On the Discrete Logarithm Problem on Algebraic Tori

Rob Granger[1] Fré Vercauteren[2]

[1]granger@cs.bris.ac.uk
University of Bristol, University of Waterloo

[2]fvercaut@esat.kuleuven.be
Katholieke Universiteit Leuven

15th August / CRYPTO 2005

Outline

Motivation

Consider the extension field $\mathbb{F}_{p^n}$.

- Let g be a generator of $\mathbb{F}_{p^n}^{\times}$, and let $h \in \langle g \rangle$
- DLP: Given g and h, compute s such that $g^s = h$

Basic question: Are all extension fields of the same size equally secure?

Motivation

Consider the extension field $\mathbb{F}_{p^n}$.

- Let g be a generator of $\mathbb{F}_{p^n}^{\times}$, and let $h \in \langle g \rangle$
- DLP: Given g and h, compute s such that $g^s = h$

Basic question: Are all extension fields of the same size equally secure?

Motivation

Current approaches to the DLP

Two methods:

- Pohlig-Hellman reduction $+$ square root algorithm
- Index calculus in full multiplicative group $\mathbb{F}_{p^n}^{\times}$

Implications:

- Use prime order subgroup of size $\geq$ 160 bits which does not embed into a subfield
- Choose $\mathbb{F}_{p^n}$ of size $\geq$ 1024 bits

Better question: Do these measures alone ensure security?

Motivation
Current approaches to the DLP

Two methods:

- Pohlig-Hellman reduction $+$ square root algorithm
- Index calculus in full multiplicative group $\mathbb{F}_{p^n}^{\times}$

Implications:

- Use prime order subgroup of size $\geq$ 160 bits which does not embed into a subfield
- Choose $\mathbb{F}_{p^n}$ of size $\geq$ 1024 bits

Better question: Do these measures alone ensure security?

Motivation

A pertinent example

Take two “cryptographically secure” fields:

- $F_1 = \mathbb{F}_{p_1^{29}}$
- $F_2 = \mathbb{F}_{p_2^{30}}$

Assume that:

- $\lfloor 29 \cdot \log_2 p_1 \rfloor = \lfloor 30 \cdot \log_2 p_2 \rfloor = 1024$
- $F_1^\times$ and $F_2^\times$ both contain prime order subgroups $\geq$ 160-bits which do not embed into a proper subfield

Better question still: Are $F_1^\times$ and $F_2^\times$ equally secure?

Motivation

A pertinent example

Take two “cryptographically secure” fields:

- $F_1 = \mathbb{F}_{p_1^{29}}$
- $F_2 = \mathbb{F}_{p_2^{30}}$

Assume that:

- $\lfloor 29 \cdot \log_2 p_1 \rfloor = \lfloor 30 \cdot \log_2 p_2 \rfloor = 1024$
- $F_1^\times$ and $F_2^\times$ both contain prime order subgroups $\geq$ 160-bits which do not embed into a proper subfield

Better question still: Are $F_1^\times$ and $F_2^\times$ equally secure?

Motivation

A pertinent example

Take two “cryptographically secure” fields:

- $F_1 = \mathbb{F}_{p_1^{29}}$
- $F_2 = \mathbb{F}_{p_2^{30}}$

Assume that:

- $\lfloor 29 \cdot \log_2 p_1 \rfloor = \lfloor 30 \cdot \log_2 p_2 \rfloor = 1024$
- $F_1^\times$ and $F_2^\times$ both contain prime order subgroups $\geq$ 160-bits which do not embed into a proper subfield

Better question still: Are $F_1^\times$ and $F_2^\times$ equally secure?

Motivation
A pertinent example

Take two "cryptographically secure" fields:

- $F_1 = \mathbb{F}_{p_1^{29}}$
- $F_2 = \mathbb{F}_{p_2^{30}}$

Assume that:

- $\lfloor 29 \cdot \log_2 p_1 \rfloor = \lfloor 30 \cdot \log_2 p_2 \rfloor = 1024$
- $F_1^\times$ and $F_2^\times$ both contain prime order subgroups $\geq$ 160-bits which do not embed into a proper subfield

Better question still: Are $F_1^\times$ and $F_2^\times$ equally secure?

