Leakage-Resilient Non-Malleable Secret Sharing in Non-Compartmentalised Models

Fuchun Lin 

M. Cheraghchi  V. Guruswami  R. Safavi-Naini  H. Wang 

WMTC, June 13-14, 2019

Outline

1. Background
   - Non-Malleable Code
   - Non-Malleable Secret Sharing
2. Non-Compartmentalised LR-SS
   - Defining Non-Compartmentalised LR-SS
   - Constructing Non-Compartmentalised LR-SS
3. Non-Compartmentalised LR-NM-SS
   - Defining affine LR-NM-SS
   - Constructing affine LR-NM-SS
4. Conclusion

Since this is a short talk, most time will be spent on Item 1 :)

Bit-wise Independent Tampering

There are $2^2 = 4$ functions $\{0,1\} \to \{0,1\}$: $\mathsf{Set0}, \mathsf{Set1}, \mathsf{Keep}, \mathsf{Flip}$.

$$\mathsf{Set0}(b) = 0,\ \mathsf{Set1}(b) = 1,\ \mathsf{Keep}(b) = b,\ \mathsf{Flip}(b) = b + 1$$

There are $(2^N)^{2^N}$ functions $\{0,1\}^N \to \{0,1\}^N$. Consider the following tiny subset $\mathcal{F}_{\mathsf{BIT}}$ of cardinality $|\mathcal{F}_{\mathsf{BIT}}| = 4^N$.

$$f \in \mathcal{F}_{\mathsf{BIT}} : f = (f_1, \ldots, f_N),\ \text{where } f_i \in \{\mathsf{Set0}, \mathsf{Set1}, \mathsf{Keep}, \mathsf{Flip}\}$$

In particular,

- $f = (\mathsf{Flip}, \ldots, \mathsf{Flip})$ means flip every single bit of the vector
- $f = (\mathsf{Set1}, \ldots, \mathsf{Set1})$ means overwrite with the all-one vector

Intuition of Non-Malleability

Some observations and a question concerning coding for $\mathcal{F}_{\mathsf{BIT}}$:

- It is impossible to correct the error caused by all $f \in \mathcal{F}_{\mathsf{BIT}}$
- It is impossible to detect the error caused by all $f \in \mathcal{F}_{\mathsf{BIT}}$
- Is there a meaningful guarantee achievable through coding?

Real life example of bidding.

$$\text{Honest bidder} \xrightarrow{\mathsf{c}} \text{Receiver}$$

$$\text{Competitor} \xrightarrow{f(\mathsf{c}), f \in \mathcal{F}_{\mathsf{BIT}}} \text{Receiver}$$

The Competitor wins maximally if he/she is able to bid one dollar higher than the Honest bidder.

NM: Allow message tampering (even completely overwrite), but want to prevent the tampering from being message-specific.

Definition

We need randomised codes with probabilistic Enc and deterministic Dec (a.k.a. coding schemes).

Definition 1 ([DPW18])

Let $\mathcal{F}$ be a family of tampering functions. For each $f \in \mathcal{F}$ and $\mathsf{m} \in \{0,1\}^k$, define the tampering-experiment

$$\mathrm{Tamper}_{\mathsf{m}}^{f} = \left\{ \begin{array}{c} \mathsf{x} \leftarrow \mathsf{Enc}(\mathsf{m}), \tilde{\mathsf{x}} = f(\mathsf{x}), \tilde{\mathsf{m}} = \mathsf{Dec}(\tilde{\mathsf{x}}) \\ \mathrm{Output}\ \tilde{\mathsf{m}}, \end{array} \right\}.$$

which is a random variable over the randomness of the encoding function Enc. A coding scheme (Enc, Dec) is *non-malleable with respect to* $\mathcal{F}$ if for each $f \in \mathcal{F}$, there exists a distribution $\mathcal{D}_f$ over the set $\{0,1\}^k \bigcup \{\perp, \mathsf{same}^*\}$, such that, for all $\mathsf{m} \in \{0,1\}^k$, we have:

$$\mathrm{Tamper}_{\mathsf{m}}^{f} \overset{\varepsilon}{\approx} \left\{ \begin{array}{c} \tilde{\mathsf{m}} \leftarrow \mathcal{D}_f \\ \mathrm{Output}\ \mathsf{m}\ \mathrm{if}\ \tilde{\mathsf{m}} = \mathsf{same}^*,\ \mathrm{and}\ \tilde{\mathsf{m}}\ \mathrm{otherwise}; \end{array} \right\}$$

and $\mathcal{D}_f$ is efficiently samplable given oracle access to $f(\cdot)$.

