Politechnika Warszawska

WYDZIAŁ ELEKTRONIKI
I TECHNIK INFORMACYJNYCH

Instytut Cyberbezpieczeństwa

Raport z postępu pracowni dyplomowej

na kierunku Telekomunikacja
w specjalności Techniki Teleinformatyczne

Akceleracja sprzętowa kryptoanalizy algorytmów kryptograficznych

Andrzej Tłomak
Numer albumu 311450

promotor
dr. hab. inż. Mariusz Rawski

WARSZAWA 2024

Akceleracja sprzętowa kryptoanalizy algorytmów kryptograficznych

Streszczenie. Celem tego etapu było zapoznanie się z literaturą opisującą aktualny State of the Art kryptoanalizy systemów opartych o krzywe eliptyczne w ciałach skończonych, zapoznanie się z teorią oraz podstawami matematycznymi zagadnienie krzywych eliptycznych w kryptografii oraz przygotowanie środowiska do pracy z wykorzystaniem technologii CUDA.

Słowa kluczowe: Krzywe eliptyczne, Kryptografia, Kryptoanaliza, CUDA, FPGA, Algorytm rho Pollard'a

Hardware acceleration of cryptoanalysis of cryptograhpic algorithms

Abstract. The objective of this phase included a review of the literature describing the current State-of-Art in cryptoanalysis of systems based on Elliptic curves in Finite Fields. It involved getting deeper knowledge of theory and mathematical fundation of Elliptic curves as well as setting up an environment to develop implementation utillizing CUDA technology.

Keywords: Elliptic curves, Cryptography, Cryptoanalysis, CUDA, FPGA, rho Pollard algorithm

załącznik nr 3 do zarządzenia
nr 28 /2016 Rektora PW

Politechnika Warszawska

……..
miejscowość i data

……..
imię i nazwisko studenta
……..
numer albumu
…….…….
kierunek studiów

OŚWIADCZENIE

Świadomy/-a odpowiedzialności karnej za składanie fałszywych zeznań oświadczam, że niniejsza praca dyplomowa została napisana przeze mnie samodzielnie, pod opieką kierującego pracą dyplomową.

Jednocześnie oświadczam, że:

- niniejsza praca dyplomowa nie narusza praw autorskich w rozumieniu ustawy z dnia 4 lutego 1994 roku o prawie autorskim i prawach pokrewnych (Dz.U. z 2006 r. Nr 90, poz. 631 z późn. zm.) oraz dóbr osobistych chronionych prawem cywilnym,
- niniejsza praca dyplomowa nie zawiera danych i informacji, które uzyskałem/-am w sposób niedozwolony,
- niniejsza praca dyplomowa nie była wcześniej podstawą żadnej innej urzędowej procedury związanej z nadawaniem dyplomów lub tytułów zawodowych,
- wszystkie informacje umieszczone w niniejszej pracy, uzyskane ze źródeł pisanych i elektronicznych, zostały udokumentowane w wykazie literatury odpowiednimi odnośnikami,
- znam regulacje prawne Politechniki Warszawskiej w sprawie zarządzania prawami autorskimi i prawami pokrewnymi, prawami własności przemysłowej oraz zasadami komercjalizacji.

Oświadczam, że treść pracy dyplomowej w wersji drukowanej, treść pracy dyplomowej zawartej na nośniku elektronicznym (płycie kompaktowej) oraz treść pracy dyplomowej w module APD systemu USOS są identyczne.

.....
czytelny podpis studenta

Spis treści

1. Wprowadzenie

Celem tej pracy była implementacja akecleracji sprzętowej algorytmu do kryptoanalizy kryptosystemów opartych o problem logarytmu dyskretnego. Jednym z prostszych sposobów akceleracji algorytmów które pozwalają na ich zrównoleglenie, jest wykorzystanie procesorów graficznych GPGPU. Praca ta skupia się na wykorzystaniu framework'u Nvidia CUDA wraz z kartą graficzną Nvidia GTX 2070 Super do przyśpieszenia kryptoanalizy krzywej ECCp-79 z listy Certicom.

