Cryptanalysis of the generalised Legendre pseudorandom function

Novak Kaluderovic, Thorsten Kleinjung, Dusan Kostic

July 3, 2020

EPFL

Background

Legendre PRF

Legendre PRF

Damgård, 1988 [Dam90]: The Legendre PRF

Legendre PRF

Damgård, 1988 [Dam90]: The Legendre PRF

$$\mathcal{O}_k(x) = \left(\frac{x+k}{p}\right), \quad k \in \mathbb{F}_p$$

Legendre PRF

Damgård, 1988 [Dam90]: The Legendre PRF

$$\mathcal{O}_k(x) = \left(\frac{x+k}{p}\right), \quad k \in \mathbb{F}_p$$

Russell, Shparlinski, 2004 [RS04]: The Generalised Legendre PRF

Damgård, 1988 [Dam90]: The Legendre PRF

$$\mathcal{O}_k(x) = \left(\frac{x+k}{p}\right), \quad k \in \mathbb{F}_p$$

Russell, Shparlinski, 2004 [RS04]: The Generalised Legendre PRF

$$\mathcal{O}_f(x) = \left(\frac{f(x)}{p}\right), \quad f \in \mathbb{F}_p[x]_r$$

Use-cases

Use-cases

Orders of magnitude slower than cryptographic PRFs.

Use-cases

Orders of magnitude slower than cryptographic PRFs.

Grassi et al., 2016 [GRR+16]: Suitable for multiparty computation.

Use-cases

Orders of magnitude slower than cryptographic PRFs.

Grassi et al., 2016 [GRR+16]: Suitable for multiparty computation.

Ethereum, 2019 [Fei19]: Plans to incorporate it in the blockchain.

Use-cases

Orders of magnitude slower than cryptographic PRFs.

Grassi et al., 2016 [GRR+16]: Suitable for multiparty computation.

Ethereum, 2019 [Fei19]: Plans to incorporate it in the blockchain.

Ethereum, 2019 [Fei19]: Online challenges to break the function.

Results

Problem
Given access to $\mathcal{O}_f$, find f.

Results

Problem
Given access to $\mathcal{O}_f$, find f.

Solution
Table-based collision search.

Results

Problem
Given access to $\mathcal{O}_f$, find f.

Solution
Table-based collision search.

General case
Table: $O(p^3)$, Search: $O(p^{r-3})$

Results

Problem
Given access to $\mathcal{O}_f$, find f.

Solution
Table-based collision search.

General case
Table: $O(p^3)$, Search: $O(p^{r-3})$

Limited query case
Table: $O(M^2/\log p)$, Search: $O(p^r \log p/M^2)$

Limited query case

Limited query case

Khovratovich [Kho19]: Table size:$O(1) \sim O(\frac{M}{\log p})$.

Limited query case

Khovratovich [Kho19]: Table size:$O(1) \sim O(\frac{M}{\log p})$.

Beullens et al. [BBUV19]: Table size $O(\frac{M^2}{\log^2 p})$.

Limited query case

Khovratovich [Kho19]: Table size:$O(1) \sim O(\frac{M}{\log p})$.

Beullens et al. [BBUV19]: Table size $O(\frac{M^2}{\log^2 p})$.

Us: Table size $O(\frac{M^2}{\log p})$.

General case

Khovratovich [Kho19]: Linear yield $\sim p$.

General case

Khovratovich [Kho19]: Linear yield $\sim p$.

Beullens et al. [BBUV19]: Quadratic yield $\sim p^2$.

General case

Khovratovich [Kho19]: Linear yield $\sim p$.

Beullens et al. [BBUV19]: Quadratic yield $\sim p^2$.

Us: Cubic yield $\sim p^3$.

Legendre sequence
Let $a \in \mathbb{F}_p$ and $L \in \mathbb{N}$,

$$\{a\}_L := \left(\frac{a}{p}\right), \left(\frac{a+1}{p}\right), \left(\frac{a+2}{p}\right), \dots, \left(\frac{a+L-1}{p}\right).$$

Legendre sequence
Let $a \in \mathbb{F}_p$ and $L \in \mathbb{N}$,

$$\{a\}_L := \left(\frac{a}{p}\right), \left(\frac{a+1}{p}\right), \left(\frac{a+2}{p}\right), \dots, \left(\frac{a+L-1}{p}\right).$$

Assumption
For $L = 2\lfloor \log p \rfloor$ we have

$$\{a\}_L = \{b\}_L \text{ if and only if } a = b.$$

Legendre Sequences

Legendre sequence
Let $a \in \mathbb{F}_p$ and $L \in \mathbb{N}$,

$$\{a\}_L := \left(\frac{a}{p}\right), \left(\frac{a+1}{p}\right), \left(\frac{a+2}{p}\right), \ldots, \left(\frac{a+L-1}{p}\right).$$

