The Legendre Pseudorandom Function as a Multivariate Quadratic Cryptosystem

Security and Applications

István András Seres[1], Máté Horváth[2] and Péter Burcsi[1]

[1]Eötvös Loránd University, [2]Budapest University of Technology and Economics

2021 February 26

https://eprint.iacr.org/2021/182.pdf

Table of Contents

1. Preliminaries
   - The dawn of a new era of symmetric key primitives
   - Mathematical and cryptographic background
2. The Legendre PRF as MQ
   - The underdetermined case
   - The overdetermined case
3. Security of the Legendre PRF
   - Direct algebraic attacks
   - Interpolation attacks
   - Rank attacks
   - A security argument from algebraic geometry
4. Applications
   - The Legendre Verifiable random function
   - An OPPRF from the Legendre PRF
5. Future Directions

Table of Contents

1. Preliminaries
   - The dawn of a new era of symmetric key primitives
   - Mathematical and cryptographic background
2. The Legendre PRF as MQ
   - The underdetermined case
   - The overdetermined case
3. Security of the Legendre PRF
   - Direct algebraic attacks
   - Interpolation attacks
   - Rank attacks
   - A security argument from algebraic geometry
4. Applications
   - The Legendre Verifiable random function
   - An OPPRF from the Legendre PRF
5. Future Directions

Dominating trends in cryptography

- "Software is eating the world" (Marc Andreessen in Wall Street Journal (2011))

Dominating trends in cryptography

- "Software is eating the world" (Marc Andreessen in Wall Street Journal (2011))
- Zero-knowledge proofs (ZKPs) and secure multi-party computation (MPC) protocols are eating the crypto-world!

Dominating trends in cryptography

- "Software is eating the world" (Marc Andreessen in Wall Street Journal (2011))
- Zero-knowledge proofs (ZKPs) and secure multi-party computation (MPC) protocols are eating the crypto-world!
- Traditional symmetric key primitives have large multiplicative complexity, e.g. AES, SHA-3 etc.

Dominating trends in cryptography

- "Software is eating the world" (Marc Andreessen in Wall Street Journal (2011))
- Zero-knowledge proofs (ZKPs) and secure multi-party computation (MPC) protocols are eating the crypto-world!
- Traditional symmetric key primitives have large multiplicative complexity, e.g. AES, SHA-3 etc.
- Large multiplicative complexity causes enormous overhead in ZKPs and MPC protocols.

Dominating trends in cryptography

- "Software is eating the world" (Marc Andreessen in Wall Street Journal (2011))
- Zero-knowledge proofs (ZKPs) and secure multi-party computation (MPC) protocols are eating the crypto-world!
- Traditional symmetric key primitives have large multiplicative complexity, e.g. AES, SHA-3 etc.
- Large multiplicative complexity causes enormous overhead in ZKPs and MPC protocols.
- Surge of new hash-function designs aiming for low multiplicative complexity, e.g. MiMC [AGR+16], Poseidon [GKR+20], Marvellous, Jarvis, Friday [AD18].

Dominating trends in cryptography

- "Software is eating the world" (Marc Andreessen in Wall Street Journal (2011))
- Zero-knowledge proofs (ZKPs) and secure multi-party computation (MPC) protocols are eating the crypto-world!
- Traditional symmetric key primitives have large multiplicative complexity, e.g. AES, SHA-3 etc.
- Large multiplicative complexity causes enormous overhead in ZKPs and MPC protocols.
- Surge of new hash-function designs aiming for low multiplicative complexity, e.g. MiMC [AGR+16], Poseidon [GKR+20], Marvellous, Jarvis, Friday [AD18].
- Their cryptanalysis is still an active and ongoing research! For instance, see [ACG+19, LP19].

The Legendre Symbol and PRF

Definition (Legendre Symbol)

Let p be an odd prime. The Legendre Symbol of a and p is

$$\left(\frac{a}{p}\right) := \begin{cases} 1, & a \ has \ modular \ square - roots \mod p \\ -1, & a \ has \ no \ modular \ square - roots \mod p \\ 0, & if \ a \equiv 0 \mod p \end{cases}$$

The Legendre Symbol and PRF

Definition (Legendre Symbol)

Let p be an odd prime. The Legendre Symbol of a and p is

$$\left(\frac{a}{p}\right) := \begin{cases} 1, & a\ has\ modular\ square-roots \mod p \\ -1, & a\ has\ no\ modular\ square-roots \mod p \\ 0, & if\ a \equiv 0 \mod p \end{cases}$$

Definition (Sequential Legendre PRF)

Damgård proposed using the sequence of consecutive Legendre symbols with respect to a large prime p as a pseudorandom generator [Dam88]. Let $\{a\}_K$ denote the following sequence.

$$\{a\}_K := \left(\frac{K}{p}\right), \left(\frac{K+1}{p}\right), \ldots, \left(\frac{K+a-1}{p}\right)$$

Pseudo-randomness of the Legendre PRF

There is a vast literature on asserting the high level of pseudo-randomness of quadratic (and higher order) characters.

- Pólya-Vinogradov inequality for character sums. Consecutive values of $\left(\frac{a}{p}\right)$ mimic a random variable i.e.

$$\sum_{a=M+1}^{M+N} \left(\frac{a}{p}\right) \leq \sqrt{p} \log p$$

Pseudo-randomness of the Legendre PRF

There is a vast literature on asserting the high level of pseudo-randomness of quadratic (and higher order) characters.

- Pólya-Vinogradov inequality for character sums. Consecutive values of $\left(\frac{a}{p}\right)$ mimic a random variable i.e.
$$\sum_{a=M+1}^{M+N} \left(\frac{a}{p}\right) \leq \sqrt{p}\log p$$
- Peralta: n-grams are asymptotically equally distributed [Per92]

Pseudo-randomness of the Legendre PRF

There is a vast literature on asserting the high level of pseudo-randomness of quadratic (and higher order) characters.

- Pólya-Vinogradov inequality for character sums. Consecutive values of $\left(\frac{a}{p}\right)$ mimic a random variable i.e.
$$\sum_{a=M+1}^{M+N} \left(\frac{a}{p}\right) \leq \sqrt{p} \log p$$
- Peralta: n-grams are asymptotically equally distributed [Per92]
- Mauduit&Sárközy: "Legendre symbol sequences are the most natural candidate for pseudorandomness" [MS97]

Pseudo-randomness of the Legendre PRF

There is a vast literature on asserting the high level of pseudo-randomness of quadratic (and higher order) characters.

- Pólya-Vinogradov inequality for character sums. Consecutive values of $\left(\frac{a}{p}\right)$ mimic a random variable i.e.
$$\sum_{a=M+1}^{M+N} \left(\frac{a}{p}\right) \leq \sqrt{p} \log p$$
- Peralta: n-grams are asymptotically equally distributed [Per92]
- Mauduit&Sárközy: "Legendre symbol sequences are the most natural candidate for pseudorandomness" [MS97]
- Ding: high linear complexity of the Legendre symbol [DHS98]

Pseudo-randomness of the Legendre PRF

There is a vast literature on asserting the high level of pseudo-randomness of quadratic (and higher order) characters.

