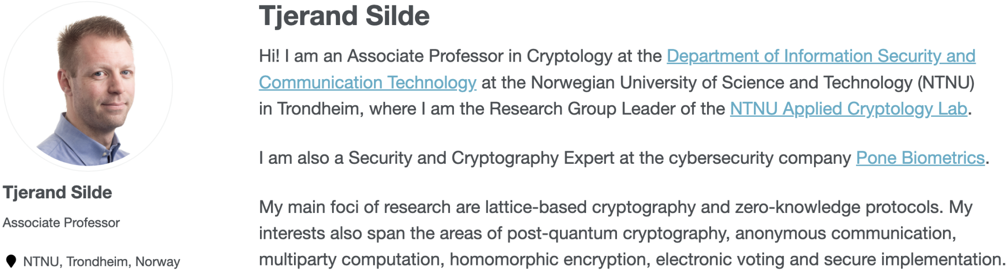
NTNU | Norwegian University of Science and Technology

LATTICE-BASED ELECTRONIC VOTING

Diego F. Aranha, Carsten Baum, Kristian Gjøsteen, Patrick Hough,
Caroline Sandsbråten, **Tjerand Silde** and Thor Tunge

About



Tjerand Silde

Associate Professor

NTNU, Trondheim, Norway

Tjerand Silde

Hi! I am an Associate Professor in Cryptology at the Department of Information Security and Communication Technology at the Norwegian University of Science and Technology (NTNU) in Trondheim, where I am the Research Group Leader of the NTNU Applied Cryptology Lab.

I am also a Security and Cryptography Expert at the cybersecurity company Pone Biometrics.

My main foci of research are lattice-based cryptography and zero-knowledge protocols. My interests also span the areas of post-quantum cryptography, anonymous communication, multiparty computation, homomorphic encryption, electronic voting and secure implementation.

Figure: Personal website: `tjerandsilde.no`

Teaching

This will be the official course website for TTM4205 Secure Cryptographic Implementations during Fall of 2023.

Course Description

The course covers how to implement, analyse, attack, protect and securely compose cryptographic algorithms in practice. It goes in depth on how to implement computer arithmetic, attacking implementations using side-channel attacks and fault injection, exploit padding oracles and low-entropy randomness, utilise techniques to defend against these attacks, and how to securely design misuse-resistant APIs.

See the full course description at ntnu.edu/studies/courses/TTM4205.

NTNU Applied Cryptology Lab 2023

NTNU Applied Cryptology Lab 2024

Contents

Electronic Voting

Preliminaries

Protocols

Voting Schemes

Performance

Contents

Electronic Voting

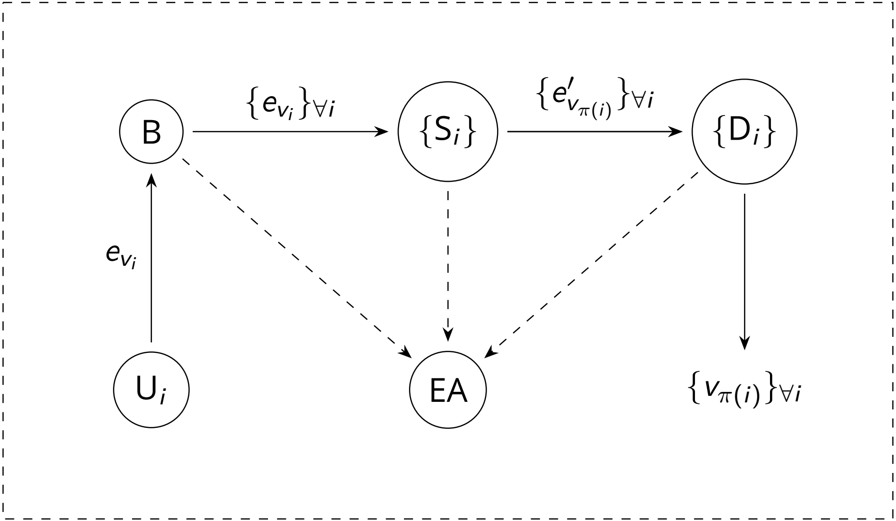
Preliminaries

Protocols

Voting Schemes

Performance

Electronic Voting



Contents

Electronic Voting

Preliminaries

Protocols

Voting Schemes

Performance

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 $[m] \overset{?}{=} \texttt{Com}(m; \boldsymbol{r}_m)$.

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.

Here we can use the BDLOP18 lattice-based commitment scheme.

Proof of Linearity

Let

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

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.

Here we can use the BDLOP18 proof of linear relations.

Amortized Proof of Boundedness

Let

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

for bounded norm values x_i. Let Π_{BND} be a sigma-protocol for this relation.