Assumption
For $L = \lfloor \log p \log \log p \rfloor$ we have

$$\{a\}_L = \{b\}_L \text{ if and only if } a = b.$$

Legendre Sequences

Generalised Legendre sequence
Let $f \in \mathbb{F}_p[x]_r$ and $L \in \mathbb{N}$,

$$\{f\}_L := \left(\frac{f(0)}{p}\right), \left(\frac{f(1)}{p}\right), \left(\frac{f(2)}{p}\right), \ldots, \left(\frac{f(L-1)}{p}\right).$$

Generalised assumption:
For $L = r\lfloor \log p \log\log p \rfloor$ we have

$$\{f\}_L = \{g\}_L \text{ if and only if } f = g.$$

Algorithm

Table-based collision search

Table:
Make a table with many Legendre sequences $\{f_m\}_L$ such that

Table-based collision search

Table:
Make a table with many Legendre sequences $\{f_m\}_L$ such that

- The sequence $\{f_m\}_L$ can be computed from $\mathcal{O}_f$.

Table-based collision search

Table:
Make a table with many Legendre sequences $\{f_m\}_L$ such that

- The sequence $\{f_m\}_L$ can be computed from $\mathcal{O}_f$.
- From f_m we can obtain f.

Table-based collision search

Table:
Make a table with many Legendre sequences $\{f_m\}_L$ such that

- The sequence $\{f_m\}_L$ can be computed from $\mathcal{O}_f$.
- From f_m we can obtain f.

Search:
Generate random $g(x)$ and look for $\{g\}_L$ in the table.

Table-based collision search

Table:
Make a table with many Legendre sequences $\{f_m\}_L$ such that

- The sequence $\{f_m\}_L$ can be computed from $\mathcal{O}_f$.
- From f_m we can obtain f.

Search:
Generate random $g(x)$ and look for $\{g\}_L$ in the table.

If $\{g\}_L = \{f_m\}_L$ then $g = f_m$, and we can obtain f.

Möbius transformations

Rational transformations of $\mathbb{P}^1$:

$$\begin{aligned} \varphi_m : \ \mathbb{P}^1 &\longrightarrow \mathbb{P}^1 \\ [x : y] &\longmapsto [ax + by : cx + dy], \end{aligned}$$

Möbius transformations

Rational transformations of $\mathbb{P}^1$:

$$\begin{aligned}\varphi_m :\ \mathbb{P}^1 &\longrightarrow \mathbb{P}^1 \\ [x:y] &\longmapsto [ax+by : cx+dy],\end{aligned}$$

Isomorphic to $PGL_2(\mathbb{F}_p)$ given by $\varphi_m \leftrightarrow m = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$.

Möbius transformations

Rational transformations of $\mathbb{P}^1$:

$$\begin{aligned} \varphi_m :\ \mathbb{P}^1 &\longrightarrow \mathbb{P}^1 \\ [x : y] &\longmapsto [ax + by : cx + dy], \end{aligned}$$

Isomorphic to $PGL_2(\mathbb{F}_p)$ given by $\varphi_m \leftrightarrow m = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$.

Action on monic polynomials:

Möbius transformations

Rational transformations of $\mathbb{P}^1$:

$$\begin{aligned} \varphi_m :\ \mathbb{P}^1 &\longrightarrow \mathbb{P}^1 \\ [x : y] &\longmapsto [ax + by : cx + dy], \end{aligned}$$

Isomorphic to $PGL_2(\mathbb{F}_p)$ given by $\varphi_m \leftrightarrow m = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$.

Action on monic polynomials:

$$m \cdot f(x) = f_m(x)$$

Möbius transformations

Rational transformations of $\mathbb{P}^1$:

$$\begin{aligned} \varphi_m :\ \mathbb{P}^1 &\longrightarrow \mathbb{P}^1 \\ [x : y] &\longmapsto [ax + by : cx + dy], \end{aligned}$$

Isomorphic to $PGL_2(\mathbb{F}_p)$ given by $\varphi_m \leftrightarrow m = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$.

Action on monic polynomials:

$$m \cdot f(x) = f_m(x) := f\left(\tfrac{ax+b}{cx+d}\right)$$

Möbius transformations

Rational transformations of $\mathbb{P}^1$:

$$\begin{aligned}\varphi_m : \mathbb{P}^1 &\longrightarrow \mathbb{P}^1\\ [x:y] &\longmapsto [ax+by : cx+dy],\end{aligned}$$

Isomorphic to $PGL_2(\mathbb{F}_p)$ given by $\varphi_m \leftrightarrow m = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$.

Action on monic polynomials:

$$m \cdot f(x) = f_m(x) := f\left(\tfrac{ax+b}{cx+d}\right)$$

Möbius transformations

Rational transformations of $\mathbb{P}^1$:

$$\begin{aligned}\varphi_m :\ \mathbb{P}^1 &\longrightarrow \mathbb{P}^1\\ [x:y] &\longmapsto [ax+by : cx+dy],\end{aligned}$$

Isomorphic to $PGL_2(\mathbb{F}_p)$ given by $\varphi_m \leftrightarrow m = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$.