- Pólya-Vinogradov inequality for character sums. Consecutive values of $\left(\frac{a}{p}\right)$ mimic a random variable i.e.
$$\sum_{a=M+1}^{M+N} \left(\frac{a}{p}\right) \leq \sqrt{p} \log p$$
- Peralta: n-grams are asymptotically equally distributed [Per92]
- Mauduit&Sárközy: "Legendre symbol sequences are the most natural candidate for pseudorandomness" [MS97]
- Ding: high linear complexity of the Legendre symbol [DHS98]
- Gyarmati&Mauduit&Sárközy: good cross correlation of the Legendre symbol sequences [GMS14]

Hard problems and cryptographic assumptions

Definition (Shifted Legendre Symbol Problem)

Let K be uniformly sampled from $\mathbb{F}_p$, and define $\mathcal{O}_{Leg}$ to be an oracle that takes $x \in \mathbb{F}_p$ and outputs $\left(\frac{K+x}{p}\right)$. Then the Shifted Legendre Symbol (SLS) problem is to find K given oracle access to $\mathcal{O}_{Leg}$ with non-negligible probability.

Hard problems and cryptographic assumptions

Definition (Shifted Legendre Symbol Problem)

Let K be uniformly sampled from $\mathbb{F}_p$, and define $\mathcal{O}_{Leg}$ to be an oracle that takes $x \in \mathbb{F}_p$ and outputs $\left(\frac{K+x}{p}\right)$. Then the Shifted Legendre Symbol (SLS) problem is to find K given oracle access to $\mathcal{O}_{Leg}$ with non-negligible probability.

Definition (Multivariate Quadratic (MQ) problem)

Given a random system of quadratic polynomials $\mathbf{f} = (f_1(x_1, \ldots, x_n), \ldots, f_m(x_1, \ldots, x_n)) \in \mathbb{F}[x_1, \ldots, x_n]^m$, find a common zero $\mathbf{x_0} \in \mathbb{F}^n$ of the polynomials $f_1, \ldots, f_m$.

State of the art Legendre PRF key-recovery attacks

- Russel and Shparlinski's attack runs in $\tilde{\mathcal{O}}(p)$

State of the art Legendre PRF key-recovery attacks

- Russel and Shparlinski's attack runs in $\tilde{\mathcal{O}}(p)$
- Birthday-bound attack in $\mathcal{O}(\sqrt{p}\log p)$ by Khovratovich [Kho19]

State of the art Legendre PRF key-recovery attacks

- Russel and Shparlinski's attack runs in $\tilde{\mathcal{O}}(p)$
- Birthday-bound attack in $\mathcal{O}(\sqrt{p}\log p)$ by Khovratovich [Kho19]
- Table-based collision search by Beullens et al. [BBUV20]

State of the art Legendre PRF key-recovery attacks

- Russel and Shparlinski's attack runs in $\tilde{\mathcal{O}}(p)$
- Birthday-bound attack in $\mathcal{O}(\sqrt{p}\log p)$ by Khovratovich [Kho19]
- Table-based collision search by Beullens et al. [BBUV20]
- Improved key-recovery attack in $\mathcal{O}(\sqrt{p\log\log p})$ by Kaluderovic et al. [KKK20].

State of the art Legendre PRF key-recovery attacks

- Russel and Shparlinski's attack runs in $\tilde{\mathcal{O}}(p)$
- Birthday-bound attack in $\mathcal{O}(\sqrt{p}\log p)$ by Khovratovich [Kho19]
- Table-based collision search by Beullens et al. [BBUV20]
- Improved key-recovery attack in $\mathcal{O}(\sqrt{p \log\log p})$ by Kaluderovic et al. [KKK20].

Relevant research directions:

- Could we have sub-exponential key-recovery attacks?

State of the art Legendre PRF key-recovery attacks

- Russel and Shparlinski's attack runs in $\tilde{\mathcal{O}}(p)$
- Birthday-bound attack in $\mathcal{O}(\sqrt{p}\log p)$ by Khovratovich [Kho19]
- Table-based collision search by Beullens et al. [BBUV20]
- Improved key-recovery attack in $\mathcal{O}(\sqrt{p\log\log p})$ by Kaluderovic et al. [KKK20].

Relevant research directions:

- Could we have sub-exponential key-recovery attacks?
- What about provable security of the Legendre PRF?

State of the art Legendre PRF key-recovery attacks

- Russel and Shparlinski's attack runs in $\tilde{\mathcal{O}}(p)$
- Birthday-bound attack in $\mathcal{O}(\sqrt{p}\log p)$ by Khovratovich [Kho19]
- Table-based collision search by Beullens et al. [BBUV20]
- Improved key-recovery attack in $\mathcal{O}(\sqrt{p \log \log p})$ by Kaluderovic et al. [KKK20].

Relevant research directions:

- Could we have sub-exponential key-recovery attacks?
- What about provable security of the Legendre PRF?
- Would it be possible to connect it to other cryptographic assumptions?

Table of Contents

1. Preliminaries
   - The dawn of a new era of symmetric key primitives
   - Mathematical and cryptographic background
2. The Legendre PRF as MQ
   - The underdetermined case
   - The overdetermined case
3. Security of the Legendre PRF
   - Direct algebraic attacks
   - Interpolation attacks
   - Rank attacks
   - A security argument from algebraic geometry
4. Applications
   - The Legendre Verifiable random function
   - An OPPRF from the Legendre PRF
5. Future Directions

Roadmap for relating the breakage of the Legendre PRF to solving an MQ problem

- Show that a successful Legendre PRF key-recovery attacker can solve an MQ instance

Roadmap for relating the breakage of the Legendre PRF to solving an MQ problem

- Show that a successful Legendre PRF key-recovery attacker can solve an MQ instance
- We aim to find as many independent equations as possible given the psuedo-random binary sequence!

Roadmap for relating the breakage of the Legendre PRF to solving an MQ problem

- Show that a successful Legendre PRF key-recovery attacker can solve an MQ instance
- We aim to find as many independent equations as possible given the psuedo-random binary sequence!
- Try to characterize the ideal of the resulting MQ instance!

Roadmap for relating the breakage of the Legendre PRF to solving an MQ problem

- Show that a successful Legendre PRF key-recovery attacker can solve an MQ instance
- We aim to find as many independent equations as possible given the psuedo-random binary sequence!
- Try to characterize the ideal of the resulting MQ instance!
- What is the group structure of the solutions of the MQ instance?

Roadmap for relating the breakage of the Legendre PRF to solving an MQ problem

- Show that a successful Legendre PRF key-recovery attacker can solve an MQ instance
- We aim to find as many independent equations as possible given the psuedo-random binary sequence!
- Try to characterize the ideal of the resulting MQ instance!
- What is the group structure of the solutions of the MQ instance?
- Examples

The MQ instance in the undetermined case

- Let's fix an aribtrary $r \in \mathbb{F}^*$ quadratic non-residue. Let $b_i := \left(\frac{K+i}{p}\right)$ and x_i be the corresponding unknown.

The MQ instance in the undetermined case

- Let's fix an aribtrary $r \in \mathbb{F}^*$ quadratic non-residue. Let $b_i := \left(\frac{K+i}{p}\right)$ and x_i be the corresponding unknown.
- Let x_i denote one of the square roots of $K+i$, if $b_i = 1$. Otherwise it denotes the square-root of $r(K+i)$.