Motivation

A pertinent example

Take two "cryptographically secure" fields:

- $F_1 = \mathbb{F}_{p_1^{29}}$
- $F_2 = \mathbb{F}_{p_2^{30}}$

Assume that:

- $\lfloor 29 \cdot \log_2 p_1 \rfloor = \lfloor 30 \cdot \log_2 p_2 \rfloor = 1024$
- $F_1^\times$ and $F_2^\times$ both contain prime order subgroups $\geq$ 160-bits which do not embed into a proper subfield

Better question still: Are $F_1^\times$ and $F_2^\times$ equally secure?

Motivation

A pertinent example

Take two "cryptographically secure" fields:

- $F_1 = \mathbb{F}_{p_1^{29}}$
- $F_2 = \mathbb{F}_{p_2^{30}}$

Assume that:

- $\lfloor 29 \cdot \log_2 p_1 \rfloor = \lfloor 30 \cdot \log_2 p_2 \rfloor = 1024$
- $F_1^\times$ and $F_2^\times$ both contain prime order subgroups $\geq$ 160-bits which do not embed into a proper subfield

Better question still: Are $F_1^\times$ and $F_2^\times$ equally secure?

Group decomposition

The identity $|\mathbb{F}_{p^n}^\times| = p^n - 1 = \prod_{d|n} \Phi_d(p)$, with $\Phi_d(\cdot)$ the d-th cyclotomic polynomial $\Longrightarrow$

- $\Phi_d(p) | (p^d - 1)$ and so subgroup of this order embeds into $\mathbb{F}_{p^d} \subset \mathbb{F}_{p^n}$
- subgroup of order $\Phi_n(p)$ can not be attacked by index calculus in proper subfields of $\mathbb{F}_{p^n}$
- subgroup of order $\Phi_n(p)$ is "cryptographically strongest" subgroup of $\mathbb{F}_{p^n}^\times$

In particular, $|\Phi_n(p)| = O(p^{\phi(n)})$.

Group decomposition

The identity $|\mathbb{F}_{p^n}^\times| = p^n - 1 = \prod_{d|n} \Phi_d(p)$, with $\Phi_d(\cdot)$ the d-th cyclotomic polynomial $\Longrightarrow$

- $\Phi_d(p) | (p^d - 1)$ and so subgroup of this order embeds into $\mathbb{F}_{p^d} \subset \mathbb{F}_{p^n}$
- subgroup of order $\Phi_n(p)$ can not be attacked by index calculus in proper subfields of $\mathbb{F}_{p^n}$
- subgroup of order $\Phi_n(p)$ is "cryptographically strongest" subgroup of $\mathbb{F}_{p^n}^\times$

In particular, $|\Phi_n(p)| = O(p^{\phi(n)})$.

Group decomposition

The identity $|\mathbb{F}_{p^n}^\times| = p^n - 1 = \prod_{d|n} \Phi_d(p)$, with $\Phi_d(\cdot)$ the d-th cyclotomic polynomial $\Longrightarrow$

- $\Phi_d(p)|(p^d - 1)$ and so subgroup of this order embeds into $\mathbb{F}_{p^d} \subset \mathbb{F}_{p^n}$
- subgroup of order $\Phi_n(p)$ can not be attacked by index calculus in proper subfields of $\mathbb{F}_{p^n}$
- subgroup of order $\Phi_n(p)$ is "cryptographically strongest" subgroup of $\mathbb{F}_{p^n}^\times$

In particular, $|\Phi_n(p)| = O(p^{\phi(n)})$.

Group decomposition

The identity $|\mathbb{F}_{p^n}^\times| = p^n - 1 = \prod_{d|n} \Phi_d(p)$, with $\Phi_d(\cdot)$ the d-th cyclotomic polynomial $\Longrightarrow$

- $\Phi_d(p)|(p^d - 1)$ and so subgroup of this order embeds into $\mathbb{F}_{p^d} \subset \mathbb{F}_{p^n}$
- subgroup of order $\Phi_n(p)$ can not be attacked by index calculus in proper subfields of $\mathbb{F}_{p^n}$
- subgroup of order $\Phi_n(p)$ is "cryptographically strongest" subgroup of $\mathbb{F}_{p^n}^\times$

In particular, $|\Phi_n(p)| = O(p^{\phi(n)})$.