Action on monic polynomials:

$$m \cdot f(x) = f_m(x) := f\left(\tfrac{ax+b}{cx+d}\right)(cx+d)^r$$

Möbius transformations

Rational transformations of $\mathbb{P}^1$:

$$\begin{aligned}\varphi_m :\ \mathbb{P}^1 &\longrightarrow \mathbb{P}^1\\ [x:y] &\longmapsto [ax+by : cx+dy],\end{aligned}$$

Isomorphic to $PGL_2(\mathbb{F}_p)$ given by $\varphi_m \leftrightarrow m = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$.

Action on monic polynomials:

$$m \cdot f(x) = f_m(x) := f(\tfrac{ax+b}{cx+d})(cx+d)^r$$

Möbius transformations

Rational transformations of $\mathbb{P}^1$:

$$\begin{aligned}\varphi_m : \mathbb{P}^1 &\longrightarrow \mathbb{P}^1 \\ [x : y] &\longmapsto [ax + by : cx + dy],\end{aligned}$$

Isomorphic to $PGL_2(\mathbb{F}_p)$ given by $\varphi_m \leftrightarrow m = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$.

Action on monic polynomials:

$$m \cdot f(x) = f_m(x) := f\left(\tfrac{ax+b}{cx+d}\right)(cx+d)^r / (f(\tfrac{a}{c})c^r)$$

Möbius transformations

Alternative point of view:

Möbius transformations

Alternative point of view: If $f(x) = \prod_{i=1}^{r}(x - \alpha_i)$

Möbius transformations

Alternative point of view: If $f(x) = \prod_{i=1}^{r}(x - \alpha_i)$ then

$$f_m(x) = \prod_{i=1}^{r}(x - m^{-1}\alpha_i) = \prod_{i=1}^{r}(x - \frac{d\alpha_i - b}{-c\alpha_i + a}).$$

Möbius transformations

Alternative point of view: If $f(x) = \prod_{i=1}^{r}(x - \alpha_i)$ then

$$f_m(x) = \prod_{i=1}^{r}(x - m^{-1}\alpha_i) = \prod_{i=1}^{r}(x - \frac{d\alpha_i - b}{-c\alpha_i + a}).$$

$$f_m(x) = f(\tfrac{ax+b}{cx+d})(cx+d)^r/(f(\tfrac{a}{c})c^r).$$

Möbius transformations

Alternative point of view: If $f(x) = \prod_{i=1}^{r}(x - \alpha_i)$ then

$$f_m(x) = \prod_{i=1}^{r}(x - m^{-1}\alpha_i) = \prod_{i=1}^{r}(x - \frac{d\alpha_i - b}{-c\alpha_i + a}).$$

$$f_m(x) = f(\tfrac{ax+b}{cx+d})(cx+d)^r/(f(\tfrac{a}{c})c^r).$$

Computing $\{f_m\}_L$ from $\mathcal{O}_f$:

$$\left(\frac{f_m(x)}{p}\right) = \mathcal{O}_f\left(\frac{ax+b}{cx+d}\right)\left(\frac{cx+d}{p}\right)^r \mathcal{O}_f\left(\frac{a}{c}\right)\left(\frac{c}{p}\right)^r.$$

Möbius transformations

Alternative point of view: If $f(x) = \prod_{i=1}^{r}(x - \alpha_i)$ then

$$f_m(x) = \prod_{i=1}^{r}(x - m^{-1}\alpha_i) = \prod_{i=1}^{r}(x - \frac{d\alpha_i - b}{-c\alpha_i + a}).$$

$$f_m(x) = f(\tfrac{ax+b}{cx+d})(cx+d)^r/(f(\tfrac{a}{c})c^r).$$

Computing $\{f_m\}_L$ from $\mathcal{O}_f$:

$$\left(\frac{f_m(x)}{p}\right) = \mathcal{O}_f\left(\frac{ax+b}{cx+d}\right)\left(\frac{cx+d}{p}\right)^r \mathcal{O}_f\left(\frac{a}{c}\right)\left(\frac{c}{p}\right)^r.$$

Cost per sequence: $L+1$ oracle queries and $L+1$ Legendre symbol computations $\rightarrow$ 1 Legendre sequence.

Möbius transformations

Alternative point of view: If $f(x) = \prod_{i=1}^{r}(x - \alpha_i)$ then

$$f_m(x) = \prod_{i=1}^{r}(x - m^{-1}\alpha_i) = \prod_{i=1}^{r}(x - \frac{d\alpha_i - b}{-c\alpha_i + a}).$$

$$f_m(x) = f(\tfrac{ax+b}{cx+d})(cx+d)^r/(f(\tfrac{a}{c})c^r).$$

Computing $\{f_m\}_L$ from $\mathcal{O}_f$:

$$\left(\frac{f_m(x)}{p}\right) = \mathcal{O}_f\left(\frac{ax+b}{cx+d}\right)\left(\frac{cx+d}{p}\right)^r \mathcal{O}_f\left(\frac{a}{c}\right)\left(\frac{c}{p}\right)^r.$$

Cost per sequence: $L+1$ oracle queries and $L+1$ Legendre symbol computations $\to$ 1 Legendre sequence.