The MQ instance in the undetermined case

- Let's fix an aribtrary $r \in \mathbb{F}^*$ quadratic non-residue. Let $b_i := \left(\frac{K+i}{p}\right)$ and x_i be the corresponding unknown.
- Let x_i denote one of the square roots of $K+i$, if $b_i = 1$. Otherwise it denotes the square-root of $r(K+i)$.
- Eeach consecutive pair of Legendre symbols add a new quadratic equation to the MQ problem.

The MQ instance in the undetermined case

- Let's fix an aribtrary $r \in \mathbb{F}^*$ quadratic non-residue. Let $b_i := \left(\frac{K+i}{p}\right)$ and x_i be the corresponding unknown.
- Let x_i denote one of the square roots of $K+i$, if $b_i = 1$. Otherwise it denotes the square-root of $r(K+i)$.
- Eeach consecutive pair of Legendre symbols add a new quadratic equation to the MQ problem.

Therefore we have the following four cases.

The MQ instance in the undetermined case

- Let's fix an aribtrary $r \in \mathbb{F}^*$ quadratic non-residue. Let $b_i := \left(\frac{K+i}{p}\right)$ and x_i be the corresponding unknown.
- Let x_i denote one of the square roots of $K+i$, if $b_i = 1$. Otherwise it denotes the square-root of $r(K+i)$.
- Eeach consecutive pair of Legendre symbols add a new quadratic equation to the MQ problem.

Therefore we have the following four cases.If $b_i = b_{i+1} = 1$, then we know that $x_{i+1}^2 = K + i + 1$ and $x_i^2 = K + i$, hence

$$x_{i+1}^2 - x_i^2 = 1.$$

The MQ instance in the undetermined case (contd.)

If $b_i = b_{i+1} = -1$, then we have that $x_{i+1}^2 = r(K+i+1)$ and $x_i^2 = r(K+i)$, hence

The MQ instance in the undetermined case (contd.)

If $b_i = b_{i+1} = -1$, then we have that $x_{i+1}^2 = r(K + i + 1)$ and $x_i^2 = r(K + i)$, hence

$$x_{i+1}^2 - x_i^2 = r.$$

Finally if $b_i = 1 = -b_{i+1}$ or $b_i = -1 = -b_{i+1}$ then we obtain the following two quadratic equations:

The MQ instance in the undetermined case (contd.)

If $b_i = b_{i+1} = -1$, then we have that $x_{i+1}^2 = r(K+i+1)$ and $x_i^2 = r(K+i)$, hence

$$x_{i+1}^2 - x_i^2 = r.$$

Finally if $b_i = 1 = -b_{i+1}$ or $b_i = -1 = -b_{i+1}$ then we obtain the following two quadratic equations:

$$x_{i+1}^2 - r x_i^2 = r, \qquad x_{i+1}^2 - r^{-1} x_i^2 = 1.$$

The MQ instance in the undetermined case (contd.)

If $b_i = b_{i+1} = -1$, then we have that $x_{i+1}^2 = r(K+i+1)$ and $x_i^2 = r(K+i)$, hence

$$x_{i+1}^2 - x_i^2 = r.$$

Finally if $b_i = 1 = -b_{i+1}$ or $b_i = -1 = -b_{i+1}$ then we obtain the following two quadratic equations:

$$x_{i+1}^2 - r x_i^2 = r, \qquad x_{i+1}^2 - r^{-1} x_i^2 = 1.$$

Remarks:

- Sparse MQ instance without linear terms
- Very peculiar polynomial structure unlike regular MQ instances
- "Minimality" of the sparseness

Example

Let $p = \texttt{0xffffffffffffffffffffdd}$ and $K = \texttt{0x27aaa97c746c22e12d0f}$.

Example

Let $p =$ `0xffffffffffffffffdd` and $K =$ `0x27aaa97c746c22e12d0f`. The smallest quadratic non-residue $\mod p$ is 2.

Example

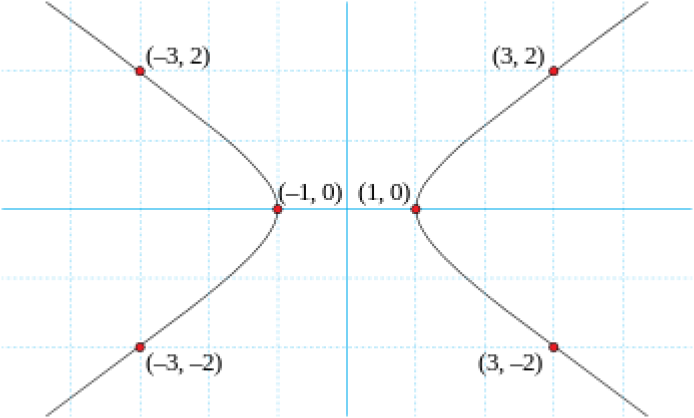
Let $p = \texttt{0xffffffffffffffffffdd}$ and $K = \texttt{0x27aaa97c746c22e12d0f}$.The smallest quadratic non-residue $\mod p$ is 2.We display the MQ instance induced by the evaluation of the linear Legendre PRF, $\{6\}_K = (1, 1, 1, -1, -1, 1)$.

Example

Let $p = \texttt{0xffffffffffffffffffdd}$ and $K = \texttt{0x27aaa97c746c22e12d0f}$.The smallest quadratic non-residue mod p is 2.We display the MQ instance induced by the evaluation of the linear Legendre PRF, $\{6\}_K = (1, 1, 1, -1, -1, 1)$.The complete MQ instance corresponding to $\{6\}_K$ has the following form:

$$\begin{aligned} x_1^2 - x_0^2 &= 1 \\ x_2^2 - x_1^2 &= 1 \\ x_3^2 - 2x_2^2 &= 2 \\ x_4^2 - x_3^2 &= 2 \\ 2x_5^2 - x_4^2 &= 2 \end{aligned}$$

A somewhat cripple analogy



$$x^2 - 2y^2 = 1$$

Gröbner-basis of the ideal

Theorem

Given $\{n\}_K = (b_0, \ldots, b_{n-1})$ and its corresponding ideal $I = \langle f_1, f_2, \ldots, f_m \rangle$, where $m = n - 1$ as defined by the equations from the previous slides. Its Gröbner basis consists of the polynomials g_i, for $i \in [0, n-2]$ such that,

$$g_i = \begin{cases} x_i^2 - x_{n-1}^2 + (n-i), \textit{if} \quad b_{n-1} = 1 \wedge b_i = 1 \\ x_i^2 - r x_{n-1}^2 + r(n-i), \textit{if} \quad b_{n-1} = 1 \wedge b_i = -1 \\ x_i^2 - r^{-1} x_{n-1}^2 + (n-i), \textit{if} \quad b_{n-1} = -1 \wedge b_i = 1 \\ x_i^2 - x_{n-1}^2 + r(n-i), \textit{if} \quad b_{n-1} = -1 \wedge b_i = -1 \end{cases} \tag{1}$$

Specifically, $I = \langle g_0, \ldots, g_{n-2} \rangle$ and $G := (g_i)_{i=0}^{n-2}$ is a reduced Gröbner-basis.