Group decomposition

The identity $|\mathbb{F}_{p^n}^\times| = p^n - 1 = \prod_{d|n} \Phi_d(p)$, with $\Phi_d(\cdot)$ the d-th cyclotomic polynomial $\Longrightarrow$

- $\Phi_d(p)|(p^d - 1)$ and so subgroup of this order embeds into $\mathbb{F}_{p^d} \subset \mathbb{F}_{p^n}$
- subgroup of order $\Phi_n(p)$ can not be attacked by index calculus in proper subfields of $\mathbb{F}_{p^n}$
- subgroup of order $\Phi_n(p)$ is “cryptographically strongest” subgroup of $\mathbb{F}_{p^n}^\times$

In particular, $|\Phi_n(p)| = O(p^{\phi(n)})$.

Motivation

Back to F_1 and F_2...

Strongest subgroups have orders $O(p_1^{28})$ and $O(p_2^8)$ respectively, so

$$|\log \Phi_{29}(p_1)|/|\log \Phi_{30}(p_2)| \approx 3.5$$

- Hence if there is a native attack in these subgroups then it should be more efficient for F_2 than for F_1.

Question: How can one exploit properties of these subgroups in an attack?

Answer: Interpret them as algebraic tori!

Motivation

Back to F_1 and F_2...

Strongest subgroups have orders $O(p_1^{28})$ and $O(p_2^8)$ respectively, so

$$|\log \Phi_{29}(p_1)|/|\log \Phi_{30}(p_2)| \approx 3.5$$

- Hence if there is a native attack in these subgroups then it should be more efficient for F_2 than for F_1.

Question: How can one exploit properties of these subgroups in an attack?

Answer: Interpret them as algebraic tori!

Motivation

Back to F_1 and F_2...

Strongest subgroups have orders $O(p_1^{28})$ and $O(p_2^8)$ respectively, so

$$|\log \Phi_{29}(p_1)|/|\log \Phi_{30}(p_2)| \approx 3.5$$

- Hence if there is a native attack in these subgroups then it should be more efficient for F_2 than for F_1.

Question: How can one exploit properties of these subgroups in an attack?

Answer: Interpret them as algebraic tori!

Motivation

Back to F_1 and F_2...

Strongest subgroups have orders $O(p_1^{28})$ and $O(p_2^8)$ respectively, so

$$|\log \Phi_{29}(p_1)|/|\log \Phi_{30}(p_2)| \approx 3.5$$

- Hence if there is a native attack in these subgroups then it should be more efficient for F_2 than for F_1.

Question: How can one exploit properties of these subgroups in an attack?

Answer: Interpret them as algebraic tori!

Overview of Results

- First direct index calculus attack on Algebraic Tori
- Practical upper bounds for the DLP in cryptographically relevant tori
- Fields of the same size previously thought to be equally secure are not always so

Overview of Results

- First direct index calculus attack on Algebraic Tori
- Practical upper bounds for the DLP in cryptographically relevant tori
- Fields of the same size previously thought to be equally secure are not always so

Overview of Results

- First direct index calculus attack on Algebraic Tori
- Practical upper bounds for the DLP in cryptographically relevant tori
- Fields of the same size previously thought to be equally secure are not always so

Background on Algebraic Tori

- Consider the degree n extension $K = \mathbb{F}_{q^n}$ of $k = \mathbb{F}_q$.
  - Galois group $\mathrm{Gal}(K/k) = \langle \sigma \rangle$ with $\sigma : K \longrightarrow K : \alpha \mapsto \alpha^q$
  - The norm map of K w.r.t. k is defined as

$$N_{K/k}(\alpha) = \prod_{i=0}^{n-1} \sigma^i(\alpha) = \alpha^{(q^n-1)/(q-1)}$$

  - The $\mathbb{F}_q$-rational points on the algebraic torus T_n are

$$\begin{aligned} T_n(\mathbb{F}_q) &= \{\alpha \in \mathbb{F}_{q^n} \mid N_{K/k_d}(\alpha) = 1 \text{ for all } k \subseteq k_d \subsetneq K\} \\ &= \{\alpha \in \mathbb{F}_{q^n} \mid \alpha^{\Phi_n(q)} = 1\} \end{aligned}$$

where $\Phi_n(x)$ is the n-th cyclotomic polynomial.