Amortised over all $m \in PGL_2(\mathbb{F}_p)$: p oracle queries and p Legendre symbols $\to$ $(p^3 - p)$ Legendre sequences.

Polynomial types

Lemma
Let $f \in \mathbb{F}_p[x]_r$ be irreducible with $3 \leq r < p$ and consider the action of $PGL_2(\mathbb{F}_p)$ on f. The stabiliser of f is a cyclic group of order $r' \mid \gcd(r, p^2 - 1)$.

Polynomial types

Lemma
Let $f \in \mathbb{F}_p[x]_r$ be irreducible with $3 \leq r < p$ and consider the action of $PGL_2(\mathbb{F}_p)$ on f. The stabiliser of f is a cyclic group of order $r' \mid \gcd(r, p^2 - 1)$.

Three polynomial types

- *Good*: Irreducible and trivial stabiliser
- *Bad*: Irreducible and non-trivial stabiliser
- *Ugly*: Reducible

Good polynomials

Good polynomials

Precomputation
Create a table T containing $\{f_m\}_L$ for all $m \in PGL_2(\mathbb{F}_p)$.

In total $p^3 - p$ sequences.

Good polynomials

Precomputation
Create a table T containing $\{f_m\}_L$ for all $m \in PGL_2(\mathbb{F}_p)$.

In total $p^3 - p$ sequences.

Search
Try random $g(x)$ of degree r and compute $\{g\}_L$ until a hit is found.

Precomputation
Create a table T containing $\{f_m\}_L$ for all $m \in PGL_2(\mathbb{F}_p)$.

In total $p^3 - p$ sequences.

Search
Try random $g(x)$ of degree r and compute $\{g\}_L$ until a hit is found.

Expected run-time: $O(p^{r-3})$ trials.

Bad polynomials

Bad polynomials

Precomputation
Find the stabiliser of f which we know to by cyclic of order $r' \mid r$.

Bad polynomials

Precomputation
Find the stabiliser of f which we know to by cyclic of order $r' \mid r$.

- Trivial: Enumerate $PGL_2(\mathbb{F}_p)$ and isolate matrices that fix f.
  Cost: $O(p^3)$.

Bad polynomials

Precomputation
Find the stabiliser of f which we know to by cyclic of order $r' \mid r$.

- Trivial: Enumerate $PGL_2(\mathbb{F}_p)$ and isolate matrices that fix f. Cost: $O(p^3)$.
- Non-trivial: Enumerate elements of order r' and isolate matrices that fix f. Cost $O(p^2 \log r)$.

Bad polynomials

Precomputation
Find the stabiliser of f which we know to by cyclic of order $r' \mid r$.

- Trivial: Enumerate $PGL_2(\mathbb{F}_p)$ and isolate matrices that fix f. Cost: $O(p^3)$.
- Non-trivial: Enumerate elements of order r' and isolate matrices that fix f. Cost $O(p^2 \log r)$.

Precompute a table with $O(p)$ many sequences $\{f_m\}_L$ such that f_m is fixed by a diagonal matrix.

Bad polynomials

Precomputation
Find the stabiliser of f which we know to by cyclic of order $r' \mid r$.

- Trivial: Enumerate $PGL_2(\mathbb{F}_p)$ and isolate matrices that fix f. Cost: $O(p^3)$.
- Non-trivial: Enumerate elements of order r' and isolate matrices that fix f. Cost $O(p^2 \log r)$.

Precompute a table with $O(p)$ many sequences $\{f_m\}_L$ such that f_m is fixed by a diagonal matrix.

Search
Try random $g(x)$ of degree r that are fixed by a diagonal matrix.

The number of such polynomials is $O(p^{r/r'})$.

Bad polynomials

Precomputation
Find the stabiliser of f which we know to by cyclic of order $r' \mid r$.

- Trivial: Enumerate $PGL_2(\mathbb{F}_p)$ and isolate matrices that fix f. Cost: $O(p^3)$.
- Non-trivial: Enumerate elements of order r' and isolate matrices that fix f. Cost $O(p^2 \log r)$.

Precompute a table with $O(p)$ many sequences $\{f_m\}_L$ such that f_m is fixed by a diagonal matrix.

Search
Try random $g(x)$ of degree r that are fixed by a diagonal matrix.

The number of such polynomials is $O(p^{r/r'})$.

Expected run-time: $O(p^{r/r'-1})$ trials.