We have approximate proofs by BBCdGL18 and exact proofs by BLNS20.

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 the BGV12 paper.

Contents

Electronic Voting

Preliminaries

Protocols

Voting Schemes

Performance

Shuffle

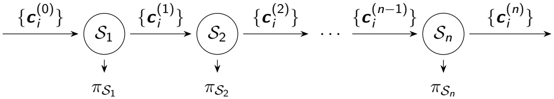
- Public information: sets of commitments $\{[m_i]\}_{i=1}^{\tau}$ and messages $\{\hat{m}_i\}_{i=1}^{\tau}$.
- P knows the openings $\{(m_i, \boldsymbol{r}_{m_i})\}_{i=1}^{\tau}$ of the commitments $\{[m_i]\}_{i=1}^{\tau}$,

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

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

Extending the Shuffle

- We extend the shuffle to ciphertext vectors instead of single messages
- We create a mix-net as follows:
  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



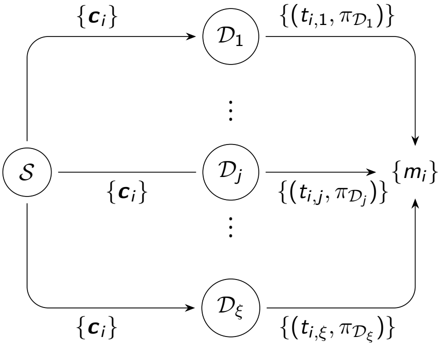
$\{\boldsymbol{c}_i^{(0)}\}$
$\mathcal{S}_1$
$\{\boldsymbol{c}_i^{(1)}\}$
$\mathcal{S}_2$
$\{\boldsymbol{c}_i^{(2)}\}$
$\cdots$
$\{\boldsymbol{c}_i^{(n-1)}\}$
$\mathcal{S}_n$
$\{\boldsymbol{c}_i^{(n)}\}$
$\pi_{\mathcal{S}_1}$
$\pi_{\mathcal{S}_2}$
$\pi_{\mathcal{S}_n}$

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_{BND} to prove that E_j is bounded.

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



Contents

Electronic Voting

Preliminaries

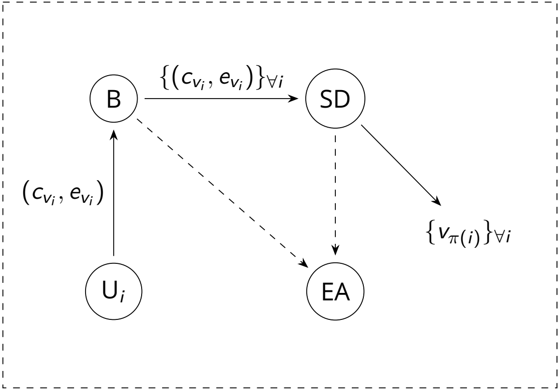
Protocols

Voting Schemes

Performance

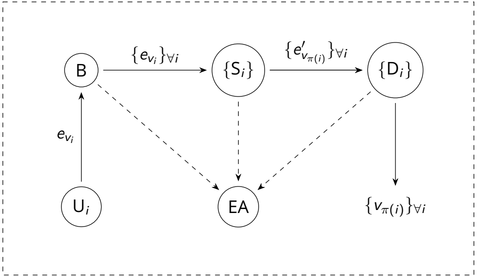
Verifiable Shuffle-Decryption

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


B
SD
U$_i$
EA
$\{(c_{v_i}, e_{v_i})\}_{\forall i}$
(c_{v_i}, e_{v_i})
$\{v_{\pi(i)}\}_{\forall i}$

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.



Contents

Electronic Voting

Preliminaries

Protocols

Voting Schemes

Performance

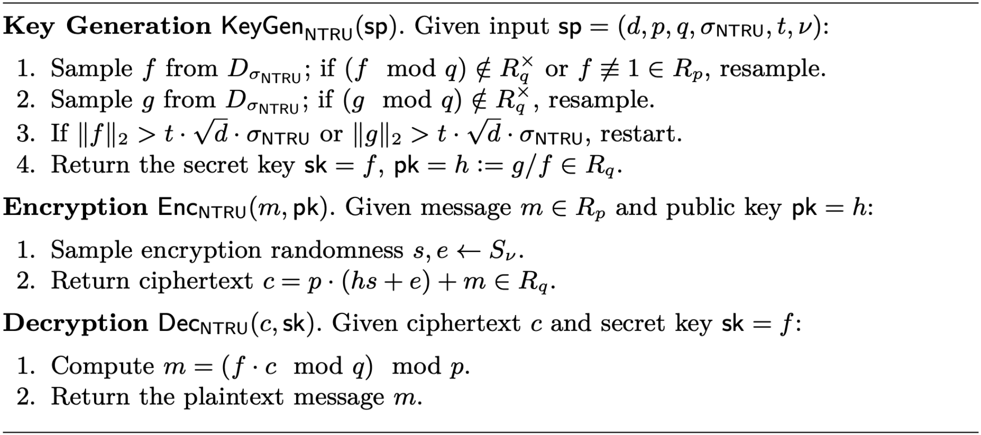
Proof of Shuffle [CT-RSA'21]

- Optimal parameters for the commitment scheme is $q \approx 2^{32}$ and $N = 2^{10}$.
- 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.