Rationality

- T_n is in fact an algebraic variety over $\mathbb{F}_q$ of dimension $\phi(n)$

Definition

T_n is called rational if there exists birational map defined over $\mathbb{F}_q$

$$\psi : \mathbb{A}^{\phi(n)} \longrightarrow T_n$$

- Implication: if T_n rational then compression factor $n/\phi(n)$
- Theorem: T_n is rational for $n = p_1^{e_1} p_2^{e_2}$ with p_i prime

A Brief History

Torus-based systems in the last decade

System	Year	Embedding Field	Compression
LUC	'95	$\mathbb{F}_{p^2}$	2
Gong-Harn	'99	$\mathbb{F}_{p^3}$	3/2
XTR	'00	$\mathbb{F}_{p^6}$	3
XTR-extension	'01	$\mathbb{F}_{p^{6m}}$	3
CEILIDH	'03	$\mathbb{F}_{p^6}$	3
T_{30}	'05	$\mathbb{F}_{p^{30}}$	30/8

- All pairing-based protocols map to tori as well.

Security Assumptions

- $T_n(\mathbb{F}_q) \subset \mathbb{F}_{q^n}^\times \implies$ DLP in $T_n(\mathbb{F}_q)$ is no harder than DLP in $\mathbb{F}_{q^n}^\times$
- The identity $x^n - 1 = \prod_{d|n} \Phi_d(x) \in \mathbb{Z}[x]$, plus Pohlig-Hellman reduction $\implies$

$$\text{DLP in } \{T_d(\mathbb{F}_q)\}_{d|n} \iff \text{DLP in } \mathbb{F}_{q^n}^\times$$

- Since other tori embed into subfields, we deduce

$$\text{DLP in } T_n(\mathbb{F}_q) \iff \text{DLP in } \mathbb{F}_{q^n}^\times$$

- Conclusion: weak torus $\implies$ weak embedding field

Security Assumptions

- $T_n(\mathbb{F}_q) \subset \mathbb{F}_{q^n}^\times \implies$ DLP in $T_n(\mathbb{F}_q)$ is no harder than DLP in $\mathbb{F}_{q^n}^\times$
- The identity $x^n - 1 = \prod_{d|n} \Phi_d(x) \in \mathbb{Z}[x]$, plus Pohlig-Hellman reduction $\implies$

$$\text{DLP in } \{T_d(\mathbb{F}_q)\}_{d|n} \iff \text{DLP in } \mathbb{F}_{q^n}^\times$$

- Since other tori embed into subfields, we deduce

$$\text{DLP in } T_n(\mathbb{F}_q) \iff \text{DLP in } \mathbb{F}_{q^n}^\times$$

- Conclusion: weak torus $\implies$ weak embedding field

Security Assumptions

- $T_n(\mathbb{F}_q) \subset \mathbb{F}_{q^n}^{\times} \implies$ DLP in $T_n(\mathbb{F}_q)$ is no harder than DLP in $\mathbb{F}_{q^n}^{\times}$
- The identity $x^n - 1 = \prod_{d|n} \Phi_d(x) \in \mathbb{Z}[x]$, plus Pohlig-Hellman reduction $\implies$

$$\text{DLP in } \{T_d(\mathbb{F}_q)\}_{d|n} \iff \text{DLP in } \mathbb{F}_{q^n}^{\times}$$

- Since other tori embed into subfields, we deduce

$$\text{DLP in } T_n(\mathbb{F}_q) \iff \text{DLP in } \mathbb{F}_{q^n}^{\times}$$

- Conclusion: weak torus $\implies$ weak embedding field

Security Assumptions

- $T_n(\mathbb{F}_q) \subset \mathbb{F}_{q^n}^\times \implies$ DLP in $T_n(\mathbb{F}_q)$ is no harder than DLP in $\mathbb{F}_{q^n}^\times$
- The identity $x^n - 1 = \prod_{d|n} \Phi_d(x) \in \mathbb{Z}[x]$, plus Pohlig-Hellman reduction $\implies$

$$\text{DLP in } \{T_d(\mathbb{F}_q)\}_{d|n} \iff \text{DLP in } \mathbb{F}_{q^n}^\times$$

- Since other tori embed into subfields, we deduce

$$\text{DLP in } T_n(\mathbb{F}_q) \iff \text{DLP in } \mathbb{F}_{q^n}^\times$$

- Conclusion: weak torus $\implies$ weak embedding field

A Native Algorithm?