Ugly polynomials

Ugly polynomials

Let $f(x) = l(x)h(x)$ with $r_h \geq r/2$ the degree of $h(x)$.

Ugly polynomials

Let $f(x) = l(x)h(x)$ with $r_h \geq r/2$ the degree of $h(x)$.

Precomputation
Table T_1 containing $\{f_m\}_L$ for all $m \in PGL_2(\mathbb{F}_p)$.

Ugly polynomials

Let $f(x) = l(x)h(x)$ with $r_h \geq r/2$ the degree of $h(x)$.

Precomputation
Table T_1 containing $\{f_m\}_L$ for all $m \in PGL_2(\mathbb{F}_p)$.

Table T_2 containing $\{l'\}_L$ for all polynomials l' of degree $r - r_h$.

Ugly polynomials

Let $f(x) = l(x)h(x)$ with $r_h \geq r/2$ the degree of $h(x)$.

Precomputation
Table T_1 containing $\{f_m\}_L$ for all $m \in PGL_2(\mathbb{F}_p)$.

Table T_2 containing $\{l'\}_L$ for all polynomials l' of degree $r - r_h$.

Table T containing $\{f_m\}_L\{l'\}_L$ for all m and l'. Size: $O(p^{3+r-r_h})$.

Ugly polynomials

Let $f(x) = l(x)h(x)$ with $r_h \geq r/2$ the degree of $h(x)$.

Precomputation
Table T_1 containing $\{f_m\}_L$ for all $m \in PGL_2(\mathbb{F}_p)$.

Table T_2 containing $\{l'\}_L$ for all polynomials l' of degree $r - r_h$.

Table T containing $\{f_m\}_L\{l'\}_L$ for all m and l'. Size: $O(p^{3+r-r_h})$.

Search
Try random $h'(x)$ of degree r_h until a hit is found.

Ugly polynomials

Let $f(x) = l(x)h(x)$ with $r_h \geq r/2$ the degree of $h(x)$.

Precomputation
Table T_1 containing $\{f_m\}_L$ for all $m \in PGL_2(\mathbb{F}_p)$.

Table T_2 containing $\{l'\}_L$ for all polynomials l' of degree $r - r_h$.

Table T containing $\{f_m\}_L\{l'\}_L$ for all m and l'. Size: $O(p^{3+r-r_h})$.

Search
Try random $h'(x)$ of degree r_h until a hit is found.

$$\{h'\}_L = \{f_m\}_L\{l'\}_L \Rightarrow f(x) = h'_{m^{-1}}(x)l'_{m^{-1}}(x)$$

Ugly polynomials

Let $f(x) = l(x)h(x)$ with $r_h \geq r/2$ the degree of $h(x)$.

Precomputation
Table T_1 containing $\{f_m\}_L$ for all $m \in PGL_2(\mathbb{F}_p)$.

Table T_2 containing $\{l'\}_L$ for all polynomials l' of degree $r - r_h$.

Table T containing $\{f_m\}_L\{l'\}_L$ for all m and l'. Size: $O(p^{3+r-r_h})$.

Search
Try random $h'(x)$ of degree r_h until a hit is found.

$$\{h'\}_L = \{f_m\}_L\{l'\}_L \Rightarrow f(x) = h'_{m^{-1}}(x)l'_{m^{-1}}(x)$$

Expected run-time: $O(p^{r_h-3})$ trials.

Limited query and the linear prf

polynomial type	search	precomputation	memory
Good	$p^{r-3} r \log p$	$p^3 r \log p$	$p^3 r \log p$
Bad	$p^{r/r'-1} r'' r \log p$	$p^2 r \log p$	$(p/r'') r \log p$
Ugly	$p^{r_h-3} r \log p$	$p^{r-r_h+3} r \log p$	$p^{r-r_h+3} r \log p$

Limited query and the linear prf

polynomial type	search	precomputation	memory
Good	$p^{r-3} r \log p$	$p^3 r \log p$	$p^3 r \log p$
Bad	$p^{r/r'-1} r'' r \log p$	$p^2 r \log p$	$(p/r'') r \log p$
Ugly	$p^{r_h-3} r \log p$	$p^{r-r_h+3} r \log p$	$p^{r-r_h+3} r \log p$

General case run-time: $\tilde{O}(p^3 + p^{r-3})$

Limited query and the linear prf

polynomial type	search	precomputation	memory
Good	$p^{r-3} r \log p$	$p^3 r \log p$	$p^3 r \log p$
Bad	$p^{r/r'-1} r'' r \log p$	$p^2 r \log p$	$(p/r'') r \log p$
Ugly	$p^{r_h-3} r \log p$	$p^{r-r_h+3} r \log p$	$p^{r-r_h+3} r \log p$

General case run-time: $\tilde{O}(p^3 + p^{r-3})$

For $r < 6$ can be lowered to $\tilde{O}(p^{r/2} + p^{r/2})$ by limiting the table.

