High-Performance Cryptology on GPUs

Joppe W. Bos

Microsoft Research, Redmond

GTC 2013 – Session S3018

Joint work with the Laboratory for Cryptologic Algorithms, EPFL

Cryptology

$\swarrow \qquad \searrow$

Cryptography

$\downarrow$

"secure communication in the presence of third parties"

Cryptanalysis

$\downarrow$

"obtaining the original meaning of encrypted data without using the corresponding secret material"

Cryptology

$\swarrow$ $\searrow$

Cryptography	Cryptanalysis
$\downarrow$	$\downarrow$
"secure communication in the presence of third parties"	*"obtaining the original meaning of encrypted data without using the corresponding secret material"*

Three main areas

- Public-key cryptography: e.g. RSA, (EC)DSA, (EC)DH
- Symmetric cryptography: e.g. AES
- Cryptographic hash functions: e.g. SHA-256, SHA-512, SHA-3

Motivation

Can we use the parallel compute power of GPUs to

- enhance the performance of cryptographic primitives
  - high-throughput
  - low-latency
- speed-up the security assessment of these cryptographic primitives

Motivation

Can we use the parallel compute power of GPUs to

- enhance the performance of cryptographic primitives
  - high-throughput
  - low-latency
- speed-up the security assessment of these cryptographic primitives

We have done similar experiments before...

[1] J.W. Bos, M.E. Kaihara, T. Kleinjung, A.K. Lenstra, P.L. Montgomery: Solving a 112-bit Prime Elliptic Curve Discrete Logarithm Problem on Game Consoles using Sloppy Reduction. In International Journal of Applied Cryptography, 2012


[free busy full down disabled absent]
$ cluster-nodes
Master: M
Slave: S

High-Throughput Hashing

- cloud computing
- high-end servers
- distributed databases

High-Throughput Hashing

- cloud computing
- high-end servers
- distributed databases

SHA-512: random message ranging from 32KB and 128KB

CPU: Intel Core i7-3520M 2.9 GHz, 2 cores

9.37 cycles / byte $\rightarrow$ 295 MB / second / core

High-Throughput Hashing

- cloud computing
- high-end servers
- distributed databases

SHA-512: random message ranging from 32KB and 128KB

CPU: Intel Core i7-3520M 2.9 GHz, 2 cores

9.37 cycles / byte → 295 MB / second / core

NVIDIA GeForce GTX 590, 1.215 GHz (2× GF110)

Batch size	512	1024	2048	4096
Throughput (MB / second / GF110)	670	1100	1650	2100
Speedup	2.27	3.73	5.59	7.12

[2] J. W. Bos, D. Stefan: Performance Analysis of the SHA-3 Candidates on Exotic Multi-core Architectures. In Cryptographic Hardware and Embedded Systems (CHES) 2010
[3] D. A. Osvik, J. W. Bos, D. Stefan, D. Canright: Fast Software AES Encryption. In Fast Software Encryption (FSE) 2010

Public-Key Cryptosystems based on elliptic curves

Elliptic Curves over prime fields – Definition

Let $p > 3$ be a prime, then any $a, b \in \mathbf{F}_p$ such that $4a^3 + 27b^2 \neq 0$ define an elliptic curve $E_{a,b}$ over $\mathbf{F}_p$. The zero point $\mathfrak{o}$, together with the set of points $(x, y) \in \mathbf{F}_p \times \mathbf{F}_p$ which satisfy the short affine Weierstrass equation

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

form an abelian group $E_{a,b}(\mathbf{F}_p)$.

Public-Key Cryptosystems based on elliptic curves

Elliptic Curves over prime fields – Definition

Let $p > 3$ be a prime, then any $a, b \in \mathbf{F}_p$ such that $4a^3 + 27b^2 \neq 0$ define an elliptic curve $E_{a,b}$ over $\mathbf{F}_p$. The zero point $\mathcal{O}$, together with the set of points $(x, y) \in \mathbf{F}_p \times \mathbf{F}_p$ which satisfy the short affine Weierstrass equation

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

form an abelian group $E_{a,b}(\mathbf{F}_p)$.