Note that dependence on f is unavoidable, for example, when f completely overwrite.

A sufficient condition: $\mathsf{Dec}(f(\mathsf{Enc}(\mathsf{m}^0))) \overset{\varepsilon}{\approx} \mathsf{Dec}(f(\mathsf{Enc}(\mathsf{m}^1)))$.

A General Construction Approach

- The most studied NMC model is the $\mathcal{F}_{C-\text{split}}$, for a small constant C.

  $$f \in \mathcal{F}_{C-\text{split}} : f = (f_1, \ldots, f_C), \text{ where } f_i : \{0,1\}^{N/C} \to \{0,1\}^{N/C}$$

  The most difficult case is when $C = 2$, which leads to first instances of NM-SS and LR-NM-SS (see Page 9).
- [CG17] proposed a general approach: a weaker sufficient condition than $\mathrm{Dec}(f(\mathrm{Enc}(\mathrm{m}^0))) \overset{\varepsilon}{\sim} \mathrm{Dec}(f(\mathrm{Enc}(\mathrm{m}^1)))$:

  $$(\mathrm{Dec}(\mathrm{Enc}(\mathrm{U}_k)), \mathrm{Dec}(f(\mathrm{Enc}(\mathrm{U}_k)))) \overset{\varepsilon}{\sim} (\mathrm{U}_k, \mathrm{Dec}(f(\mathrm{Enc}(\mathrm{U}_k))))$$

  Intuition: assume we encode a uniform message U_k instead of a particular $\mathrm{m} \in \{0,1\}^k$. Suppose $\mathrm{Enc}(\mathrm{U}_k) = \mathrm{U}_N$, it becomes

  $$(\mathrm{Dec}(\mathrm{U}_N), \mathrm{Dec}(f(\mathrm{U}_N))) \overset{\varepsilon}{\sim} (\mathrm{U}_k, \mathrm{Dec}(f(\mathrm{U}_N)))$$

Non-Compartmentalised Tampering

- Non-compartmentalized tampering model was first studied by [AGM+15] for non-malleability against permutation composed with $\mathcal{F}_{\mathsf{BIT}}$, and shown useful in constructing non-malleable string commitments.
- There are a few other non-compartmentalized tampering families studied for non-malleable codes: local functions [CKR16], affine functions $\mathcal{F}_{\mathsf{affine}}$ [CL17], small-depth circuits [BDG+18] and decision tree [BGW19].
- In particular, the non-compartmentalised tampering $\mathcal{F}_{\mathsf{affine}}$ can be handled using the general approach of [CG17].

$$(\mathsf{Dec}(\mathsf{U}_N), \mathsf{Dec}(f(\mathsf{U}_N))) \overset{\varepsilon}{\sim} (\mathsf{U}_k, \mathsf{Dec}(f(\mathsf{U}_N)))$$

Threshold Secret Sharing

Threshold secret sharing ($(t+1)$-out-of-n)

- Correctness: reconstruct the secret given any $t+1$ shares
- Privacy: distribution of any t shares is independent of secret

$$\mathsf{Share}(\mathsf{m}^0)_{t/n} \overset{\varepsilon}{\sim} \mathsf{Share}(\mathsf{m}^1)_{t/n}$$

[LCG+19] proposed a binary SS construction: $\mathsf{ECC}(\mathsf{Enc}(\cdot))$

$$(\mathsf{Dec}(\mathsf{Enc}(\mathsf{U}_k)), \mathsf{ECC}(\mathsf{Enc}(\mathsf{U}_k))_{t/n}) \overset{\varepsilon}{\sim} (\mathsf{U}_k, \mathsf{ECC}(\mathsf{Enc}(\mathsf{U}_k))_{t/n}),$$

where $\mathsf{Enc} : \{0,1\}^k \to \{0,1\}^K$ and $\mathsf{ECC} : \{0,1\}^K \to \{0,1\}^N$.
Intuition: assume we encode a uniform message U_k instead of a particular $\mathsf{m} \in \{0,1\}^k$. Suppose $\mathsf{Enc}(\mathsf{U}_k) = \mathsf{U}_K$, it becomes

$$(\mathsf{Dec}(\mathsf{U}_K), \mathsf{ECC}(\mathsf{U}_K)_{t/n}) \overset{\varepsilon}{\sim} (\mathsf{U}_k, \mathsf{ECC}(\mathsf{U}_K)_{t/n}),$$