Example (contd.)

The Gröbner-basis of the previous example of $\{6\}_K$ consists of the following quadratic bi-variate polynomials:

$$x_0^2 - x_5^2 + 5$$
$$x_1^2 - x_5^2 + 4$$
$$x_2^2 - x_5^2 + 3$$
$$x_3^2 - 2x_5^2 + 4$$
$$x_4^2 - 2x_5^2 + 2$$

Adding new, independent polynomials to the MQ problem

Observe that in these cases, we can express the modular square root function $\mathrm{sqrt}_p : \mathbb{F}_p^* \to \mathbb{F}_p^*$ as a polynomial function as follows:

$$\mathrm{sqrt}_p(x) = \begin{cases} \pm x^{\frac{p+1}{4}} \mod p, & \text{if} \quad p \equiv 3 \mod 4 \\ \pm x(2x)^{\frac{p-5}{8}}(4x^{\frac{p-1}{4}} - 1) \mod p, & \text{if} \quad p \equiv 5 \mod 8 \end{cases}$$

Adding new, independent polynomials to the MQ problem

Observe that in these cases, we can express the modular square root function $\mathrm{sqrt}_p : \mathbb{F}_p^* \to \mathbb{F}_p^*$ as a polynomial function as follows:

$$\mathrm{sqrt}_p(x) = \begin{cases} \pm x^{\frac{p+1}{4}} \mod p, & \text{if} \quad p \equiv 3 \mod 4 \\ \pm x(2x)^{\frac{p-5}{8}}(4x^{\frac{p-1}{4}} - 1) \mod p, & \text{if} \quad p \equiv 5 \mod 8 \end{cases}$$

By this observation, we can obtain $\mathcal{O}(\log^2 p)$ new polynomials, one for each quadratic term $x_i x_j$:

$$x_i x_j = \mathrm{sqrt}_p(r^{L_0(x_i)+L_0(x_j)}(K+i)(K+j)).$$

Adding new, independent polynomials to the MQ problem

Observe that in these cases, we can express the modular square root function $\mathrm{sqrt}_p : \mathbb{F}_p^* \to \mathbb{F}_p^*$ as a polynomial function as follows:

$$\mathrm{sqrt}_p(x) = \begin{cases} \pm x^{\frac{p+1}{4}} \mod p, & \text{if} \quad p \equiv 3 \mod 4 \\ \pm x(2x)^{\frac{p-5}{8}}(4x^{\frac{p-1}{4}} - 1) \mod p, & \text{if} \quad p \equiv 5 \mod 8 \end{cases}$$

By this observation, we can obtain $\mathcal{O}(\log^2 p)$ new polynomials, one for each quadratic term $x_i x_j$:

$$x_i x_j = \mathrm{sqrt}_p(r^{L_0(x_i)+L_0(x_j)}(K+i)(K+j)).$$

In a similar fashion, we can add new polynomials involving the linear terms of the unknowns for every $i \neq j$:

$$x_i = \mathrm{sqrt}_p(r^{L_0(x_i)-L_0(x_j)}(x_j^2 - r^{L_0(x_j)}(j-i)))$$

Table of Contents

1 Preliminaries
- The dawn of a new era of symmetric key primitives
- Mathematical and cryptographic background

2 The Legendre PRF as MQ
- The underdetermined case
- The overdetermined case

3 Security of the Legendre PRF
- Direct algebraic attacks
- Interpolation attacks
- Rank attacks
- A security argument from algebraic geometry

4 Applications
- The Legendre Verifiable random function
- An OPPRF from the Legendre PRF

5 Future Directions

Solving directly the MQ problem

- So, why not just solve the problem directly with Gröbner basis?

Solving directly the MQ problem

- So, why not just solve the problem directly with Gröbner basis?
- In the undetermined case, the Gröbner-basis seemingly does not help at all! It seems that we can't do better than just brute-force.

Solving directly the MQ problem

- So, why not just solve the problem directly with Gröbner basis?
- In the undetermined case, the Gröbner-basis seemingly does not help at all! It seems that we can't do better than just brute-force.
- New polynomials in the overdetermined case doesn't help. The degree of regularity behaves just like in random MQ systems;

m	n	d_{reg} Random MQ	d_{reg} Legendre MQ
7	7	3	3
8	8	4	4
9	9	4	4
10	10	5	5
11	11	5	5

Interpolation attacks

- Goal: construct a cryptosystem's defining polynomial withouth knowing the secret key.

Interpolation attacks

- Goal: construct a cryptosystem's defining polynomial withouth knowing the secret key.
- If the resulting polynomial is low-degree, then we can factor it efficiently and one of the roots will be the secret key/plaintext/PRF seed.

Interpolation attacks

- Goal: construct a cryptosystem's defining polynomial withouth knowing the secret key.
- If the resulting polynomial is low-degree, then we can factor it efficiently and one of the roots will be the secret key/plaintext/PRF seed.
- What is the degree of the Legendre PRF as a univariate polynomial?

$$F_K(a) = \sum_{i=0}^{a-1} 2^{a-1-i}(K+i)^{\frac{p-1}{2}} \mod p$$

Interpolation attacks

- Goal: construct a cryptosystem's defining polynomial withouth knowing the secret key.
- If the resulting polynomial is low-degree, then we can factor it efficiently and one of the roots will be the secret key/plaintext/PRF seed.
- What is the degree of the Legendre PRF as a univariate polynomial?

$$F_K(a) = \sum_{i=0}^{a-1} 2^{a-1-i}(K+i)^{\frac{p-1}{2}} \mod p$$

Note that $deg(F_K(a)) = \frac{p-1}{2}$, i.e. the degree of the polynomial representing the Legendre PRF has almost full degree over $\mathbb{F}_p$, that is exponential in the security parameter.

The MinRank attack

We rewrite each generator polynomial f_i in the ideal $I = \langle f_1, \ldots, f_m \rangle$ induced by the Legendre PRF, as folllows:

$$f_i(x_1, \ldots, x_n) = \sum_{i,j=1}^{n} a_{ij} x_i x_j + \sum_{i=1}^{n} b_i x_i + c = \mathbf{x}^T A_i \mathbf{x} + B\mathbf{x} + c,$$

The MinRank attack

We rewrite each generator polynomial f_i in the ideal $I = \langle f_1, \ldots, f_m \rangle$ induced by the Legendre PRF, as folllows:

$$f_i(x_1, \ldots, x_n) = \sum_{i,j=1}^{n} a_{ij} x_i x_j + \sum_{i=1}^{n} b_i x_i + c = \mathbf{x}^T A_i \mathbf{x} + B\mathbf{x} + c,$$

Each polynomial f_i can be represented in the extension field, in the following form:

$$\mathcal{F}_i(X) = \sum_{i,j=1}^{n} \alpha_{ij} X^{q^{i-1}+q^{j-1}} + \sum_{i=1}^{n} \beta_i X^{q^{i-1}} + \gamma = \mathbf{X}^T M_i \mathbf{X} + N_i \mathbf{X} + \gamma, \qquad (2)$$

Group structure of the solutions of a Legendre key-recovery attack

- We saw that every K such that the symbol of K and $K+1$ is a certain Legendre symbol sequence (e.g., $(1,1)$) lies on a Pell-conic.