Standards (NIST)

ECDSA as standardized in FIPS 186-3: Digital Signature Standard (DSS)

128-bit security level corresponds to 256-bit ECC keys / 3072-bit RSA keys

ECC is an order of magnitude faster [NSA] for 128-bit security

Cryptanalysis

The Certicom ECC Challenge

"to increase industry understanding and appreciation for the difficulty of the elliptic curve discrete logarithm problem"

ECC2K-130 challenge is over $E(\mathbf{F}_{2^{131}})$

[4] D. V. Bailey, L. Batina, D. J. Bernstein, P. Birkner, J. W. Bos, H.-C. Chen, C.-M. Cheng, G. van Damme, G. de Meulenaer, L. J. D. Perez, J. Fan, T. Güneysu, F. Gurkaynak, T. Kleinjung, T. Lange, N. Mentens, R. Niederhagen, C. Paar, F. Regazzoni, P. Schwabe, L. Uhsadel, A. Van Herrewege, B.-Y. Yang: Breaking ECC2K-130, Cryptology ePrint Archive, Report 2009/541

Cryptanalysis

The Certicom ECC Challenge

"to increase industry understanding and appreciation for the difficulty of the elliptic curve discrete logarithm problem"

ECC2K-130 challenge is over $E(\mathbf{F}_{2^{131}})$

Cost to solve the ECC2K-130 on different platforms

FPGA (XC3S5000, 111 MHz):	$\approx$	610 year
GTX 295:	$\approx$	1070 year
PlayStation 3:	$\approx$	2650 year
Core-2 Q6850 (4 cores):	$\approx$	3040 year

[4] D. V. Bailey, L. Batina, D. J. Bernstein, P. Birkner, J. W. Bos, H.-C. Chen, C.-M. Cheng, G. van Damme, G. de Meulenaer, L. J. D. Perez, J. Fan, T. Güneysu, F. Gurkaynak, T. Kleinjung, T. Lange, N. Mentens, R. Niederhagen, C. Paar, F. Regazzoni, P. Schwabe, L. Uhsadel, A. Van Herrewege, B.-Y. Yang: Breaking ECC2K-130, Cryptology ePrint Archive, Report 2009/541

Cryptanalysis

The Certicom ECC Challenge

"to increase industry understanding and appreciation for the difficulty of the elliptic curve discrete logarithm problem"

ECC2K-130 challenge is over $E(\mathbf{F}_{2^{131}})$

Cost to solve the ECC2K-130 on different platforms

FPGA (XC3S5000, 111 MHz):	$\approx$	610 year
GTX 295:	$\approx$	1070 year
PlayStation 3:	$\approx$	2650 year
Core-2 Q6850 (4 cores):	$\approx$	3040 year

256-bit keys are roughly 10^{19} times as difficult to break

[4] D. V. Bailey, L. Batina, D. J. Bernstein, P. Birkner, J. W. Bos, H.-C. Chen, C.-M. Cheng, G. van Damme, G. de Meulenaer, L. J. D. Perez, J. Fan, T. Güneysu, F. Gurkaynak, T. Kleinjung, T. Lange, N. Mentens, R. Niederhagen, C. Paar, F. Regazzoni, P. Schwabe, L. Uhsadel, A. Van Herrewege, B.-Y. Yang: Breaking ECC2K-130, Cryptology ePrint Archive, Report 2009/541

Cryptography, NIST-p224, 112-bit security

High-throughput is often not the most important factor
Low-latency is often much more valuable

Cryptography, NIST-p224, 112-bit security

High-throughput is often not the most important factor
Low-latency is often much more valuable

Option 1 Parallel $\mathbf{F}_p$ arithmetic (p prime)

- Try and implement a multi-core version of modular multiplication using a residue number system
- One of the few techniques to speed-up RSA on many-core platforms

Cryptography, NIST-p224, 112-bit security

High-throughput is often not the most important factor
Low-latency is often much more valuable

Option 1 Parallel $\mathbf{F}_p$ arithmetic (p prime)

- Try and implement a multi-core version of modular multiplication using a residue number system
- One of the few techniques to speed-up RSA on many-core platforms

Option 2 Parallel EC-arithmetic

Idea: for ECC we have more freedom

- Compute the $\mathbf{F}_p$ arithmetic per thread for throughput
- Compute the EC-arithmetic in parallel

Use the Montgomery-ladder.