- Observation: Finite field embedding introduces redundancy in an attack, so ideally we want to work directly on the torus. How?
- Use affine representation of T_n!
- Problem: T_n not a UFD, so no natural notion of smoothness
- Solution: Impose a notion of smoothness algebraically (Gaudry 2004)
  - Define a factor base in T_n which generates 'enough' of T_n, and which also permits an algebraic decomposition
  - Then use standard index calculus technique

A Native Algorithm?

- Observation: Finite field embedding introduces redundancy in an attack, so ideally we want to work directly on the torus. How?
- Use affine representation of T_n!
- Problem: T_n not a UFD, so no natural notion of smoothness
- Solution: Impose a notion of smoothness algebraically (Gaudry 2004)
  - Define a factor base in T_n which generates ‘enough’ of T_n, and which also permits an algebraic decomposition
  - Then use standard index calculus technique

A Native Algorithm?

- Observation: Finite field embedding introduces redundancy in an attack, so ideally we want to work directly on the torus. How?
- Use affine representation of T_n!
- Problem: T_n not a UFD, so no natural notion of smoothness
- Solution: Impose a notion of smoothness algebraically (Gaudry 2004)
  - Define a factor base in T_n which generates 'enough' of T_n, and which also permits an algebraic decomposition
  - Then use standard index calculus technique

A Native Algorithm?

- Observation: Finite field embedding introduces redundancy in an attack, so ideally we want to work directly on the torus. How?
- Use affine representation of T_n!
- Problem: T_n not a UFD, so no natural notion of smoothness
- Solution: Impose a notion of smoothness algebraically (Gaudry 2004)
  - Define a factor base in T_n which generates 'enough' of T_n, and which also permits an algebraic decomposition
  - Then use standard index calculus technique

A Native Algorithm?

- Observation: Finite field embedding introduces redundancy in an attack, so ideally we want to work directly on the torus. How?
- Use affine representation of T_n!
- Problem: T_n not a UFD, so no natural notion of smoothness
- Solution: Impose a notion of smoothness algebraically (Gaudry 2004)
  - Define a factor base in T_n which generates 'enough' of T_n, and which also permits an algebraic decomposition
  - Then use standard index calculus technique

A Native Algorithm?

- Observation: Finite field embedding introduces redundancy in an attack, so ideally we want to work directly on the torus. How?
- Use affine representation of T_n!
- Problem: T_n not a UFD, so no natural notion of smoothness
- Solution: Impose a notion of smoothness algebraically (Gaudry 2004)
  - Define a factor base in T_n which generates 'enough' of T_n, and which also permits an algebraic decomposition
  - Then use standard index calculus technique

The Torus $T_2(\mathbb{F}_{q^m})$

- Let $\mathbb{F}_{q^{2m}} = \mathbb{F}_{q^m}[\gamma]/(\gamma^2 - \delta)$, with $\delta \in \mathbb{F}_{q^m} \setminus \mathbb{F}_q$ non-square (q odd)
- For $\alpha = \alpha_0 + \gamma\alpha_1 \in \mathbb{F}_{q^{2m}}$, the norm is

$$N_{K/k}(\alpha) = \alpha \cdot \sigma(\alpha) = (\alpha_0 + \gamma\alpha_1)(\alpha_0 - \gamma\alpha_1) = \alpha_0^2 - \delta\alpha_1^2$$

- By definition, the torus $T_2(\mathbb{F}_{q^m})$ is given by

$$T_2(\mathbb{F}_{q^m}) = \{x + \gamma y \in \mathbb{F}_{q^{2m}} : x^2 - \delta y^2 = 1\}.$$