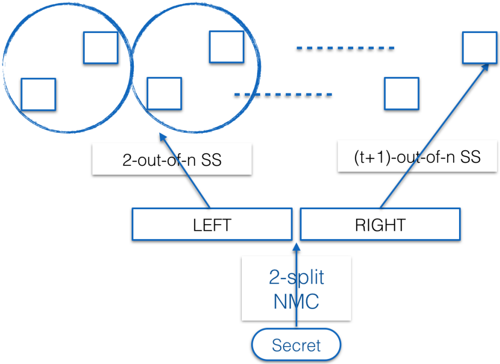
From 2-Split State NMC to NM-SS

[ADKO15] Non-malleability in 2-split state model implies privacy as a 2-out-of-2 secret sharing. One then has a 2-out-of-2 secret sharing that is also non-malleable with respect to $\mathcal{F}_{2-\text{split}}$.

[GK18a] defined and constructed $(t+1)$-out-of-n NM-SS with respect to $\mathcal{F}_{n-\text{split}}$.

- Correctness + NM: is more settled (still has some variations in so called *continuous tampering* models)
- Privacy + NM: (1) separately satisfied; (2) $t+1$ shares divided into two groups ...

Construction of [GK18a] and Renewed Interest in LR-SS



To make the idea work, the 2-out-of-n SS should be an LR-SS to facilitate the independence of the two states.

LR-SS, NM-SS, LR-NM-SS

Table 1 : List of papers on LR-SS, NM-SS, LR-NM-SS for $n > 2$ players

Reference	Acc. Stru.	LR-SS	NM-SS	LR-NM-SS	N-Comp.
[DP07]		N-adap. Ind. L.			No
[BDIR18]	r-out-of-n	N-adap. Ind. L.			No
[GK18a]	2-out-of-n r-out-of-n	N-adap. Ind. L.	Ind. T., Joint T.		No No
[GK18b]	Arbitrary n-out-of-n		Ind. T. Joint T.		No No
[BS18]	4-monotone		Continuous Ind. T.		No
[ADN+18]	Arbitrary 3-monotone	N-adap. Ind. L.	Continuous Ind. T. N-adap. conc. recon.		No No
[SV18]	r-out-of-n 4-monotone	N-adap. Ind. L.	Ind. T.		No No
[KMS18]	Arbitrary	Adap. Joint L.	Ind. T.	T.←Ind. L.	No
[FV19]*	Arbitrary	Ind. noisy L.	Continuous Ind. T. Adap. conc. recon.	T.←Ind. noisy L.	No
This work	r-out-of-n	N-adap. Affine L.			Yes
	r-out-of-n	Adap. Affine L.			Yes
	r-out-of-n	Adap. Affine L.	Bit-wise Ind. T.	T.← Affine L.$^{(r-1)/P}$	Yes
	r-out-of-n	Adap. Affine L.	Affine T.	T.← Affine L.$^{(r-1)/P}$	Yes

Leakage-Resilient Storage (LRS) [DDV10]

Consider the following game between adversary $\mathcal{A}$ and oracle $\mathcal{O}$.

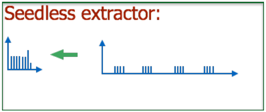
1. The adversary $\mathcal{A}$ chooses a pair of messages $\mathrm{m}_0, \mathrm{m}_1 \in \{0,1\}^\ell$ and sends them to the oracle $\mathcal{O}$.
2. The oracle $\mathcal{O}$ chooses a random bit $b \in \{0,1\}$ and compute $\mathrm{Enc}(\mathrm{m}_b)$.
3. The following is executed θ times, for $i = 1, \ldots, \theta$:
    1. $\mathcal{A}$ selects a function $l_i\colon \{0,1\}^N \to \{0,1\}^{c_i}$ from a set $\mathcal{L}$ of leakage functions, and sends it to $\mathcal{O}$,
    2. $\mathcal{O}$ sends $l_i(\mathrm{Enc}(\mathrm{m}_b))$ to $\mathcal{A}$. This is called $\mathcal{A}$ retrieves c_i bits through $\mathcal{L}$-leakage.

$\mathcal{A}$ is called β-bounded $\mathcal{L}$-leakage adversary if $\sum_{i=1}^{\theta} c_i \leq \beta$.