2. Wstęp teoretyczny

2.1. Ciało skończone

Ciało skończone jest to ciało ze skończoną liczbą elementów. Aby ciało spełniało wszystkie założenia, musi ono być rzędu p gdzie p jest liczbą pierwszą. Dopuszczalne są również ciała rzędu p^n gdzie p to liczba pierwsza oraz $n \geqslant 1$. W zastosowaniach kryptograficznych, często stosowanym ciałem jest tzw. ciało binarne postaci $\mathbb{F}_{2^n}$. Challange ECC Certicom dopuszcza rozwiązania zarówno na ciele binarnym jak i ciele rzędu liczba pierwsza. W swojej pracy skupiłem się wyłącznie na ciałach rzędu p liczba pierwsza.

2.2. Grupa

2.3. Problem logarytmu dyskretnego

Problem logarytmu dyskretnego (**DLP**) jest podstawą wielu kryptosystemów. Jednymi z bardziej znanych są kryptosystem ElGamala oraz protokół wymiany kluczy Diffie-Hellmana'a.
Problem logarytmu dyskretnego można zdefiniować na grupach cyklicznych. zarówno na grupie multiplikatywnej $(\mathbb{G}, \cdot)$ oraz grupie addytywnej $(\mathbb{G}, +)$, przy odpowiednim zdefiniowaniu działań grupowych.

Definicja 1. *Jeżeli G jest grupą cykliczną a γ jej generatorem, to logarytmem dyskretnym elementu $\alpha \in G$ nazywamy najmniejszą nieujemną liczbę całkowitą x taką, że:*

$$x = \log_\gamma \alpha$$

operacji dodawania na krzywej eliptycznej $(\mathbb{E}, +)$ [3].

2.4. DLP w grupie multiplikatywnej

Jeżeli $\mathbb{G}$ to (skończona) grupa multiplikatywna, $\alpha \in \mathbb{G}$ to element rzędu n oraz $\beta \in <\alpha>$ (jest w podgrupie generowanej przez α), to uważane za problematyczne jest znalezienie takiej liczby a:

$$a \in \mathbb{Z} \text{ oraz } 0 \leq a \leq n-1$$

że:

$$\alpha^a = \beta$$

Liczbę a można przedstawić jako:

$$\log_\alpha \beta$$

2.5. DLP w grupie addytywnej

W przypadku krypografii opartej o krzywe eliptyczne, DLP dotyczy grupy addytywnej $(\mathbb{E}, +)$ zdefiniowanej na krzywej eliptycznej. Niech α jest rzędu n. W takim przypadku, ponieważ operacją na grupie jest dodawanie modulo n, to działanie potęgowania przedstawia się jako:

$$\alpha \cdot a = \beta \pmod{n}$$

Przy odpowiednim wyborze grupy addytywnej, rozwiązanie problemu logarytmu dyskretnego, tj. znalezienie a, jest trudne [2][3].

2.6. Krzywe eliptyczne

Krzywą eliptyczną nieosobliwą nad ciałem $\mathbb{K}$ o charakterystyce różnej od 2 i 3 definiuje się za jako zbiór rozwiązań $(x, y) \in \mathbb{R} \times \mathbb{R}$ równiania: [3]

$$y^2 = x^3 + ax + b$$

przy założeniu, że stałe a, b takie, że:

$$4a^3 + 27b^2 \neq 0$$

Jest to tak zwana forma *W*eierstrassa krzywej eliptycznej.

2.6.1. Krzywe eliptyczne na liczbach rzeczywistych

Krzywe eliptyczne zdefiniowane na liczbach rzeczywistych nie są kluczowe w systemach kryptograficznych[2][3], ale takie ustawienia pozwalają na prostsze przedstawienie niektórych zagadnień np. dodawnie punktów na krzywej.