Limited query and the linear prf

polynomial type	search	precomputation	memory
Good	$p^{r-3} r \log p$	$p^3 r \log p$	$p^3 r \log p$
Bad	$p^{r/r'-1} r'' r \log p$	$p^2 r \log p$	$(p/r'') r \log p$
Ugly	$p^{r_h-3} r \log p$	$p^{r-r_h+3} r \log p$	$p^{r-r_h+3} r \log p$

General case run-time: $\tilde{O}(p^3 + p^{r-3})$

For $r < 6$ can be lowered to $\tilde{O}(p^{r/2} + p^{r/2})$ by limiting the table.

Oracle queries needed : p

Limited query and the linear prf

polynomial type	search	precomputation	memory
Good	$p^{r-3} r \log p$	$p^3 r \log p$	$p^3 r \log p$
Bad	$p^{r/r'-1} r'' r \log p$	$p^2 r \log p$	$(p/r'') r \log p$
Ugly	$p^{r_h-3} r \log p$	$p^{r-r_h+3} r \log p$	$p^{r-r_h+3} r \log p$

General case run-time: $\tilde{O}(p^3 + p^{r-3})$

For $r < 6$ can be lowered to $\tilde{O}(p^{r/2} + p^{r/2})$ by limiting the table.

Oracle queries needed : $p - o(p/L)$.

Limited query and the linear prf

polynomial type	search	precomputation	memory
Good	$p^{r-3} r \log p$	$p^3 r \log p$	$p^3 r \log p$
Bad	$p^{r/r'-1} r'' r \log p$	$p^2 r \log p$	$(p/r'') r \log p$
Ugly	$p^{r_h-3} r \log p$	$p^{r-r_h+3} r \log p$	$p^{r-r_h+3} r \log p$

General case run-time: $\tilde{O}(p^3 + p^{r-3})$

For $r < 6$ can be lowered to $\tilde{O}(p^{r/2} + p^{r/2})$ by limiting the table.

Oracle queries needed : $p - o(p/L)$.

What if oracle queries are limited?

Limited query and the linear prf

Limited query and the linear prf

How many different group actions can we obtain from $M << p$ queries?

Limited query and the linear prf

How many different group actions can we obtain from $M << p$ queries?

Affine linear transformations

Limited query and the linear prf

How many different group actions can we obtain from $M << p$ queries?

Affine linear transformations

$$G = \left\{ \left(\begin{smallmatrix} d & i \\ 0 & 1 \end{smallmatrix} \right) \middle| d \in \mathbb{F}_p^*, i \in \mathbb{F}_p \right\} \leqslant \mathrm{PGL}_2(\mathbb{F}_p).$$

Limited query and the linear prf

How many different group actions can we obtain from $M << p$ queries?

Affine linear transformations

$$G = \left\{ \left(\begin{smallmatrix} d & i \\ 0 & 1 \end{smallmatrix} \right) \middle| d \in \mathbb{F}_p^*, i \in \mathbb{F}_p \right\} \leqslant \mathrm{PGL}_2(\mathbb{F}_p).$$

$\left(\begin{smallmatrix} d & i \\ 0 & 1 \end{smallmatrix} \right) \cdot f = f_{i,d}(x) = f(dx + i)/d^r.$

How many different group actions can we obtain from $M << p$ queries?

Affine linear transformations

$$G = \left\{ \left(\begin{smallmatrix} d & i \\ 0 & 1 \end{smallmatrix} \right) \middle| d \in \mathbb{F}_p^*, i \in \mathbb{F}_p \right\} \leqslant \mathrm{PGL}_2(\mathbb{F}_p).$$

$\left(\begin{smallmatrix} d & i \\ 0 & 1 \end{smallmatrix} \right) \cdot f = f_{i,d}(x) = f(dx + i)/d^r.$

$$\left(\frac{f_{i,d}(x)}{p} \right) = \mathcal{O}_f\,(dx + i) \left(\frac{d}{p} \right)^r.$$

Example

Query $\mathcal{O}_f$ at $[0, M)$.

Example

Query $\mathcal{O}_f$ at $[0, M)$.

$\{f\}_M$ 0 $M-1$

Example

Query $\mathcal{O}_f$ at $[0, M)$.

$\{f\}_M$ 0 $M-1$

$\{f_{i,d}\}_L$

Example

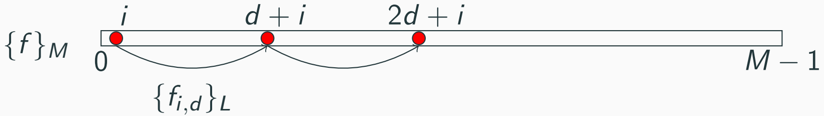
Query $\mathcal{O}_f$ at $[0, M)$.

Example

Query $\mathcal{O}_f$ at $[0, M)$.