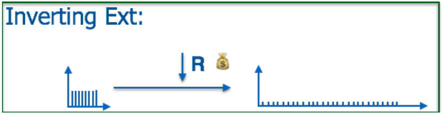
We consider structured Non-Compartmentalised $\mathcal{L}$, such as $\mathcal{L}_{\mathrm{affine}}$.

Affine Leakage-Resilient Secret Sharing

Randomness Extractors



&



Recall that the weaker condition proposed by [LCG$^+$19]

$$(\mathsf{Dec}(\mathsf{U}_K), \mathsf{ECC}(\mathsf{U}_K)_{t/n}) \overset{\varepsilon}{\sim} (\mathsf{U}_k, \mathsf{ECC}(\mathsf{U}_K)_{t/n}),$$

where $\mathsf{Enc} : \{0,1\}^k \to \{0,1\}^K$ and $\mathsf{ECC} : \{0,1\}^K \to \{0,1\}^N$.
If we use an affine extractor $\mathsf{Dec}(\cdot) = \mathsf{aExt}(\cdot)$ and an affine ECC, then the obtained secret sharing is LR-SS with respect to $\mathcal{L}_{\mathsf{affine}}$.

$$(\mathsf{Dec}(\mathsf{U}_K), \underline{\mathsf{ECC}(\mathsf{U}_K)_{t/n}, l_\beta(\mathsf{aECC}(\mathsf{U}_K))})$$
$$\overset{\varepsilon}{\sim} (\mathsf{U}_k, \underline{\mathsf{ECC}(\mathsf{U}_K)_{t/n}, l_\beta(\mathsf{aECC}(\mathsf{U}_K))})$$

Near optimal affine LR-SS

Depending on using a *seeded* or *seedless* extractor, we construct affine LR-SS against adaptive and non-adaptive adversaries.

- non-adaptive adversary: with secret length ℓ and information ratio $\frac{\ell+\beta+o(\ell)}{\ell}$
- adaptive adversary: a new construction of invertible $\mathrm{aExt}(\cdot)$ for LR-SS (of independent interest, substantially improves the lower bound of binary SS in [LCG^{+}19])

Note that $\frac{\ell+\beta+o(\ell)}{\ell}$ is almost the best one can achieve. Intuitively, any $t+1$ shares contain the full information about the ℓ bits secret, while t shares among them do not contain any information. Now there are β bits information about these $t+1$ shares leaked to an unconditional adversary. An information ratio of $\frac{\ell+\beta}{\ell}$ would be optimal.

Defining affine LR-NM-SS

- We consider a general tampering family $\mathcal{F}$
- The tampering adversary choose $f \in \mathcal{F}$ based on the leakage and any unauthorised set of shares

Previous LR-NM-SS only consider $\mathcal{F} = \mathcal{F}_{n\text{-split}}$ and the tampering adversary choose $f \in \mathcal{F}$ based on the leakage only.

High Level Idea for affine LR-NM-SS

Recall that

- Non-malleability $\Longleftarrow$ $\mathsf{Dec}(f(\mathsf{Enc}(\mathsf{m}^0))) \overset{\varepsilon}{\sim} \mathsf{Dec}(f(\mathsf{Enc}(\mathsf{m}^1)))$. [CG17] proposed a weaker sufficient condition:

$$(\mathsf{Dec}(\mathsf{U}_N), \mathsf{Dec}(f(\mathsf{U}_N))) \overset{\varepsilon}{\sim} (\mathsf{U}_k, \mathsf{Dec}(f(\mathsf{U}_N)))$$

- According to previous section: affine LR-SS $\Longleftarrow$

$$(\mathsf{Dec}(\mathsf{U}_K), \underline{\mathsf{ECC}(\mathsf{U}_K)_{t/n}, l_\beta(\mathsf{aECC}(\mathsf{U}_K))}) \overset{\varepsilon}{\sim} (\mathsf{U}_k, \underline{\mathsf{ECC}(\mathsf{U}_K)_{t/n}, l_\beta(\mathsf{aECC}(\mathsf{U}_K))})$$