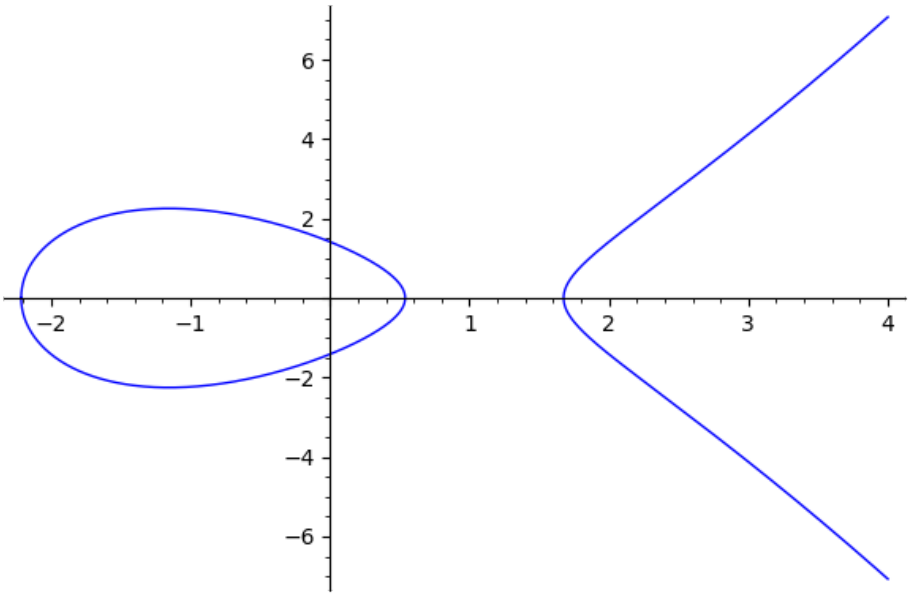
Krzywe eliptyczne na ciele skończonym

Krzywa eliptyczna na ciele skończonym jest stosowana w kryptografii. Z powodu charakterystyki ciała, jej wykres nie przypomina krzywej na liczbach rzeczywistych. Krzywa taka składa się z punktów, których współrzędne należą do ciała na którym jest opisana. Wszystkie operacje na krzywej, takie jak dodawanie, wykonuje się również z zastosowaniem operacji modulo rzędu ciała.

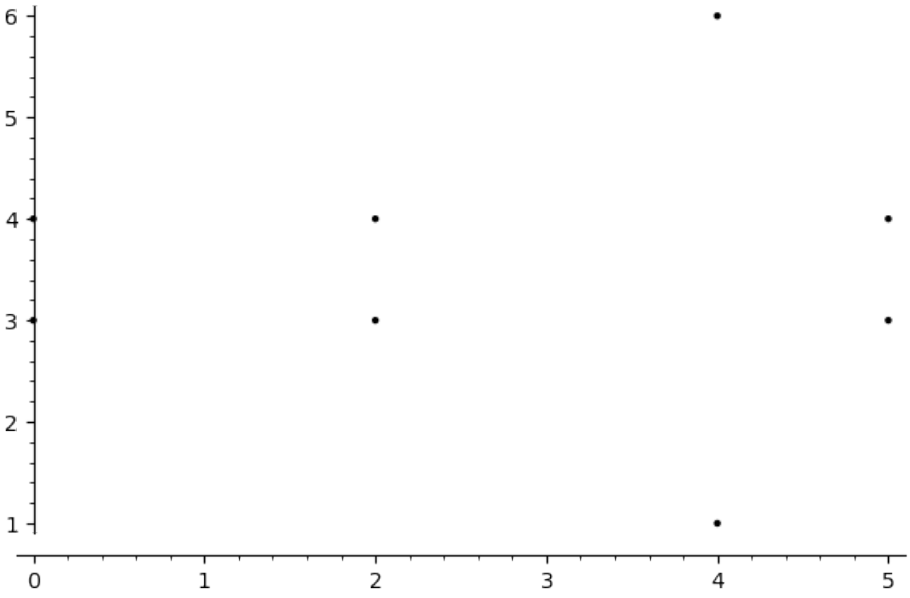
2.6.2. Dodawanie punktów na krzywej eliptycznej

Przedstwienie krzywej eliptycznej na ciele liczb rzeczywistych, umożliwia proste zwizualizowanie geometrycznej interpretacji dodawania punktów leżących na krzywej.