Verifiable Mixing and Decryption [CCS'23]

- Optimal parameters for the system is $q \approx 2^{78}$ and $N = 2^{12}$.
- Commitments and ciphertexts are of size ≈ 80 KB each.
- The mixing proof is of size $\approx 370\tau$ KB and takes $\approx 261\tau$ ms.
- The decryption proof is of size $\approx 157\tau$ KB and takes $\approx 138\tau$ ms.

NTRU Encryption



Key Generation $\mathsf{KeyGen}_{\mathsf{NTRU}}(\mathsf{sp})$. Given input $\mathsf{sp} = (d, p, q, \sigma_{\mathsf{NTRU}}, t, \nu)$:

1. Sample f from $D_{\sigma_{\mathsf{NTRU}}}$; if $(f \mod q) \notin R_q^\times$ or $f \not\equiv 1 \in R_p$, resample.
2. Sample g from $D_{\sigma_{\mathsf{NTRU}}}$; if $(g \mod q) \notin R_q^\times$, resample.
3. If $\|f\|_2 > t \cdot \sqrt{d} \cdot \sigma_{\mathsf{NTRU}}$ or $\|g\|_2 > t \cdot \sqrt{d} \cdot \sigma_{\mathsf{NTRU}}$, restart.
4. Return the secret key $\mathsf{sk} = f$, $\mathsf{pk} = h := g/f \in R_q$.

Encryption $\mathsf{Enc}_{\mathsf{NTRU}}(m, \mathsf{pk})$. Given message $m \in R_p$ and public key $\mathsf{pk} = h$:

1. Sample encryption randomness $s, e \leftarrow S_\nu$.
2. Return ciphertext $c = p \cdot (hs + e) + m \in R_q$.

Decryption $\mathsf{Dec}_{\mathsf{NTRU}}(c, \mathsf{sk})$. Given ciphertext c and secret key $\mathsf{sk} = f$:

1. Compute $m = (f \cdot c \mod q) \mod p$.
2. Return the plaintext message m.

Fig. 1. The encryption scheme `NTRUEncrypt` adapted from [SS13].

NTRU Encryption

NTRU ciphertexts consist of one ring element instead of two. We also wanted to decrease the dimension and moduli to reduce ciphertext sizes, but this was not possible based on current security analysis on ternary secrets.

We analysed the concrete security of NTRU for arbitrary standard deviations σ, and we found that the "fatigue point" for NTRU is $q = 0.0058 \cdot \sigma^2 \cdot d^{2.484}$.

We combined this with exact zero-knowledge proofs of boundedness to get tighter bounds and smaller parameters (but more expensive proofs).

NTRU Mixing Network [ePrint'23]

- Optimal parameters for the overall system is $q \approx 2^{59}$ and $N = 2^{11}$.

Scheme	Ciphertexts	Shuffle	Dist. Dec.	Total
CCS'23 [KB]	80	370	157	2188
NTRU [KB]	15	130	85	875
CCS'23 [ms]	0.74	261	138	1182
NTRU [ms]	0.20	62	328	576

Table: Per vote comparison of ciphertexts, shuffle proofs, decryption proofs, and overall with 4 servers. Shuffles are sequential, while decryption is run in parallel.

- *Lattice-Based Proof of Shuffle and Applications to Electronic Voting*, Diego F. Aranha, Carsten Baum, Kristian Gjøsteen, Tjerand Silde, and Thor Tunge. Published at CT-RSA 2021, eprint.iacr.org/2023/1318

- *Verifiable Mix-Nets and Distributed Decryption for Voting from Lattice-Based Assumptions*, Diego F. Aranha, Carsten Baum Kristian Gjøsteen, and Tjerand Silde. Published at ACM CCS 2023, eprint.iacr.org/2022/422

- *Concrete NTRU Security and Advances in Practical Lattice-Based Electronic Voting*, Patrick Hough, Caroline Sandsbråten, and Tjerand Silde. Available at IACR ePrint 2023/993, eprint.iacr.org/2023/933

Thank you! Questions?