Putting two things together:

$$(\mathsf{Dec}(\mathsf{U}_K), \underline{\mathsf{ECC}(\mathsf{U}_K)_{t/n}, l_\beta(\mathsf{aECC}(\mathsf{U}_K))}, \mathsf{Dec}(f(\mathsf{aECC}(\mathsf{U}_K)))) \overset{\varepsilon}{\sim} (\mathsf{U}_k, \underline{\mathsf{ECC}(\mathsf{U}_K)_{t/n}, l_\beta(\mathsf{aECC}(\mathsf{U}_K))}, \mathsf{Dec}(f(\mathsf{aECC}(\mathsf{U}_K))))$$

Seedless & Seeded Non-Malleable Extractors

Definition 2 ([CG17])

A function nmExt: $\{0,1\}^n \to \{0,1\}^m$ is a (k,ε)-seedless non-malleable extractor with respect to a class $\mathcal{X}$ of sources over $\{0,1\}^n$ and a class $\mathcal{F}$ of tampering functions acting on $\{0,1\}^n$, if for every $\mathsf{X} \in \mathcal{X}$ with min-entropy k and every $f \in \mathcal{F}$, there is a distribution $\mathcal{D}_f$ over $\{0,1\}^m \cup \{\mathsf{same}^*\}$ such that for an independent Y sampled from $\mathcal{D}_f$, we have

$$\mathsf{SD}(\mathsf{nmExt}(\mathsf{X}), \mathsf{nmExt}(f(\mathsf{X})); \mathsf{U_m}, \mathsf{Copy}(\mathsf{Y}, \mathsf{U}_m)) \leq \varepsilon,$$

where the two copies of U_m denote the same random variable and $\mathsf{Copy}(\mathsf{y},\mathsf{u}) = \mathsf{y}$ always except when $\mathsf{y} = \mathsf{same}^*$, in which case it outputs u.

Definition 3 ([DW09])

A seeded (k,ε)-non-malleable extractor is a function $\mathsf{nmExt} : \{0,1\}^d \times \{0,1\}^n \to \{0,1\}^m$ such that given any (n,k)-source X, an independent uniform seed $\mathsf{Z} \in \{0,1\}^d$, for any (deterministic) function $\mathcal{A} : \{0,1\}^d \to \{0,1\}^d$ such that $\mathcal{A}(\mathsf{z}) \neq \mathsf{z}$ for any z, we have

$$\mathsf{SD}(\underline{\mathsf{Z}, \mathsf{nmExt}(\mathsf{Z},\mathsf{X})}, \mathsf{nmExt}(\mathcal{A}(\mathsf{Z}),\mathsf{X}); \underline{\mathsf{Z}, \mathsf{U_m}}, \mathsf{nmExt}(\mathcal{A}(\mathsf{Z}),\mathsf{X})) \leq \varepsilon.$$

Summary of Results

Table 2 : List of papers on LR-SS, NM-SS, LR-NM-SS for $n > 2$ players

Reference	Access Structure	Design Goal	Leakage/Tampering Model
[DP07]	Round complexity based	LR-SS	Independent Leakage (Ind. L.)
[BDIR18]	r-out-of-n	LR-SS	Ind. L.
[GK18a]	2-out-of-n r-out-of-n r-out-of-n	LR-SS NM-SS NM-SS	Ind. L. Independent Tampering (Ind. T.) Joint Tampering (Joint T.)
[GK18b]	Arbitrary n-out-of-n	NM-SS NM-SS	Ind. T. Joint T.
[BS18]	Arbitrary (4-monotone)	CNM-SS	Continuous Ind. T. (CNM-SS)
[ADN+18]	Arbitrary Arbitrary (3-monotone)	LR-SS CNM-SS	Ind. L. N-adap. concurrent reconstruct
[SV18]	r-out-of-n Arbitrary (4-monotone)	LR-SS NM-SS	Ind. L. $\leftarrow r - 2$ shares Ind. T.
[KMS18]	Arbitrary Arbitrary	CLR-SS LR-NM-SS	Continuous adap. Joint Leakage Ind. T. $\leftarrow$Ind. L.
[FV19]*	Arbitrary	LR-CNM-SS	Ind. noisy L. Adap. concurrent reconstruct
This work	r-out-of-n r-out-of-n r-out-of-n	LR-SS LR-NM-SS LR-NM-SS	Affine L. ——first NComp. L. Bit-wise Ind. T. $\leftarrow$ Affine L. NComp.T.$\leftarrow$Affine L. —first NComp.T.

Divesh Aggarwal, Stefan Dziembowski, Tomasz Kazana, and Maciej Obremski.
Leakage-resilient non-malleable codes.
In *Theory of Cryptography Conference, TCC 2015*, pages 398–426, 2015.