Geometryczne dodawanie punktów na krzywej eliptycznej polega na połączeniu dwóch punktów P i Q prostopadłą linią, która przecina krzywą w trzecim punkcie, R'. Następnie, wynikowy punkt R, będący sumą $P + QP + Q$, znajdujemy przez odbicie punktu R' względem osi x. W przypadku dublowania punktu, czyli dodawania punktu PP

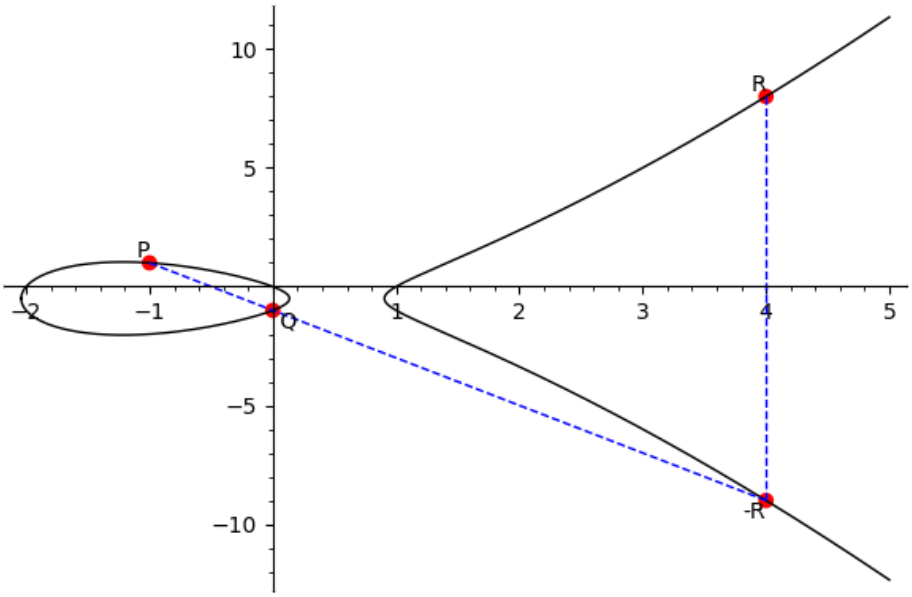


Rys. 2.1. Krzywa eliptyczna $y^2 = x^3 - 4 + 2$



Rys. 2.2. Krzywa eliptyczna $y^2 = x^3 - 4 + 2$ nad $GF(7)$

do siebie samego, rysujemy styczną do krzywej w punkcie P, która przecina krzywą w nowym punkcie. Odbicie tego punktu względem osi x daje nam wynik $2P$.
Kod w SageMath użyty do wizualizacji dodawania: listing **??**



Rys. 2.3. P + Q na krzywej eliptycznej $y^2 + y = x^3 - x^2 + 2x$

2.6.3. Dodawanie punktów na krzywej zdefiniowanej na ciele skończonym

Dodawanie punktów krzywej eliptycznej na ciele skończonym nie ma przejrzyjstej reprezentacji geometrycznej. W tym celu stosuje się podejście analityczne. Wtedy, dodawanie wygląda w następujący sposób:

1. Przypadek, gdy $P \neq Q$:

$$\lambda = \frac{y_2 - y_1}{x_2 - x_1}, \tag{1}$$

$$x_3 = \lambda^2 - x_1 - x_2, \tag{2}$$

$$y_3 = \lambda(x_1 - x_3) - y_1 \tag{3}$$

2. Przypadek, gdy $P = Q$:

$$\lambda = \frac{3x_1^2 + a}{2y_1},$$

$$x_3 = \lambda^2 - 2x_1,$$

$$y_3 = \lambda(x_1 - x_3) - y_1$$

Dodawanie punktów w SageMath listing **??**

3. State of Art

3.1. GPU

Procesory graficzne są dedykowane do wykonywania wielu równoległych obliczeń. Dzięki temu, są bardzo wydajne w zadaniach które można łatwo zrónoleglić. Wiele algorytmów do kryptoanalizy pozwala na przetwarzanie równoległe, w szczególności algorytm **rho-Pollarda**.

