



Enabling the Web with Next Generation Cryptographic Technologies

Sheueling Chang, Hans Eberle

Vipul Gupta, Nils Gura

Sun Microsystems Laboratories



The Next Wave of the Internet

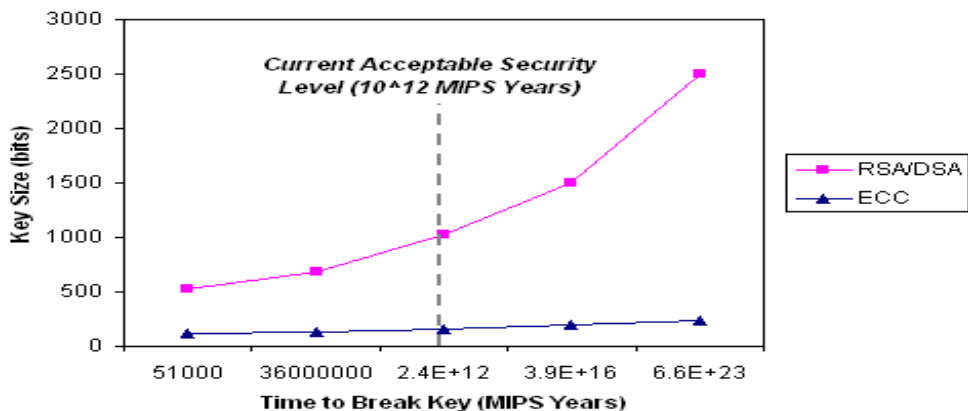


Web Server / https

- E-commerce secured by https
- Desktop deployment today mostly RSA based
- More small wireless devices connected to Internet
- Private key operations are slow on small devices, increasing RSA key sizes is of concern
- Small devices with limited CPU, bandwidth, & power consumption
- Emerging cryptographic technologies ECC/AES

Why Elliptic Curve Cryptography

COMPARISON OF SECURITY LEVELS of ECC and RSA & DSA



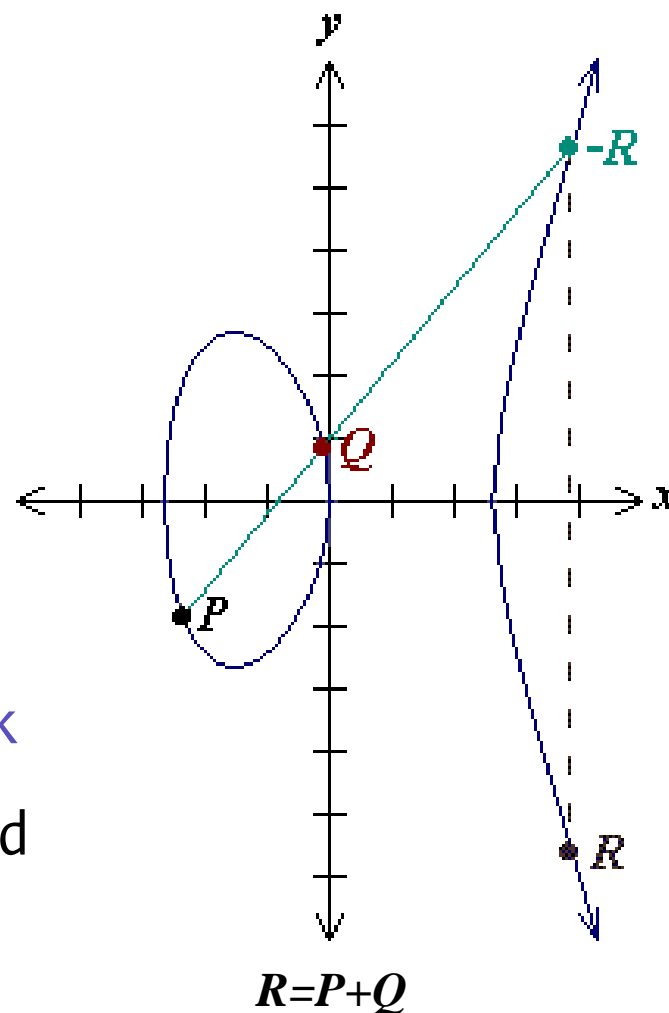
RSA/ECC keysize growth ratio

MIPS yrs	RSA	ECC	Ratio
10^4	512	113	5:1
10^{11}	1,024	160	7:1
10^{20}	2,048	224	9:1
10^{78}	21,000	571	36:1

- New public-key crypto system providing highest security strength per bit
- Government standardized ECC/AES in 2000/2001, plans to switch in 2005-2008 time frame
- ECC suitable for wireless devices
- Mobile applications will drive market