Divesh Aggarwal, Ivan Damgard, Jesper Buus Nielsen, Maciej Obremski, Erick Purwanto, Joao Ribeiro, and Mark Simkin.
Stronger leakage-resilient and non-malleable secret-sharing schemes for general access structures.
IACR Cryptology ePrint Archive, page https://eprint.iacr.org/2018/1147, 2018.

Shashank Agrawal, Divya Gupta, Hemanta K. Maji, Omkant Pandey, and Manoj Prabhakaran.
Explicit non-malleable codes against bit-wise tampering and permutations.

In *Advances in Cryptology - CRYPTO 2015*, pages 538–557, 2015.

Marshall Ball, Dana Dachman-Soled, Siyao Guo, Tal Malkin, and Li-Yang Tan.
Non-malleable codes for small-depth circuits.
In *IEEE Annual Symposium on Foundations of Computer Science, FOCS*, pages 826–837, 2018.

Fabrice Benhamouda, Akshay Degwekar, Yuval Ishai, and Tal Rabin.
On the local leakage resilience of linear secret sharing schemes.

In *Advances in Cryptology - CRYPTO 2018*, pages 531–561, 2018.

Marshall Ball, Siyao Guo, and Daniel Wichs.
Non-malleable codes for decision trees.
page https://eprint.iacr.org/2019/379, 2019.

Saikrishna Badrinarayanan and Akshayaram Srinivasan.
Revisiting non-malleable secret sharing.
IACR Cryptology ePrint Archive, page
https://eprint.iacr.org/2018/1144, 2018.

Mahdi Cheraghchi and Venkatesan Guruswami.
Non-malleable coding against bit-wise and split-state tampering.
J. Cryptology, 30(1):191–241, 2017.

Nishanth Chandran, Bhavana Kanukurthi, and Srinivasan Raghuraman.
Information-theoretic local non-malleable codes and their applications.
In *Theory of Cryptography - TCC*, pages 367–392, 2016.

Eshan Chattopadhyay and Xin Li.
Non-malleable codes and extractors for small-depth circuits, and affine functions.

In *ACM SIGACT Symposium on Theory of Computing, STOC 2017*, pages 1171–1184, 2017.

Francesco Davì, Stefan Dziembowski, and Daniele Venturi.
Leakage-resilient storage.
In *Security and Cryptography for Networks SCN*, pages 121–137, 2010.

Stefan Dziembowski and Krzysztof Pietrzak.
Intrusion-resilient secret sharing.
In *Foundations of Computer Science FOCS 2007*, pages 227–237, 2007.

Stefan Dziembowski, Krzysztof Pietrzak, and Daniel Wichs.
Non-malleable codes.
J. ACM, 65(4):20:1–20:32, 2018.

Yevgeniy Dodis and Daniel Wichs.
Non-malleable extractors and symmetric key cryptography from weak secrets.

In *ACM Symposium on Theory of Computing, STOC 2009*, pages 601–610, 2009.

Antonio Faonio and Daniele Venturi.
Non-malleable secret sharing in the computational setting: Adaptive tampering, noisy-leakage resilience, and improved rate.
IACR Cryptology ePrint Archive, page https://eprint.iacr.org/2019/105, 2019.

Vipul Goyal and Ashutosh Kumar.
Non-malleable secret sharing.
In *ACM SIGACT Symposium on Theory of Computing, STOC 2018*, pages 685–698, 2018.

Vipul Goyal and Ashutosh Kumar.
Non-malleable secret sharing for general access structures.
In *Advances in Cryptology - CRYPTO 2018*, pages 501–530, 2018.

Ashutosh Kumar, Raghu Meka, and Amit Sahai.
Leakage-resilient secret sharing.
IACR Cryptology ePrint Archive, page
https://eprint.iacr.org/2018/1138, 2018.

Fuchun Lin, Mahdi Cheraghchi, Venkatesan Guruswami, Reihaneh Safavi-Naini, and Huaxiong Wang.
Secret sharing with binary shares.
In *Innovations in Theoretical Computer Science Conference, ITCS 2019*, pages 53:1–53:20, 2019.

Akshayaram Srinivasan and Prashant Nalini Vasudevan.
Leakage resilient secret sharing and applications.
IACR Cryptology ePrint Archive, page
https://eprint.iacr.org/2018/1154, 2018.