Group structure of the solutions of a Legendre key-recovery attack

- We saw that every K such that the symbol of K and $K+1$ is a certain Legendre symbol sequence (e.g., $(1,1)$) lies on a Pell-conic.
- A single Pell conic has genus 0.

Group structure of the solutions of a Legendre key-recovery attack

- We saw that every K such that the symbol of K and $K+1$ is a certain Legendre symbol sequence (e.g., $(1,1)$) lies on a Pell-conic.
- A single Pell conic has genus 0.
- For a triplet of Legendre-symbol sequences, the solutions lie on a non-singular elliptic curve that has genus 1. Do you see a pattern?

Group structure of the solutions of a Legendre key-recovery attack

- We saw that every K such that the symbol of K and $K+1$ is a certain Legendre symbol sequence (e.g., $(1,1)$) lies on a Pell-conic.
- A single Pell conic has genus 0.
- For a triplet of Legendre-symbol sequences, the solutions lie on a non-singular elliptic curve that has genus 1. Do you see a pattern?
- For a quintuple of Legendre symbol sequence (e.g., $(1,1,-1,-1,1)$) the solutions lie on a curve with genus 5.

Group structure of the solutions of a Legendre key-recovery attack

- We saw that every K such that the symbol of K and $K+1$ is a certain Legendre symbol sequence (e.g., $(1,1)$) lies on a Pell-conic.
- A single Pell conic has genus 0.
- For a triplet of Legendre-symbol sequences, the solutions lie on a non-singular elliptic curve that has genus 1. Do you see a pattern?
- For a quintuple of Legendre symbol sequence (e.g., $(1,1,-1,-1,1)$) the solutions lie on a curve with genus 5.
- Generally speaking, the solutions of a Legendre PRF key-recovery attack lie on a high-degree algebraic curve with high genus.

Group structure of the solutions of a Legendre key-recovery attack

- We saw that every K such that the symbol of K and $K+1$ is a certain Legendre symbol sequence (e.g., $(1,1)$) lies on a Pell-conic.
- A single Pell conic has genus 0.
- For a triplet of Legendre-symbol sequences, the solutions lie on a non-singular elliptic curve that has genus 1. Do you see a pattern?
- For a quintuple of Legendre symbol sequence (e.g., $(1,1,-1,-1,1)$) the solutions lie on a curve with genus 5.
- Generally speaking, the solutions of a Legendre PRF key-recovery attack lie on a high-degree algebraic curve with high genus.
- The solutions of a Legendre key-recovery attack ***lack a group structure***!!!

Table of Contents

1. Preliminaries
   - The dawn of a new era of symmetric key primitives
   - Mathematical and cryptographic background
2. The Legendre PRF as MQ
   - The underdetermined case
   - The overdetermined case
3. Security of the Legendre PRF
   - Direct algebraic attacks
   - Interpolation attacks
   - Rank attacks
   - A security argument from algebraic geometry
4. Applications
   - The Legendre Verifiable random function
   - An OPPRF from the Legendre PRF
5. Future Directions

Efficient VRF from the Legendre PRF

The Legendre PRF evaluator wants to prove that the following binary relation $\mathcal{R} : \{0,1\}^* \times \{0,1\}^*$ holds:

$$\mathcal{R}_{PRF} = \left\{ \left(\{n\}_K, K \right) : \{n\}_K = \left(\left(\frac{K}{p} \right), \left(\frac{K+1}{p} \right), \ldots, \left(\frac{K+n-1}{p} \right) \right) \right\},$$

Efficient VRF from the Legendre PRF

The Legendre PRF evaluator wants to prove that the following binary relation $\mathcal{R} : \{0,1\}^* \times \{0,1\}^*$ holds:

$$\mathcal{R}_{PRF} = \left\{ \left(\{n\}_K, K \right) : \{n\}_K = \left(\left(\frac{K}{p}\right), \left(\frac{K+1}{p}\right), \ldots, \left(\frac{K+n-1}{p}\right) \right) \right\},$$

which is equivalent to the relation:

$$\mathcal{R}^*_{PRF} = \left\{ \left(\{n\}_K, \mathbf{x} \right) : (f_1(\mathbf{x}) = 0, f_2(\mathbf{x}) = 0, \ldots, f_m(\mathbf{x}) = 0) \right\},$$

where the multivariate quadratic polynomials $(f_i)_{i=1}^m$ are defined on the previous slides.

The arithmetic circuit representation of the Legendre VRF statement

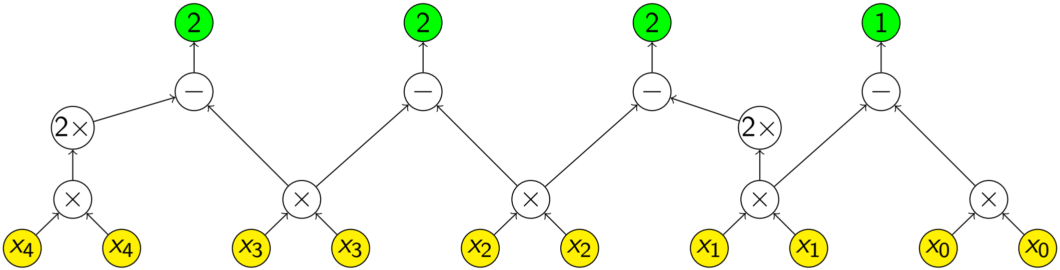


Figure: Arithmetic circuit representation of the ZKP statement that proves the relation $\mathcal{R}_{PRF} = \{\{5\}_K = (1, 1, -1, -1, 1), K\}$, where 2 is the least quadratic non-residue. Applying our arithmetization the PRF evaluator proves that it knows the zeros of the following polynomials $(2x_4^2 - x_3^2 = 2, x_3^2 - x_2^2 = 2, x_2^2 - x_1^2 = 2, x_1^2 - x_0^2 = 1)$. Nodes with $2x$ denote a multiplication gate, where one of the inputs is the constant quadratic non-residue 2. Note, that the arithmetic circuit has a constant multiplicative depth of two.

Overview of the most important VRF constructions

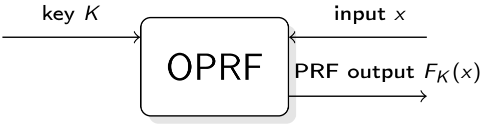
		Time complexity		
	$\lvert\pi\rvert$	Prove	Verify	Assumption
[GNP+15]	$1\mathbb{G}$	$1\mathrm{H} + 1\mathbb{G}$	$1\mathrm{H} + 1\mathbb{G}$	Factoring
[PWH+17]	$1\mathbb{G} + 2\mathbb{F}_p$	$3\mathrm{H} + 2\mathbb{G}$	$3\mathrm{H} + 4\mathbb{G}$	EC-DDH
[BGLS03]	$1\mathbb{G}$	$2\mathrm{H} + 1\mathbb{G}$	$1P$	co-DH
[DY05]	$1\mathbb{G}$	$1\mathbb{G} + 1\mathbb{F}_p$	$2\mathbb{G} + 2P$	q-DBDHI
[LBM20]	$1\mathbb{G}$	$1\mathbb{G}$	$1P$	q-DDHE
[EKS+20]	$\mathcal{O}(k + l)$	$\mathcal{O}(kl)$	$\mathcal{O}(kl)$	Module-SIS
Legendre†	$3\mathbb{G}$	$9n\mathbb{G}$	$n\mathbb{G} + 3P$	SLS, KEA
Legendre*	$\mathcal{O}(\log(n))\mathbb{G}$	$\mathcal{O}(n\log(n))\mathbb{G}$	$\mathcal{O}(\log(n))\mathbb{G}$	SLS

Table: Overview of various VRF constructions. Hashing, group operations, exponentiation and pairings are denoted as $\mathrm{H}, \mathbb{G}, \mathbb{F}_p, P$ respectively. n is the length of the Legendre sequence.