Elliptic Curve Cryptography

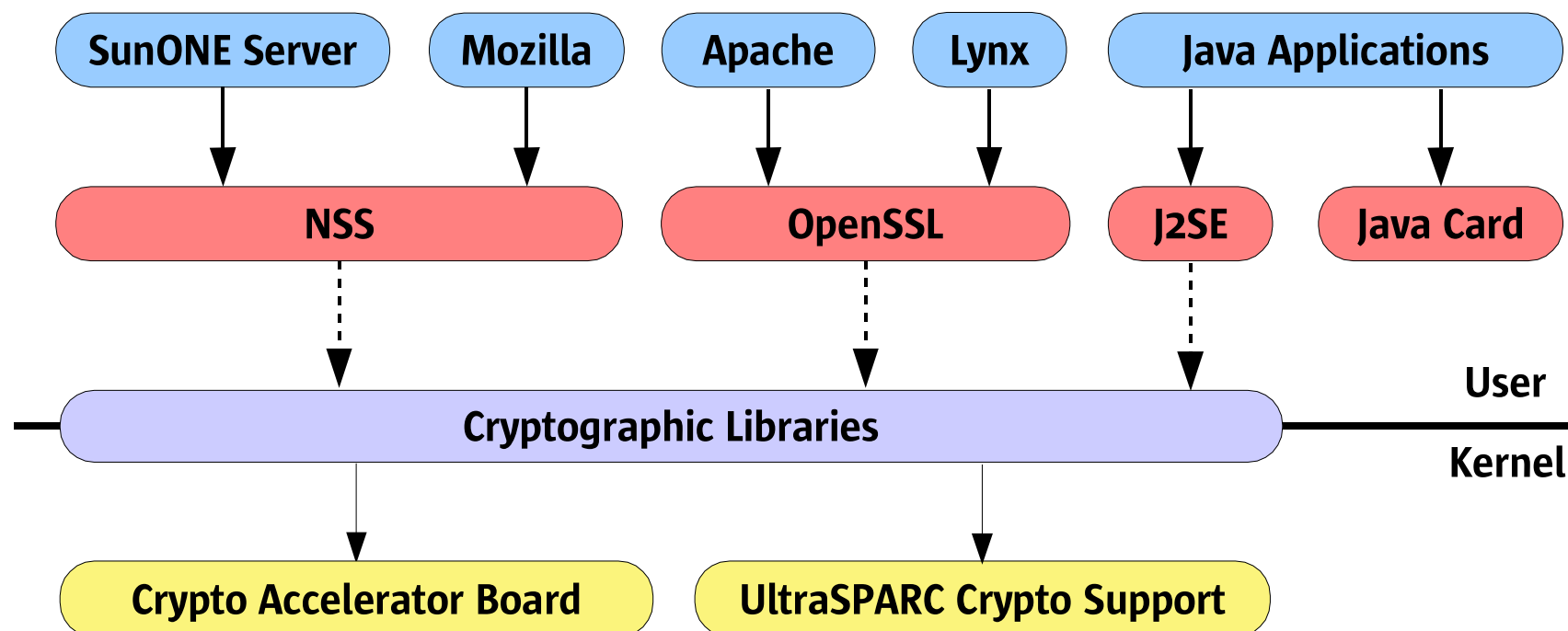
- Point multiplication $Q=k \cdot P$
- Repeated point addition and doubling:
 $9P = 2(2(2P)) + P$
- Public key operation: $Q(x,y) = k \cdot P(x,y)$
 $Q =$ public key
 $P =$ base point (curve parameter)
 $k =$ private key
- Elliptic curve discrete logarithm
 Given public key kP , find private key k
- Best known attack: Pollard's rho method
 with running time: $(\pi n)^{1/2}/2$



Promoting Industry Adoption

- Market adoption for ECC
 - IETF standardization of ECC in SSL
 - Articulate ECC benefits in real-world scenarios
- Open Source Contribution
 - Two key open source security infrastructures (OpenSSL, Mozilla/NSS)
 - Integrate ECC into key open source security libraries
- Acceleration techniques in Sun Platforms
 - Hardware crypto acceleration
 - Optimized ECC point multiplication and RSA modular multiplication
- Elliptic Curve support in J2SE, Java Card