Solving Discrete Logarithms in Smooth-Order Groups with CUDA

W roku 2012 na karcie graficznej NVIDIA Tesla M2050 osiągnięto wydajność na poziomie 51.9 miliona operacji mnożenia modularnego 768-bit na sekundę. Implementacja opierała się głównie na języku C z CUDA framework wraz z jednostkowymi segmentami w języku PTX który jest zbiorem instrukcji dla CUDA GPU. Praca ma dla mnie szczególną na tym etapie, ponieważ razem z pracą udostępniono kod implementacji na prawach open-source, dodatkowo opisuje ograniczenia i założenia jakie należy uwzględnić przy implementacji algorytmu rho-Pollarda na GPU[4].

ECC2K-130 on NVIDIA GPUs

Artykuł opisuje implementację algorytmu rho-Pollarda na karcie graficznej NVIDIA GTX 295. Autorzy wybrali krzywą Koblitza ECC2K-130. Opisano decyzje związane z wyborem bazy (w tym przypadku wybrano bazę normalną). Przedstawiono również szczegóły związane z zarządzaniem pamięcią oraz problem związany z DRAM'em karty (przy pełnej utylizacji GPU w pamięci brakowało miejsca na input) Wynik: Średnio obliczenie ECDLP na tej krzywej zajełoby 2 lata przy 534 kartach.

3.2. FPGA

Solving Discrete Logarithms in Smooth-Order Groups with CUDA

W 2014 opublikowano pracę przedstawiającą implementację FPGA na platformie Virtex-6. dedykowaną do rozwiązania logarytmu dyskretnego na 113-bitowej krzywej Koblitza. Opisano zastosowane zabiegi poprawiające optymalizację, oraz design poszczególnych moduł Na przykład w celu lepszej optymalizacji, wykorzystano bazę normalną F_{2^m} w jednym z modułów do liczenia automorfizmu punktów. Wynik po ekstrapolacji to 28 dni na rozwiązanie logarytmu na krzywej Koblitza 113 bit.

3.3. CPU

CPU nie są najwydajniejszą architekturą do wykonywania równoległych obliczeń. Zazwyczaj charakteryzują się znacznie wydajniejszymi jednostkami obliczeniowymi (rdze-

niami) niż na przykład GPGPU, ale jest ich również znacznie mnniej niż w GPGPU. CPUs są najlepiej przystosowane do przetwarzania potokowego.

A Review on solving ECDLP over Large Finite Field using Parallel Pollard's Rho (p) Method

Praca przedstawia wyniki czasowe przy obliczaniu ECDLP na ciele skończonym rzędu p do 85-bitów. Zastosowano do tego cluster CPU o 256 rdzeniach octa-core. Artykuł również jest interesujący ponieważ zwięźle opsuje background matematyczny oraz przejrzyście przedstawia wersję równoległą algorytmu rho Pollarda[1]. Wynik to 52 godziny dla krzywej na ciele rozmiaru p = 85-bitów.

Bibliografia

[1] Kaushal A Chavan, Indivar Gupta i Dinesh B Kulkarni. „A Review on Solving ECDLP over Large Finite Field Using Parallel Pollard's Rho (p) Method". W: 18 (2), s. 1–11. DOI: 10.9790/0661-1802040111. URL: www.iosrjournals.org.

[2] Andrzej Chrzęszczyk. „Algorytmy teorii liczb i kryptografii". W: btc, 2010, s. 279.

[3] Maura B. Paterson Dauglas R. Stinson. „Kryptografia w teorii i praktyce, Wydanie IV". W: PWN, 2021, s. 274.

[4] Ryan Henry i Ian Goldberg. „Solving Discrete Logarithms in Smooth-Order Groups with CUDA 1". W: (). URL: http://cacr.uwaterloo.ca/.

Wykaz symboli i skrótów

EiTI – Wydział Elektroniki i Technik Informacyjnych
PW – Politechnika Warszawska
FPGA – Field Programmable Gates Array
DLP – Discrete Logarithm Problem
GF – Galois Field (ciało skończone)

Spis rysunków

Spis tabel

Spis wydruków

Spis załączników