Oblivious Pseudorandom Functions (OPRFs)

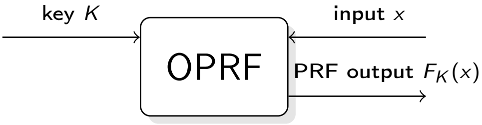
- An Oblivious PRF (OPRF) allows a sender and a receiver to evaluate a keyed PRF $F_K(x)$, such that $F_K(\cdot)$ is hold by the sender, while receiver holds x.

Oblivious Pseudorandom Functions (OPRFs)



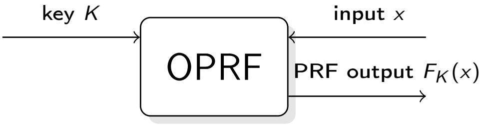
- An Oblivious PRF (OPRF) allows a sender and a receiver to evaluate a keyed PRF $F_K(x)$, such that $F_K(\cdot)$ is hold by the sender, while receiver holds x.
- At the end of the protocol sender learns nothing, while receiver outputs $F_K(x)$.

Oblivious Pseudorandom Functions (OPRFs)



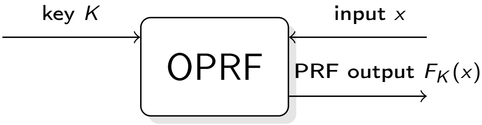
- An Oblivious PRF (OPRF) allows a sender and a receiver to evaluate a keyed PRF $F_K(x)$, such that $F_K(\cdot)$ is hold by the sender, while receiver holds x.
- At the end of the protocol sender learns nothing, while receiver outputs $F_K(x)$.
- OPRFs have many applications in cryptography, e.g. private set intersection, keyword search,. . .

Oblivious Pseudorandom Functions (OPRFs)



- An Oblivious PRF (OPRF) allows a sender and a receiver to evaluate a keyed PRF $F_K(x)$, such that $F_K(\cdot)$ is hold by the sender, while receiver holds x.
- At the end of the protocol sender learns nothing, while receiver outputs $F_K(x)$.
- OPRFs have many applications in cryptography, e.g. private set intersection, keyword search,...
- Immediate (inefficient) idea: take `_insert your favourite PRF here_` and evalute it generically in a two-party setting. This works for any PRF, like AES, SHA-3 etc.

Oblivious Pseudorandom Functions (OPRFs)



- An Oblivious PRF (OPRF) allows a sender and a receiver to evaluate a keyed PRF $F_K(x)$, such that $F_K(\cdot)$ is hold by the sender, while receiver holds x.
- At the end of the protocol sender learns nothing, while receiver outputs $F_K(x)$.
- OPRFs have many applications in cryptography, e.g. private set intersection, keyword search,. . .
- Immediate (inefficient) idea: take `_insert your favourite PRF here_` and evalute it generically in a two-party setting. This works for any PRF, like AES, SHA-3 etc.
- The MPC realisation of the Legendre PFF by [GRR+16], implies an OPRF protocol.

The Legendre OPRF

Participants: sender $\mathcal{S}$, receiver $\mathcal{R}$.

Preprocessing:

- Random square share $[s^2]$ generation,
- Beaver triple generation for the $\boxdot$ operation.

The Legendre OPRF

Participants: sender $\mathcal{S}$, receiver $\mathcal{R}$.

Preprocessing:

- Random square share $[s^2]$ generation,
- Beaver triple generation for the $\boxdot$ operation.

Input:

- $\mathcal{S}$: $K \in \mathbb{F}_p$,
- $\mathcal{R}$: $x \in \mathbb{F}_p$.

The Legendre OPRF

Participants: sender $\mathcal{S}$, receiver $\mathcal{R}$.

Preprocessing:

- Random square share $[s^2]$ generation,
- Beaver triple generation for the $\boxdot$ operation.

Input:

- $\mathcal{S}$: $K \in \mathbb{F}_p$,
- $\mathcal{R}$: $x \in \mathbb{F}_p$.

Evaluation:

1. $\mathcal{S}$, $\mathcal{R}$ share $[K], [x]$ with each other,

The Legendre OPRF

Participants: sender $\mathcal{S}$, receiver $\mathcal{R}$.

Preprocessing:

- Random square share $[s^2]$ generation,
- Beaver triple generation for the $\boxdot$ operation.

Input:

- $\mathcal{S}$: $K \in \mathbb{F}_p$,
- $\mathcal{R}$: $x \in \mathbb{F}_p$.

Evaluation:

1. $\mathcal{S}$, $\mathcal{R}$ share $[K], [x]$ with each other,
2. both compute $[c] = [s^2] \boxdot ([K] + [x])$,

The Legendre OPRF

Participants: sender $\mathcal{S}$, receiver $\mathcal{R}$.

Preprocessing:

- Random square share $[s^2]$ generation,
- Beaver triple generation for the $\boxdot$ operation.

Input:

- $\mathcal{S}$: $K \in \mathbb{F}_p$,
- $\mathcal{R}$: $x \in \mathbb{F}_p$.

Evaluation:

1. $\mathcal{S}, \mathcal{R}$ share $[K], [x]$ with each other,
2. both compute $[c] = [s^2] \boxdot ([K] + [x])$,
3. $\mathcal{S}$ sends $[c]$ to $\mathcal{R}$,

The Legendre OPRF

Participants: sender $\mathcal{S}$, receiver $\mathcal{R}$.

Preprocessing:

- Random square share $[s^2]$ generation,
- Beaver triple generation for the $\boxdot$ operation.

Input:

- $\mathcal{S}$: $K \in \mathbb{F}_p$,
- $\mathcal{R}$: $x \in \mathbb{F}_p$.

Evaluation:

1. $\mathcal{S}, \mathcal{R}$ share $[K], [x]$ with each other,
2. both compute $[c] = [s^2] \boxdot ([K] + [x])$,
3. $\mathcal{S}$ sends $[c]$ to $\mathcal{R}$,
4. $\mathcal{R}$ outputs $L_p(c) = L_p(K + x)$.

