'; \boldsymbol{r}').$$

Then the protocol Π_{Lin} is a sigma-protocol to prove the relation $x' = \alpha x + \beta$, given the commitments $[x], [x']$ and the scalars α, β.

For more details about the proof of linearity see Baum et al. [BDL+18].

Preliminaries - Amortized Proof of Shortness

Let

$$[x_1] = \mathtt{Com}(x_1; \boldsymbol{r}_1), \quad [x_2] = \mathtt{Com}(x_2; \boldsymbol{r}_2), \quad \ldots, \quad [x_n] = \mathtt{Com}(x_n; \boldsymbol{r}_n),$$

where all are commitments to short values. Then the protocol Π_A is a sigma-protocol to prove that the underlying messages of $[x_1], [x_2], \ldots, [x_n]$ are bounded.

For more details see the approximate amortized proof by Baum et al. [BBC+18] and the exact amortized proof by Bootle et al. [BLNS20].

Preliminaries - BGV Encryption

`KeyGen` samples random $a \xleftarrow{\$} R_q$, short $s \leftarrow R_q$ and noise $e \leftarrow \mathcal{N}_{\sigma_{\mathrm{E}}}$.
The algorithm outputs $\mathtt{pk} = (a, b) = (a, as + pe)$ and $\mathtt{sk} = s$.

`Enc` samples a short $r \leftarrow R_q$ and noise $e_1, e_2 \leftarrow \mathcal{N}_{\sigma_{\mathrm{E}}}$, and outputs
$(u, v) = (ar + pe_1, br + pe_2 + m)$.

`Dec` outputs $m \equiv v - su \mod q \mod p$ when noise is bounded by $\lfloor q/2 \rfloor$.

For more details about the encryption scheme see Brakerski et al. [BGV12].

Proof of Shuffle - Setting

- Public information: sets of commitments $\{[m_i]\}_{i=1}^{\tau}$ and messages $\{\hat{m}_i\}_{i=1}^{\tau}$.
- P knows the openings $\{(m_i, \boldsymbol{r}_{m_i}, f_i)\}_{i=1}^{\tau}$ of the commitments $\{[m_i]\}_{i=1}^{\tau}$,

  and P knows a permutation γ such that $\hat{m}_i = m_{\gamma^{-1}(i)}$ for all $i = 1, ..., \tau$.
- We construct a $4 + 3\tau$-move ZKPoK protocol to prove the statement:

$$R_{\mathsf{Shuffle}} = \left\{ (x, w) \;\middle|\; \begin{array}{l} x = ([m_1], \ldots, [m_\tau], \hat{m}_1, \ldots, \hat{m}_\tau, \hat{m}_i), \\ w = (\gamma, f_1, \ldots, f_\tau, \boldsymbol{r}_1, \ldots, \boldsymbol{r}_\tau), \gamma \in S_\tau, \\ \forall i \in [\tau] : \ \mathtt{Open}(\left[m_{\gamma^{-1}(i)}\right], \hat{m}_i, \boldsymbol{r}_i, f_i) = 1 \end{array} \right\}$$

Proof of Shuffle - Linear System

First, the verifier sends a challenge ρ to shift all commitments and messages $M_i = m_i - \rho$ and $\hat{M}_i = \hat{m}_i - \rho$ to ensure that all messages are invertible.

Secondly, P draws θ_i uniformly at random, and computes the commitments:

$$\begin{aligned} [D_1] &= \left[\theta_1 \hat{M}_1\right] \\ \forall j \in \{2, \ldots, \tau - 1\} : \; [D_j] &= \left[\theta_{j-1} M_j + \theta_j \hat{M}_j\right] \\ [D_\tau] &= [\theta_{\tau-1} M_\tau] \,. \end{aligned} \tag{1}$$

Proof of Shuffle - Linear System

P receives a challenge β from V and computes s_i such that the following equations are satisfied:

$$
\begin{aligned}
\beta M_1 + s_1 \hat{M}_1 &= \theta_1 \hat{M}_1 \\
\forall j \in \{2, \ldots, \tau - 1\} : \; s_{j-1} M_j + s_j \hat{M}_j &= \theta_{j-1} M_j + \theta_j \hat{M}_j \\
s_{\tau-1} M_\tau + (-1)^\tau \beta \hat{M}_\tau &= \theta_{\tau-1} M_\tau .
\end{aligned}
\tag{2}
$$