- T_2 is of dimension 1, $\# T_2(\mathbb{F}_{q^m}) = q^m + 1$ and rational, with

$$\psi : \mathbb{A}(\mathbb{F}_{q^m}) \longrightarrow T_2(\mathbb{F}_{q^m}) : z \mapsto \frac{z - \gamma}{z + \gamma}$$

Index Calculus for $T_2(\mathbb{F}_{q^m})$

- DLP: let $\langle P \rangle = T_2(\mathbb{F}_{q^m})$ and $Q = P^s$, compute s
- Let $\mathbb{F}_{q^m} = \mathbb{F}_q[t]/(f(t))$ with $f \in F_q[t]$ irreducible of degree m
- Decomposition base containing q elements:

$$\mathcal{F} = \left\{ \frac{a - \gamma}{a + \gamma} : a \in \mathbb{F}_q \right\} \subset T_2(\mathbb{F}_{q^m})$$

- Index calculus:
  - Generate random combinations $R = P^j \cdot Q^k$
  - Try to decompose R over $\mathcal{F}$
  - Collect more than q relations and find s using linear algebra

Decomposition for $T_2(\mathbb{F}_{q^m})$

- Since $(\mathrm{Res}_{\mathbb{F}_{q^m}/\mathbb{F}_q} T_2)(\mathbb{F}_q)$ is m-dimensional, given $R = P^j \cdot Q^k \in T_2(\mathbb{F}_{q^m})$, want to find m elements $P_i \in \mathcal{F}$ with

$$P_1 \cdot \cdots \cdot P_m = R$$

- Using the rationality of T_2, we can equivalently write

$$\prod_{i=1}^{m} \left(\frac{a_i - \gamma}{a_i + \gamma} \right) = \frac{r - \gamma}{r + \gamma}$$

- Note: $a_i \in \mathbb{F}_q$ are unknown, $r \in \mathbb{F}_{q^m}$ is known

Decomposition for $T_2(\mathbb{F}_{q^m})$

- Denote $\sigma_i(a_1, \ldots, a_m)$ the i-th symmetric polynomial, then

$$\frac{\sigma_m - \sigma_{m-1}\gamma + \cdots + (-1)^m\gamma^m}{\sigma_m + \sigma_{m-1}\gamma + \cdots + \gamma^m} = \frac{r-\gamma}{r+\gamma}$$

- Since $\gamma^2 = \delta \in \mathbb{F}_{q^m}$, we finally obtain

$$\frac{b_0(\sigma_1, \ldots, \sigma_m) - b_1(\sigma_1, \ldots, \sigma_m)\gamma}{b_0(\sigma_1, \ldots, \sigma_m) + b_1(\sigma_1, \ldots, \sigma_m)\gamma} = \frac{r-\gamma}{r+\gamma}$$

- Polynomials b_0 and b_1 are linear in σ_i for $i = 1, \ldots, m$
- Using affine representation, we obtain 1 equation over $\mathbb{F}_{q^m}$

$$b_0(\sigma_1, \ldots, \sigma_m) - b_1(\sigma_1, \ldots, \sigma_m)r = 0$$

Decomposition for $T_2(\mathbb{F}_{q^m})$

- Writing out on basis of $\{1, t, \ldots, t^{m-1}\}$ of $\mathbb{F}_{q^m}$ gives

  m linear equations over $\mathbb{F}_q$ in the m unknowns σ_i

- Factor $p(x) := x^m - \sigma_1 x^{m-1} + \sigma_2 x^{m-2} - \cdots + (-1)^m \sigma_m$ over $\mathbb{F}_q$

  If $p(x)$ splits completely, found a relation!

- Note: $p(x)$ splits with probability $1/m!$.