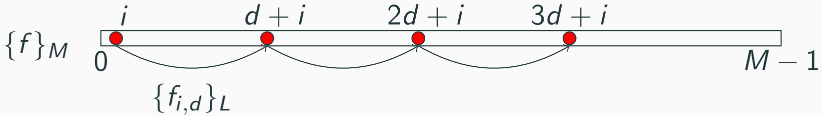
Example

Query $\mathcal{O}_f$ at $[0, M)$.



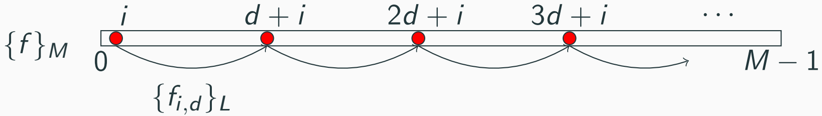
Example

Query $\mathcal{O}_f$ at $[0, M)$.



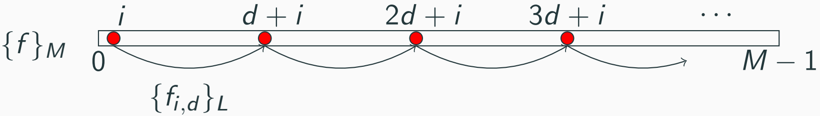
Example

Query $\mathcal{O}_f$ at $[0, M)$.



Example

Query $\mathcal{O}_f$ at $[0, M)$.



In total $\frac{M^2}{L}$ eligible (i, d) values.

Limited query and the linear prf

Precomputation

Query $\mathcal{O}_f$ at $[0, M)$. Make a table T with $O(\frac{M^2}{L})$ sequences.

Limited query and the linear prf

Precomputation
Query $\mathcal{O}_f$ at $[0, M)$. Make a table T with $O(\frac{M^2}{L})$ sequences.

Search
Try random polynomials until a hit is found in the table.

Limited query and the linear prf

Precomputation

Query $\mathcal{O}_f$ at $[0, M)$. Make a table T with $O(\frac{M^2}{L})$ sequences.

Search

Try random polynomials until a hit is found in the table.

Expected run-time: $O(\frac{p^r L}{M^2})$ trials.

Conclusions

Conclusions

- Linear PRF keys are all *weak*. Can we exploit that?

Conclusions

- Linear PRF keys are all *weak*. Can we exploit that?

- Sequences $\{f\}_L$ do not have to be defined as consecutive symbols.

Conclusions

- Linear PRF keys are all *weak*. Can we exploit that?

- Sequences $\{f\}_L$ do not have to be defined as consecutive symbols.

- Cubic yield in the limited query case?

Conclusions

- Linear PRF keys are all *weak*. Can we exploit that?

- Sequences $\{f\}_L$ do not have to be defined as consecutive symbols.

- Cubic yield in the limited query case?

Find $\mathcal{L}, \mathcal{Q} \subseteq \mathbb{P}^1$ and $\mathcal{A} \subseteq PGL_2(\mathbb{F}_p)$ such that

Conclusions

- Linear PRF keys are all *weak*. Can we exploit that?

- Sequences $\{f\}_L$ do not have to be defined as consecutive symbols.

- Cubic yield in the limited query case?

Find $\mathcal{L}, \mathcal{Q} \subseteq \mathbb{P}^1$ and $\mathcal{A} \subseteq PGL_2(\mathbb{F}_p)$ such that

$$\#\mathcal{L} = L, \quad \#\mathcal{Q} = M, \quad \#\mathcal{A} \sim M^3$$

Conclusions

- Linear PRF keys are all *weak*. Can we exploit that?

- Sequences $\{f\}_L$ do not have to be defined as consecutive symbols.

- Cubic yield in the limited query case?

Find $\mathcal{L}, \mathcal{Q} \subseteq \mathbb{P}^1$ and $\mathcal{A} \subseteq PGL_2(\mathbb{F}_p)$ such that

$$\#\mathcal{L} = L, \quad \#\mathcal{Q} = M, \quad \#\mathcal{A} \sim M^3$$

and

$$m\mathcal{L} \subseteq \mathcal{Q} \qquad \text{for all } m \in \mathcal{A}.$$

The end

Thank you for Your attention!

References i

- Ward Beullens, Tim Beyne, Aleksei Udovenko, and Giuseppe Vitto, *Cryptanalysis of the Legendre PRF and generalizations*, Cryptology ePrint Archive, Report 2019/1357, 2019, `https://eprint.iacr.org/2019/1357`.
- Ivan Damgård, *On the randomness of Legendre and Jacobi sequences*, Proceedings of the 8th Annual International Cryptology Conference on Advances in Cryptology (London, UK), CRYPTO '88, Springer-Verlag, 1990, pp. 163–172.
- Dankard Feist, *Legendre pseudo-random function*, 2019, `https://legendreprf.org/bounties`.

- 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 (New York, NY, USA), CCS '16, 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`.
- Alexander Russell and Igor E. Shparlinski, *Classical and quantum function reconstruction via character evaluation*, Journal of Complexity **20** (2004), no. 2-3, 404–422 (English).