Oblivious **Programmable** PRFs (O**P**PRFs)

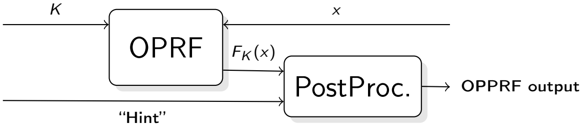
- In addition to being an OPRF, sender can also program the output of the OPRF at certain points. Specifically, sender can choose $F_K(\cdot)$ such that it holds for prescribed x_i and y_i, that $y_i = F_K(x_i)$, for some $i \in [0, n]$, where n is the number of programmed points.

Oblivious **Programmable** PRFs (O**P**PRFs)

- In addition to being an OPRF, sender can also program the output of the OPRF at certain points. Specifically, sender can choose $F_K(\cdot)$ such that it holds for prescribed x_i and y_i, that $y_i = F_K(x_i)$, for some $i \in [0, n]$, where n is the number of programmed points.
- Cornerstone of state-of-the-art Private Set Intersection protocols [KMP+17].

Oblivious **Programmable** PRFs (O**P**PRFs)

- In addition to being an OPRF, sender can also program the output of the OPRF at certain points. Specifically, sender can choose $F_K(\cdot)$ such that it holds for prescribed x_i and y_i, that $y_i = F_K(x_i)$, for some $i \in [0, n]$, where n is the number of programmed points.
- Cornerstone of state-of-the-art Private Set Intersection protocols [KMP+17].
- Kolesnikov et al [KMP+17] introduces three *generic* transformations to transform any OPRF generically to being an O**P**PRF.



The Legendre O**P**PRF

- Our "programming design" is specific for the Legendre PRF.

The Legendre O**P**PRF

- Our "programming design" is specific for the Legendre PRF.
- Just brute-force search a random prime, that satisfies the programming constraints! Caveat: exponential programming time!

The Legendre O**P**PRF

- Our "programming design" is specific for the Legendre PRF.
- Just brute-force search a random prime, that satisfies the programming constraints! Caveat: exponential programming time!
- Use quadratic reciprocity and the Chinese-Remainder Theorem. Quasi-linear programming time! Caveat: linear modulus size!

The Legendre O**P**PRF

- Our "programming design" is specific for the Legendre PRF.
- Just brute-force search a random prime, that satisfies the programming constraints! Caveat: exponential programming time!
- Use quadratic reciprocity and the Chinese-Remainder Theorem. Quasi-linear programming time! Caveat: linear modulus size!
- Luckily, in a PSI application, only a handful programmed points are needed...

Programming the Legendre PRF

- The programming constraints can be expressed as follows: find a p prime, s.t. it holds for all $i \in [0, n)$: $y_i = \left(\frac{x_i}{p}\right) = \left(\frac{p}{x_i}\right)(-1)^{\frac{(p-1)(x_i-1)}{4}}$.

Programming the Legendre PRF

- The programming constraints can be expressed as follows: find a p prime, s.t. it holds for all $i \in [0, n)$: $y_i = \left(\frac{x_i}{p}\right) = \left(\frac{p}{x_i}\right)(-1)^{\frac{(p-1)(x_i-1)}{4}}$.
- Without loss of generality, search p in the form $p \equiv 1 \mod 4$.

Programming the Legendre PRF

- The programming constraints can be expressed as follows: find a p prime, s.t. it holds for all $i \in [0, n)$: $y_i = \left(\frac{x_i}{p}\right) = \left(\frac{p}{x_i}\right)(-1)^{\frac{(p-1)(x_i-1)}{4}}$.
- Without loss of generality, search p in the form $p \equiv 1 \mod 4$.
- Now, compute $y_i(-1)^{\frac{(p-1)(x_i-1)}{4}} = \left(\frac{p}{x_i}\right)$

Programming the Legendre PRF

- The programming constraints can be expressed as follows: find a p prime, s.t. it holds for all $i \in [0, n)$: $y_i = \left(\frac{x_i}{p}\right) = \left(\frac{p}{x_i}\right)(-1)^{\frac{(p-1)(x_i-1)}{4}}$.
- Without loss of generality, search p in the form $p \equiv 1 \mod 4$.
- Now, compute $y_i(-1)^{\frac{(p-1)(x_i-1)}{4}} = \left(\frac{p}{x_i}\right)$
- Identify congruence classes m_i in $\mathbb{Z}_{x_i}$, s.t. $\left(\frac{m_i}{x_i}\right) = y_i(-1)^{\frac{(p-1)(x_i-1)}{4}}$.

Programming the Legendre PRF

- The programming constraints can be expressed as follows: find a p prime, s.t. it holds for all $i \in [0, n)$: $y_i = \left(\frac{x_i}{p}\right) = \left(\frac{p}{x_i}\right)(-1)^{\frac{(p-1)(x_i-1)}{4}}$.
- Without loss of generality, search p in the form $p \equiv 1 \mod 4$.
- Now, compute $y_i(-1)^{\frac{(p-1)(x_i-1)}{4}} = \left(\frac{p}{x_i}\right)$
- Identify congruence classes m_i in $\mathbb{Z}_{x_i}$, s.t. $\left(\frac{m_i}{x_i}\right) = y_i(-1)^{\frac{(p-1)(x_i-1)}{4}}$.
- For each i let M_i let the set of these congruence classes be $M_i = \left\{m \middle| m \in \mathbb{Z}_{x_i} \wedge b_i(-1)^{\frac{(p-1)(x_i-1)}{4}} = \left(\frac{m}{x_i}\right)\right\}$. If $m \in M_i$, then p can be sought as $p \equiv m \mod x_i$.

Programming the Legendre PRF

- The programming constraints can be expressed as follows: find a p prime, s.t. it holds for all $i \in [0, n)$: $y_i = \left(\frac{x_i}{p}\right) = \left(\frac{p}{x_i}\right)(-1)^{\frac{(p-1)(x_i-1)}{4}}$.
- Without loss of generality, search p in the form $p \equiv 1 \mod 4$.
- Now, compute $y_i(-1)^{\frac{(p-1)(x_i-1)}{4}} = \left(\frac{p}{x_i}\right)$
- Identify congruence classes m_i in $\mathbb{Z}_{x_i}$, s.t. $\left(\frac{m_i}{x_i}\right) = y_i(-1)^{\frac{(p-1)(x_i-1)}{4}}$.
- For each i let M_i let the set of these congruence classes be $M_i = \left\{m | m \in \mathbb{Z}_{x_i} \wedge b_i(-1)^{\frac{(p-1)(x_i-1)}{4}} = \left(\frac{m}{x_i}\right)\right\}$. If $m \in M_i$, then p can be sought as $p \equiv m \mod x_i$.
- Note, p is a solution of a simultaneous congruence system: $p \equiv m_i \mod x_i$, for all $i \in [0, n)$, where $m_i \in M_i$. Solve this by the Chinese-Remainder Theorem.