Complexity of T_2-algorithm

- Complexity of the T_2-algorithm to compute DLOGs in $T_2(\mathbb{F}_{q^m})$ is

$$O(m! \cdot q \cdot (m^3 + m^2 \log q) + m^3 q^2) \text{ operations in } \mathbb{F}_q$$

- Index calculus in $\mathbb{F}_{q^{2m}}^{\times}$ runs in time $L_{q^{2m}}(1/2, c)$
- For $q \simeq m!$, the T_2 algorithm runs in time $L_{q^m}(1/2, c')$

The Torus $T_6(\mathbb{F}_{q^m})$

- For $q^m \equiv 2 \text{ or } 5 \pmod 9$, let $x = \zeta_3$ and $y = \zeta_9 + \zeta_9^{-1}$
- $\mathbb{F}_{q^{3m}} = \mathbb{F}_{q^m}[y]$ and $\mathbb{F}_{q^{6m}} = \mathbb{F}_{q^{3m}}[x]$
- By definition, the $\mathbb{F}_{q^m}$-rational points on T_6 are

$$T_6(\mathbb{F}_{q^m}) = \{\alpha \in \mathbb{F}_{q^{6m}} \mid N_{\mathbb{F}_{q^{6m}}/\mathbb{F}_{q^{3m}}}(\alpha) = 1,\ N_{\mathbb{F}_{q^{6m}}/\mathbb{F}_{q^{2m}}}(\alpha) = 1\}$$

- T_6 has dimension 2, $\#T_6(\mathbb{F}_{q^m}) = \Phi_6(q^m) = q^{2m} - q^m + 1$
- Birational map $\psi : \mathbb{A}^2(\mathbb{F}_{q^m}) \longrightarrow T_6(\mathbb{F}_{q^m})$

$$\psi(\alpha_1, \alpha_2) = \frac{1 + \alpha_1 y + \alpha_2(y^2 - 2) + (1 - \alpha_1^2 - \alpha_2^2 + \alpha_1\alpha_2)x}{1 + \alpha_1 y + \alpha_2(y^2 - 2) + (1 - \alpha_1^2 - \alpha_2^2 + \alpha_1\alpha_2)x^2}$$

Index Calculus for $T_6(\mathbb{F}_{q^m})$

- DLP: let $\langle P \rangle = T_6(\mathbb{F}_{q^m})$ and $Q = P^s$, find s
- Let $\mathbb{F}_{q^m} = \mathbb{F}_q[t]/(f(t))$ with $f \in F_q[t]$ irreducible of degree m
- Decomposition base consists of $\psi(at, 0)$ for $a \in \mathbb{F}_q$

$$\mathcal{F} = \left\{ \frac{1 + (at)y + (1 - (at)^2)x}{1 + (at)y + (1 - (at)^2)x^2} : a \in \mathbb{F}_q \right\}$$

- Since $(\mathrm{Res}_{\mathbb{F}_{q^m}/\mathbb{F}_q} T_6)(\mathbb{F}_q)$ is $2m$-dimensional, to decompose $R = P^j \cdot Q^k$, want to find $P_1, \ldots, P_{2m} \in \mathcal{F}$ such that

$$P_1 \cdot \cdots \cdot P_{2m} = R$$

Decomposition for $T_6(\mathbb{F}_{q^m})$

- Let $P_i = \psi(a_i t, 0)$ with $a_i \in \mathbb{F}_q$, then

$$\prod_{i=1}^{2m} \left(\frac{1 + (a_i t)y + (1 - (a_i t)^2)x}{1 + (a_i t)y + (1 - (a_i t)^2)x^2} \right) = R = \psi(r_1, r_2)$$

- Rewriting this using elementary symmetric polynomials σ_i gives

$$\frac{b_0 + b_1 y + b_2(y^2 - 2)}{c_0 + c_1 y + c_2(y^2 - 2)} = \frac{1 + r_1 y + r_2(y^2 - 2)}{1 - r_1^2 - r_2^2 + r_1 r_2}$$

- b_k and c_k are quadratic polynomials in the σ_i for $i = 1, \ldots 2m$

Decomposition for $T_6(\mathbb{F}_{q^m})$

- Writing out on basis of $\{1, t, \ldots, t^{m-1}\}$ of $\mathbb{F}_{q^m}$ gives $3m$ quadratic equations over $\mathbb{F}_q$ in the $2m$ unknowns σ_i
- Use Gröbner basis algorithms to compute the solutions σ_i
- Factor $p(x) := x^{2m} - \sigma_1 x^{2m-1} + \sigma_2 x^{2m-2} - \cdots + (-1)^{2m}\sigma_{2m}$ over $\mathbb{F}_q$

If $p(x)$ splits completely, found a relation!