Challenges

Ethereum research challenges [Fei19]:

Challenges

Ethereum research challenges [Fei19]:

- 5 Linear Legendre PRF challenges

Challenges

Ethereum research challenges [Fei19]:

- 5 Linear Legendre PRF challenges

- Primes p of $64, 74, 84, 100$ and 148

Challenges

Ethereum research challenges [Fei19]:

- 5 Linear Legendre PRF challenges

- Primes p of $64, 74, 84, 100$ and 148

- Given $M = 2^{20}$ symbols of sequence $\{f\}_M$.

Challenges

Ethereum research challenges [Fei19]:

- 5 Linear Legendre PRF challenges

- Primes p of $64, 74, 84, 100$ and 148

- Given $M = 2^{20}$ symbols of sequence $\{f\}_M$.

- Goal to find $f = x + k$.

Challenges

Ethereum research challenges [Fei19]:

- 5 Linear Legendre PRF challenges

- Primes p of $64, 74, 84, 100$ and 148

- Given $M = 2^{20}$ symbols of sequence $\{f\}_M$.

- Goal to find $f = x + k$.

- For each challenge we used $L = 64$.

Challenges

Ethereum research challenges [Fei19]:

- 5 Linear Legendre PRF challenges

- Primes p of $64, 74, 84, 100$ and 148

- Given $M = 2^{20}$ symbols of sequence $\{f\}_M$.

- Goal to find $f = x + k$.

- For each challenge we used $L = 64$.

- Tables contained 2^{34} sequences.

Challenges

Ethereum research challenges [Fei19]:

- 5 Linear Legendre PRF challenges
- Primes p of $64, 74, 84, 100$ and 148
- Given $M = 2^{20}$ symbols of sequence $\{f\}_M$.
- Goal to find $f = x + k$.
- For each challenge we used $L = 64$.
- Tables contained 2^{34} sequences.
- About 2.2e6 trials per core-second.

Results

Table 1: Results and estimates for solving the Legendre PRF challenges. In all cases $M = 2^{20}$ consecutive queries are given.

Challenge	Prime bit size	Expected # trials	Observed # trials	Expected core-hours	Observed core-hours
0	64	2^{30}	$2^{30.78}$	290 sec	490 sec
1	74	2^{40}	$2^{39.53}$	82	59
2	84	2^{50}	$2^{46.97}$	1.4e5	1.72e4
3	100	2^{66}	-	9.1e9	-
4	148	2^{114}	-	2.5e24	-

Comparison

Khovratovich [Kho19]: Group G with $d = 1$. Table size:$O(1)$.

Beullens et al. [BBUV19]: Group G with $i < d$. Table size $\frac{M^2}{L^2}$.

Us: Full group G. Table size $\frac{M^2}{L}$.

Comparison

Khovratovich [Kho19]: Group G with $d = 1$. Table size:$O(1)$.

Beullens et al. [BBUV19]: Group G with $i < d$. Table size $\frac{M^2}{L^2}$.

Us: Full group G. Table size $\frac{M^2}{L}$.

Algorithm	expected # trials	precomputation	memory
Khovratovich	$\frac{p \log p}{M}$	M	$\log p$
Beullens et al.	$\frac{p \log^2 p}{M^2}$	M^2	$\frac{M^2}{\log p}$
Our algorithm	$\frac{p \log p}{M^2}$	$\frac{M^2}{\log p}$	M^2

General case

Khovratovich [Kho19]:Group G with $d = 1$. Table size:$O(1)$.

Beullens et al. [BBUV19]: Group G with $i < d$. Table size $\frac{p^2}{L^2}$.

Us: Full group $PGL_2(\mathbb{F}_p)$. Table size $p^3 - p$.

General case

good polynomials	search	precomputation	memory
Khovratovich	$p^{r-1} r \log p$	$r \log p$	$r \log p$
Beullens et al.	$p^{r-2} r^2 \log^2 p$	p^2	p^2
Our algorithm	$p^{r-3} r \log p$	p^3	$p^3 r \log p$

bad polynomials	search	precomputation	memory
Khovratovich	$p^{r-1} r \log p$	$r \log p$	$r \log p$
Beullens et al.	$p^{r-2} r^2 \log^2 p$	p^2	$p^{r-r_h} r \log p$
Our algorithm	$p^{r/r'-1} r'' r \log p$	$p^2 r \log p$	$(p/r'') r \log p$

ugly polynomials	search	precomputation	memory
Khovratovich	$p^{r-1} r \log p$	$r \log p$	$r \log p$
Beullens et al.	$p^{r_h} r \log p$	$p^{r-r_h} r \log p$	$p^{r-r_h} r \log p$
Our algorithm	$p^{r_h-3} r \log p$	$p^{r-r_h+3} r \log p$	$p^{r-r_h+3} r \log p$