Low-latency for ECC

Cost per bit for scalar multiplication using $E(\mathbf{F}_{p_{224}})$:

Approach	#mul in $\mathbf{F}_{p_{224}}$
State-of-the-art	$\approx 8 - 10$

[5] J. W. Bos: Low-Latency Elliptic Curve Scalar Multiplication, in International Journal of Parallel Programming, 2012

Low-latency for ECC

$$(P+Q,2Q)=(\tilde{P},\tilde{Q})=((\tilde{P}_x,\tilde{P}_z),(\tilde{Q}_x,\tilde{Q}_z))=$$

$$\begin{cases}
\tilde{P}_x = & 2(P_xQ_z+Q_xP_z)(P_xQ_x+aP_zQ_z) \\
 & +4bP_z^2Q_z^2-G_x(P_xQ_z-Q_xP_z)^2 \\
\tilde{P}_z = & (P_xQ_z-Q_xP_z)^2 \\
\tilde{Q}_x = & (Q_x^2-aQ_z^2)^2-8bQ_xQ_z^3 \\
\tilde{Q}_z = & 4(Q_xQ_z(Q_x^2+aQ_z^2)+bQ_z^4)
\end{cases}$$

Cost per bit for scalar multiplication using $E(\mathbf{F}_{p_{224}})$:

Approach	#mul in $\mathbf{F}_{p_{224}}$
State-of-the-art	$\approx 8-10$
Montgomery ladder	18

[5] J. W. Bos: Low-Latency Elliptic Curve Scalar Multiplication, in International Journal of Parallel Programming, 2012

Low-latency for ECC

$$(P+Q,2Q)=(\tilde{P},\tilde{Q})=((\tilde{P}_x,\tilde{P}_z),(\tilde{Q}_x,\tilde{Q}_z))=$$
$$\begin{cases}\tilde{P}_x = & 2(P_xQ_z+Q_xP_z)(P_xQ_x+aP_zQ_z) \\ & +4bP_z^2Q_z^2-G_x(P_xQ_z-Q_xP_z)^2 \\ \tilde{P}_z = & (P_xQ_z-Q_xP_z)^2 \\ \tilde{Q}_x = & (Q_x^2-aQ_z^2)^2-8bQ_xQ_z^3 \\ \tilde{Q}_z = & 4(Q_xQ_z(Q_x^2+aQ_z^2)+bQ_z^4)\end{cases}$$

Cost per bit for scalar multiplication using $E(\mathbf{F}_{p_{224}})$:

Approach	#mul in $\mathbf{F}_{p_{224}}$
State-of-the-art	$\approx 8-10$
Montgomery ladder	18
GPU, using 7 threads	3

- **Advantage**: latency is reduced by a factor 3
- **Disadvantage**: Use 7 threads, per warp 4 threads are idle

[5] J. W. Bos: Low-Latency Elliptic Curve Scalar Multiplication, in International Journal of Parallel Programming, 2012
[6] W. Fischer, C. Giraud, E. W. Knudsen, J. P. Seifert: Parallel scalar multiplication on general elliptic curves over $\mathbf{F}_p$ hedged against non-differential side-channel attacks. Cryptology ePrint Archive, Report 2002/007 (2002).