Overview of the state-of-the-art O**P**PRFs

OPPRF	Programming complexity	Hint size	Online communication complexity	Constraint on no. of programmed points	No. of evaluations
Lagrange	$O(n^2)$	$O(n)$	$(n+kn)\ \mathbb{G}$	space-efficiency	any
Garbled BF.	$O(n\lambda_{\mathrm{BF}})$	$n\lambda_{\mathrm{BF}}$	$(60n+kn)\ \mathbb{G}$	space-efficiency	any
Table-based	$O(n)$	$O(n)$	$(n+kn)\ \mathbb{G}$	space-efficiency	1
Legendre CRT	$O(n \log n)$	1	$\mathcal{O}(n)\ \mathbb{G}$	depends on λ	any
Legendre bruteforce	$O(2^n)$	1	$1\ \mathbb{G}$	time-efficiency	any

Table: Comparison of our results with the generic O**P**PRF constructions of [KMP+17] (relying on the OPRF of [KKRT16]). The number of programmed input positions is denoted as n, λ_{BF} is the soundness parameter of the Bloom filter, while k denotes the number of base-OTs, typically $k \approx 4\lambda$.

Table of Contents

1 Preliminaries
- The dawn of a new era of symmetric key primitives
- Mathematical and cryptographic background

2 The Legendre PRF as MQ
- The underdetermined case
- The overdetermined case

3 Security of the Legendre PRF
- Direct algebraic attacks
- Interpolation attacks
- Rank attacks
- A security argument from algebraic geometry

4 Applications
- The Legendre Verifiable random function
- An OPPRF from the Legendre PRF

5 Future Directions

Where do we go from here?

- Concrete and/or asymptotic bounds for the degree of regularity of the ideal(s)?

Where do we go from here?

- Concrete and/or asymptotic bounds for the degree of regularity of the ideal(s)?
- Novel, more efficient Legendre PRF key-recovery attacks? It would be great for for further refining the provided bit-security of the PRF.

Where do we go from here?

- Concrete and/or asymptotic bounds for the degree of regularity of the ideal(s)?
- Novel, more efficient Legendre PRF key-recovery attacks? It would be great for for further refining the provided bit-security of the PRF.
- More cryptographic assumptions?

Where do we go from here?

- Concrete and/or asymptotic bounds for the degree of regularity of the ideal(s)?
- Novel, more efficient Legendre PRF key-recovery attacks? It would be great for for further refining the provided bit-security of the PRF.
- More cryptographic assumptions?
- Real-world deployments of the Legendre PRF? Let us know!

Acknowledgements and Q&A

We thank **Gergő Zábrádi** for insightful discussions.

Thanks for the attention!
Questions?

References I

- Martin R Albrecht, Carlos Cid, Lorenzo Grassi, Dmitry Khovratovich, Reinhard Lüftenegger, Christian Rechberger, and Markus Schofnegger, *Algebraic cryptanalysis of stark-friendly designs: application to marvellous and mimc*, International Conference on the Theory and Application of Cryptology and Information Security, Springer, 2019, pp. 371–397.
- Tomer Ashur and Siemen Dhooghe, *Marvellous: a stark-friendly family of cryptographic primitives.*, IACR Cryptol. ePrint Arch. **2018** (2018), 1098.
- Martin Albrecht, Lorenzo Grassi, Christian Rechberger, Arnab Roy, and Tyge Tiessen, *Mimc: Efficient encryption and cryptographic hashing with minimal multiplicative complexity*, International Conference on the Theory and Application of Cryptology and Information Security, Springer, 2016, pp. 191–219.

References II

Ward Beullens, Tim Beyne, Aleksei Udovenko, and Giuseppe Vitto, *Cryptanalysis of the legendre prf and generalizations*, IACR Transactions on Symmetric Cryptology (2020), 313–330.

Dan Boneh, Craig Gentry, Ben Lynn, and Hovav Shacham, *Aggregate and verifiably encrypted signatures from bilinear maps*, International Conference on the Theory and Applications of Cryptographic Techniques, Springer, 2003, pp. 416–432.

Ivan Bjerre Damgård, *On the randomness of legendre and jacobi sequences*, Conference on the Theory and Application of Cryptography, Springer, 1988, pp. 163–172.

References III

- Cunsheng Ding, T Hesseseth, and Weijuan Shan, *On the linear complexity of legendre sequences*, IEEE Transactions on Information Theory **44** (1998), no. 3, 1276–1278.
- Yevgeniy Dodis and Aleksandr Yampolskiy, *A verifiable random function with short proofs and keys*, International Workshop on Public Key Cryptography, Springer, 2005, pp. 416–431.
- Muhammed F Esgin, Veronika Kuchta, Amin Sakzad, Ron Steinfeld, Zhenfei Zhang, Shifeng Sun, and Shumo Chu, *Practical post-quantum few-time verifiable random function with applications to algorand*, IACR Cryptol. ePrint Arch **2020** (2020), 1222.

References IV

- Lorenzo Grassi, Dmitry Khovratovich, Christian Rechberger, Arnab Roy, and Markus Schofnegger, *Poseidon: A new hash function for zero-knowledge proof systems*, Proceedings of the 30th USENIX Security Symposium, USENIX Association, 2020.
- Katalin Gyarmati, Christian Mauduit, and András Sárközy, *The cross-correlation measure for families of binary sequences.*, 2014.
- Sharon Goldberg, Moni Naor, Dimitrios Papadopoulos, Leonid Reyzin, Sachin Vasant, and Asaf Ziv, *Nsec5: Provably preventing dnssec zone enumeration.*, NDSS, 2015.

References V

- Lorenzo Grassi, Christian Rechberger, Dragos Rotaru, Peter Scholl, and Nigel P Smart, *Mpc-friendly symmetric key primitives*, Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, ACM, 2016, pp. 430–443.
- Dmitry Khovratovich, *Key recovery attacks on the legendre prfs within the birthday bound*, Cryptology ePrint Archive, Report 2019/862, 2019, https://eprint.iacr.org/2019/862.
- Novak Kaluderovic, Thorsten Kleinjung, and Dusan Kostic, *Improved key recovery on the legendre prf.*, IACR Cryptol. ePrint Arch. **2020** (2020), 98.

References VI

- Vladimir Kolesnikov, Ranjit Kumaresan, Mike Rosulek, and Ni Trieu, *Efficient batched oblivious PRF with applications to private set intersection*, CCS, ACM, 2016, pp. 818–829.
- Vladimir Kolesnikov, Naor Matania, Benny Pinkas, Mike Rosulek, and Ni Trieu, *Practical multi-party private set intersection from symmetric-key techniques*, CCS, ACM, 2017, pp. 1257–1272.
- Bei Liang, Gustavo Banegas, and Aikaterini Mitrokotsa, *Statically aggregate verifiable random functions and application to e-lottery*, Cryptography **4** (2020), no. 4, 37.

References VII

- Chaoyun Li and Bart Preneel, *Improved interpolation attacks on cryptographic primitives of low algebraic degree*, International Conference on Selected Areas in Cryptography, Springer, 2019, pp. 171–193.
- Christian Mauduit and András Sárközy, *On finite pseudorandom binary sequences i: Measure of pseudorandomness, the legendre symbol*, Acta Arithmetica **82** (1997), no. 4, 365–377.
- Rene Peralta, *On the distribution of quadratic residues and nonresidues modulo a prime number*, Mathematics of Computation **58** (1992), no. 197, 433–440.
- Dimitrios Papadopoulos, Duane Wessels, Shumon Huque, Moni Naor, Jan Včelák, Leonid Reyzin, and Sharon Goldberg, *Making nsec5 practical for dnssec*, Cryptology ePrintArchive, Report 2017/099 (2017).