- Note: $p(x)$ splits with probability $1/(2m)!$

Complexity of T_6-algorithm

- Complexity of the T_6-algorithm to compute DLOGs in $T_6(\mathbb{F}_{q^m})$ is

$$O((2m)! \cdot q \cdot (2^{12m} + 3^{2m} \log q) + m^3 q^2) \text{ operations in } \mathbb{F}_q$$

- Index calculus in $F^{\times}_{q^{6m}}$ runs in $L_{q^{6m}}(1/2, c)$
- For $q \simeq (2m)!2^{12m}$, the T_6 algorithm runs in time $L_{q^m}(1/2, c')$

T_6 Experimental Results

$\log_2$ of expected running times (s) of the T_6-algorithm and Pollard-Rho in a subgroup of size 2^{160}

$\log_2 \lvert\mathbb{F}_{p^{6m}}\rvert$	$\log_2 \lvert T_6(\mathbb{F}_{p^m})\rvert$	ρ	m = 1	m = 2	m = 3	m = 4	m = 5
200	67	18	25	**18**	**14**	**20**	**29**
300	100	34	42	36	**21**	**24**	**32**
400	134	52	59	54	**32**	**29**	**36**
500	167	66	75	71	44	**33**	**39**
600	200	66	93	88	55	40	**42**
700	234	66	109	105	67	48	46
800	267	66	127	122	78	57	51
900	300	68	144	139	90	65	56
1000	334	69	161	156	101	74	60

Application to $T_{30}(\mathbb{F}_p)$

A $T_{30}(\mathbb{F}_p)$ cryptosystem was proposed at EUROCRYPT 2005 with the following parameters:

- $p = 2527138379$, and so $|\mathbb{F}_{p^{30}}| \approx 2^{937}$
- $T_{30}(\mathbb{F}_p)$ contains a subgroup of order $\approx 2^{160}$

Since $\Phi_{30}(x)|\Phi_6(x^5)$, we have the inclusion $T_{30}(\mathbb{F}_p) \subset T_6(\mathbb{F}_{p^5})$, and hence one can attack the former via the latter.

Question: What does this mean in practice?

Application to $T_{30}(\mathbb{F}_p)$

A $T_{30}(\mathbb{F}_p)$ cryptosystem was proposed at EUROCRYPT 2005 with the following parameters:

- $p = 2527138379$, and so $|\mathbb{F}_{p^{30}}| \approx 2^{937}$
- $T_{30}(\mathbb{F}_p)$ contains a subgroup of order $\approx 2^{160}$

Since $\Phi_{30}(x)|\Phi_6(x^5)$, we have the inclusion $T_{30}(\mathbb{F}_p) \subset T_6(\mathbb{F}_{p^5})$, and hence one can attack the former via the latter.

Question: What does this mean in practice?

Application to $T_{30}(\mathbb{F}_p)$

To solve the DLP in $T_{30}(\mathbb{F}_p)$:

- Pollard rho time is 2^{68} seconds
- Our time is 2^{58} seconds

Note:

- This is with a non-optimised Magma implementation
- Does not use the large prime variants of Thériault, Gaudry-Thomé-Thériault and Nagao

Conclusion:

- One should increase the base field size to thwart attack
- For this field size, possibly no advantage of T_{30} over T_6

Summary

- New algorithm to solve DLP in $T_2(\mathbb{F}_{q^m})$ and $T_6(\mathbb{F}_{q^m})$
- Exploits compact representation of algebraic tori
- Upper bounds on the hardness of the DLP in $\mathbb{F}_{q^m}$ for $m > 1$
- Security of the DLP in $\mathbb{F}_{q^{30}}$ is questionable via $T_6(\mathbb{F}_{q^5})$
- Does not influence security of MNT curves over $\mathbb{F}_p$
- Does not influence security of XTR over $\mathbb{F}_p$

Future work

- Complexity of general algorithm with Diem's choice of factor base
- Possibility of using $2m$ disjoint factor bases

$$P_1 \cdot \cdots \cdot P_{2m} = R \quad \text{with } P_i \in \mathcal{F}_i, \mathcal{F}_i \cap \mathcal{F}_j = \emptyset \text{ for } i \neq j$$

- Speeding up repeated Gröbner basis computation?