Proof of Shuffle - Linear System

We can rewrite these equations as a linear system:

$$\begin{bmatrix} M_1 & \hat{M}_1 & 0 & \dots & 0 & 0 \\ 0 & M_2 & \hat{M}_2 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & M_{\tau-1} & \hat{M}_{\tau-1} \\ (-1)^\tau \hat{M}_\tau & 0 & 0 & \dots & 0 & M_\tau \end{bmatrix} \begin{bmatrix} \beta \\ s_1 \\ \vdots \\ s_{\tau-2} \\ s_{\tau-1} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}$$

We observe that the determinant of the matrix is equal to $\prod_{i=1}^{\tau} M_i - \prod_{i=1}^{\tau} \hat{M}_j$.
If the statement is false, it follows from the Schwartz–Zippel lemma that this system (with high probability) does not have a solution (over the choice of β).

Proof of Shuffle - Linear System

P uses the protocol Π_{Lin} to prove that each commitment $[D_i]$ satisfies the equations (2). In order to compute the s_i values, we can use the following fact:

Lemma

Choosing

$$s_j = (-1)^j \cdot \beta \prod_{i=1}^{j} \frac{M_i}{\hat{M}_i} + \theta_j \tag{3}$$

for all $j \in 1, \ldots, \tau - 1$ *yields a valid assignment for Equation* (2).

Proof of Shuffle - Protocol

Zero-Knowledge Proof Π_{Shuffle} of Correct Shuffle

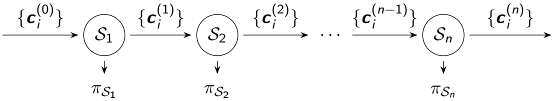
Prover, P		Verifier, V
	$\xleftarrow{\rho}$	$\rho \xleftarrow{\$} R_q \setminus \{\hat{m}_i\}_{i=1}^{\tau}$
$\hat{M}_i = \hat{m}_i - \rho$ $M_i = m_i - \rho$		$\hat{M}_i = \hat{m}_i - \rho$ $[M_i] = [m_i] - \rho$
$\theta_i \xleftarrow{\$} R_q, \forall i \in [\tau - 1]$ Compute $[D_i]$ as in Eq. (1), i.e. $[D_1] = [\theta_1 \hat{M}_1], [D_\tau] = [\theta_{\tau-1} M_\tau]$, $[D_i] = [\theta_{i-1} M_i + \theta_i \hat{M}_i]$ for $i \in [\tau - 1] \setminus \{1\}$	$\xrightarrow{\{[D_i]\}_{i=1}^{\tau}}$	
	$\xleftarrow{\beta}$	$\beta \xleftarrow{\$} R_q$
Compute $s_i, \forall i \in [\tau - 1]$ as in (3).	$\xrightarrow{\{s_i\}_{i=1}^{\tau-1}}$	
		Use Π_{Lin} to prove that (1) $\beta[M_1] + s_1 \hat{M}_1 = [D_1]$ (2) $\forall i \in [\tau - 1] \setminus \{1\} :\ s_{i-1}[M_i] + s_i \hat{M}_i = [D_i]$ (3) $s_{\tau-1}[M_\tau] + (-1)^\tau \beta \hat{M}_\tau = [D_\tau]$ i.e. all equations from (2)

Proof of Shuffle - Performance

- Optimal parameters for the commitment scheme is $q \approx 2^{32}$ and $N = 2^{10}$.
- The proof of linearity use Gaussian noise of standard deviation $\sigma_C \approx 2^{15}$.
- The prover sends 1 commitment, 1 ring-element and 1 proof per message.
- The shuffle proof is of total size $\approx 22\tau$ KB for τ messages.
- The shuffle proof takes $\approx 27\tau$ ms to compute for τ messages.

Mixing Network - Extending the Shuffle

- We extend the shuffle to ciphertexts instead of messages
- We create a mixing network that does the following:
  1. Re-randomize the ciphertexts
  2. Commit to the randomness
  3. Permute the ciphertexts
  4. Prove that shuffle is correct
  5. Prove that the randomness is short
- Integrity follows from the ZK-proofs
- Privacy if at least one server is honest

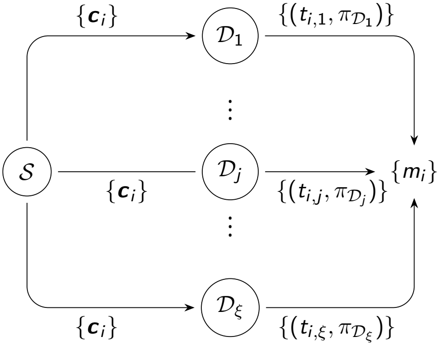


Verifiable Decryption - Distributed Decryption

Verifiable distributed decryption protocol:

- On input key s_j and ciphertext (u, v), sample large noise E_j, output $t_j = s_j u + pE_j$.
- We use Π_{Lin} to prove correct computation.
- We use Π_{A} to prove that E_j is bounded.