Results

Ref	Platform		cores / GPU	MHz	Min. L [ms]	Max. T [op/s]
[7]	8800 GTS	(1)	96	1200	305.0	1 413
[8]	8800 GTS	(1)	96	1200	30.3	3 138
	GTX 285	(1)	240	1476	24.3	9 990
New	GTX 295	(2)	240	1242	10.6	79,198
	GTX 480	(1)	480	1401	2.3	237 415
	GTX 580	(1)	512	1544	1.9	290 535
[9]	Intel core-i7 2600K		4	3400	0.09	46 176

[7] R. Szerwinski, T. Güneysu: Exploiting the power of GPUs for asymmetric cryptography. In: Cryptographic Hardware and Embedded Systems (CHES) 2008

[8] S. Antao, J. C. Bajard, L. Sousa: Elliptic curve point multiplication on GPUs. In: Application-specific Systems Architectures and Processors (ASAP) 2010

[9] E. Käsper: Fast elliptic curve cryptography in OpenSSL. In: Real-Life Cryptographic Protocols and Standardization, 2012

Results

GTX 295 (single GT200) vs GTX 285

- Latency reduced by a factor 2.3
- Throughput increased by a factor 7.9

Ref	Platform		cores / GPU	MHz	Min. L [ms]	Max. T [op/s]
[7]	8800 GTS	(1)	96	1200	305.0	1 413
[8]	8800 GTS	(1)	96	1200	30.3	3 138
[8]	GTX 285	(1)	240	1476	24.3	9 990
New	GTX 295	(2)	240	1242	10.6	79,198
New	**GTX 480**	(1)	480	1401	2.3	237 415
New	**GTX 580**	(1)	512	1544	1.9	290 535
[9]	Intel core-i7 2600K		4	3400	0.09	46 176

[7] R. Szerwinski, T. Güneysu: Exploiting the power of GPUs for asymmetric cryptography. In: Cryptographic Hardware and Embedded Systems (CHES) 2008

[8] S. Antao, J. C. Bajard, L. Sousa: Elliptic curve point multiplication on GPUs. In: Application-specific Systems Architectures and Processors (ASAP) 2010

[9] E. Käsper: Fast elliptic curve cryptography in OpenSSL. In: Real-Life Cryptographic Protocols and Standardization, 2012

Results

GTX 580 vs Intel core-i7

- CPU stills wins by a factor 21, 1.9 ms is acceptable in many scenarios
- Throughput increased by a factor 6.3

Ref	Platform		cores / GPU	MHz	Min. L [ms]	Max. T [op/s]
[7]	8800 GTS	(1)	96	1200	305.0	1 413
[8]	8800 GTS	(1)	96	1200	30.3	3 138
	GTX 285	(1)	240	1476	24.3	9 990
New	GTX 295	(2)	240	1242	10.6	79,198
	GTX 480	(1)	480	1401	2.3	237 415
	GTX 580	(1)	512	1544	1.9	290 535
[9]	Intel core-i7 2600K		4	3400	0.09	46 176

[7] R. Szerwinski, T. Güneysu: Exploiting the power of GPUs for asymmetric cryptography. In: Cryptographic Hardware and Embedded Systems (CHES) 2008

[8] S. Antao, J. C. Bajard, L. Sousa: Elliptic curve point multiplication on GPUs. In: Application-specific Systems Architectures and Processors (ASAP) 2010

[9] E. Käsper: Fast elliptic curve cryptography in OpenSSL. In: Real-Life Cryptographic Protocols and Standardization, 2012

Conclusions

- GPUs are useful as a cryptographic accelerator
- High-throughput is easy, low-latency is a challenge
- Faster (parallel) arithmetic $\rightarrow$ faster cryptanalysis: security implications

Future work on GPUs

- Optimize integer factoring using GPUs (implications for RSA)

  J. W. Bos, T. Kleinjung: ECM at Work. in Asiacrypt 2012
- Study the security of elliptic curve based schemes in more detail
- Rethink arithmetic building blocks: faster cryptography
  - Faster parallel algorithms
  - Minimize thread-communication
  - Minimize memory-per-thread