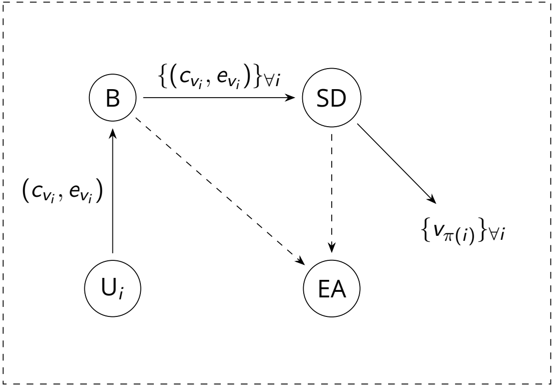
We obtain the plaintext as $m \equiv (v - t \mod q) \mod p$, where $t = t_1 + t_2 + \ldots + t_\xi$.



NTNU | Norwegian University of Science and Technology

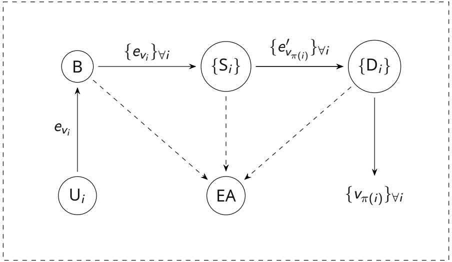
Electronic Voting - Verifiable Shuffle-Decryption

- SD both shuffle and decrypt the votes.
- Integrity follows from the ZK-proof.
- Privacy if B and SD does not collude.



Electronic Voting - Verifiable Mix-Net and Distributed Decryption

- $\{\mathcal{S}_i\}$ may consist of many shuffle-servers.
- $\{\mathcal{D}_i\}$ consists of many decryption-servers.
- Integrity follows from the ZK-proofs.
- Privacy holds if the following is true:
  1. at least one shuffle-server is honest, and
  2. at least one decryption-server is honest.



Thank you! Any questions?

NTNU | Norwegian University of Science and Technology

Carsten Baum, Jonathan Bootle, Andrea Cerulli, Rafaël del Pino, Jens Groth, and Vadim Lyubashevsky.
Sub-linear lattice-based zero-knowledge arguments for arithmetic circuits.
In Hovav Shacham and Alexandra Boldyreva, editors, *CRYPTO 2018, Part II*, volume 10992 of *LNCS*, pages 669–699. Springer, Heidelberg, August 2018.

Carsten Baum, Ivan Damgård, Vadim Lyubashevsky, Sabine Oechsner, and Chris Peikert.
More efficient commitments from structured lattice assumptions.
In Dario Catalano and Roberto De Prisco, editors, *SCN 18*, volume 11035 of *LNCS*, pages 368–385. Springer, Heidelberg, September 2018.

Zvika Brakerski, Craig Gentry, and Vinod Vaikuntanathan.
(Leveled) fully homomorphic encryption without bootstrapping.
In Shafi Goldwasser, editor, *ITCS 2012*, pages 309–325. ACM, January 2012.

Jonathan Bootle, Vadim Lyubashevsky, Ngoc Khanh Nguyen, and Gregor Seiler.
More efficient amortization of exact zero-knowledge proofs for LWE.
Cryptology ePrint Archive, Report 2020/1449, 2020.
https://eprint.iacr.org/2020/1449.

Ivan Damgård, Valerio Pastro, Nigel P. Smart, and Sarah Zakarias.
Multiparty computation from somewhat homomorphic encryption.
In Reihaneh Safavi-Naini and Ran Canetti, editors, *CRYPTO 2012*, volume 7417 of *LNCS*, pages 643–662. Springer, Heidelberg, August 2012.

C. Andrew Neff.
A verifiable secret shuffle and its application to e-voting.
In Michael K. Reiter and Pierangela Samarati, editors, *ACM CCS 2001*, pages 116–125. ACM Press, November 2001.