Web Security Architecture

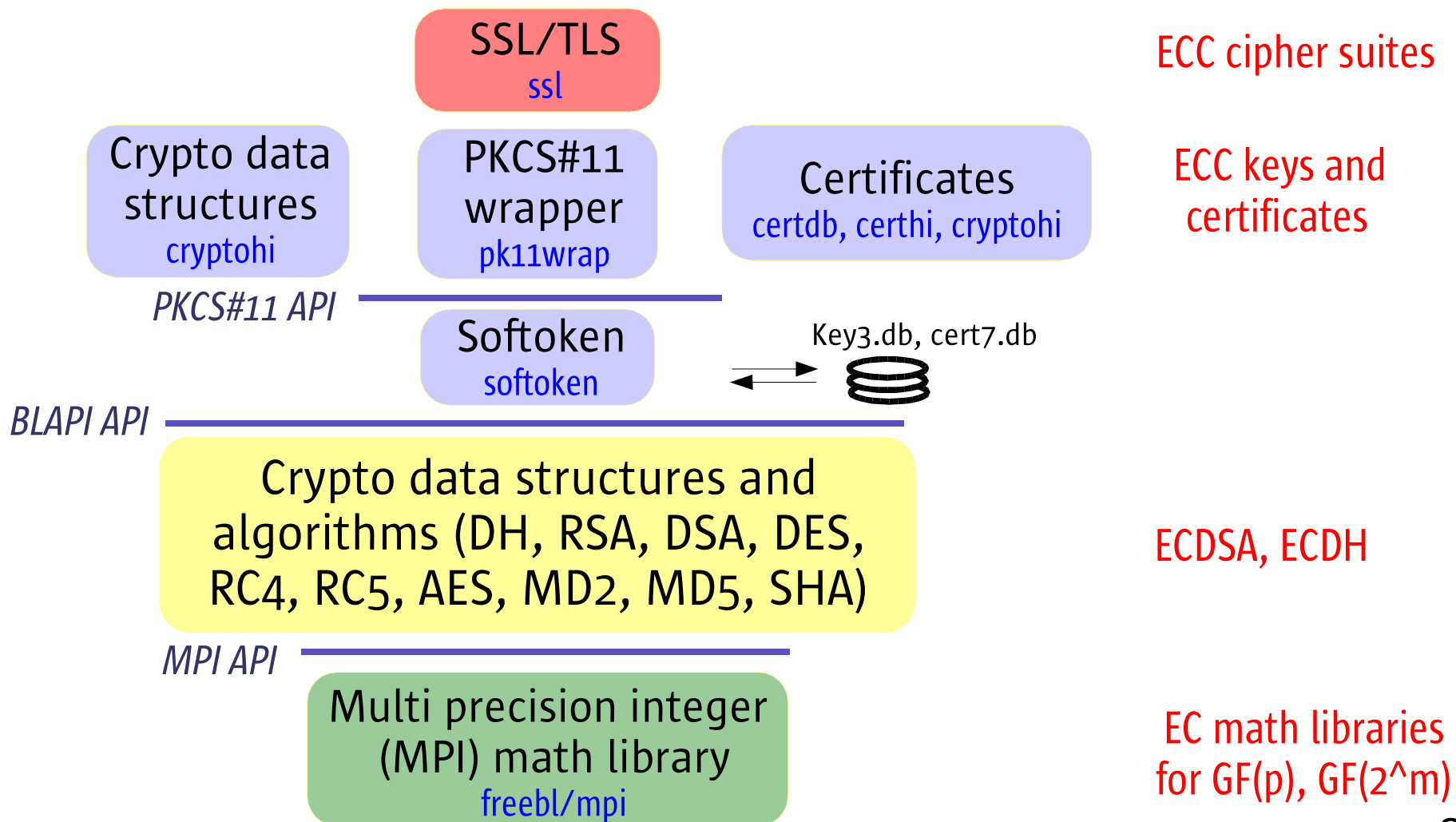


- Provide end-to-end secure communication
- Integrate ECC technology into server and client impl.
- Open standards (IETF TLS draft, Java Card, J2SE 1.5) and open source (OpenSSL, NSS, Apache, Mozilla)

Open Source Contribution

- Sun contributed Elliptic Curve Technologies to **OpenSSL** and **Mozilla/NSS**
- Source code under open source license, allows free use for commercial and non-commercial purposes
- Developers can incorporate ECC technology into innovative security-enabled products and services
- End-to-end client server secure communication
Apache server, **SunOne** server, **Mozilla**/Dillo browsers
- Source code:
<ftp://ftp.openssl.org/snapshot> (openssl-SNAP-20020911.tar.gz or later)

Open Source Contribution (Mozilla/NSS)



Java Platform – J2SE & Java Card

- **J2SE 1.5** release planned in 2004 with enhanced Elliptic Curve Support in J2SE/JCA
- New classes are added to `java.security.spec` and `java.security.interfaces` packages for ECC support
- **Java Card 2.2.1** release – ECC technology
- Support for prime integer and binary polynomial curves
- Support **ECDH** (Diffie-Hellman) key exchange and **ECDSA** for signature authentication

Sun Network Conference Badge

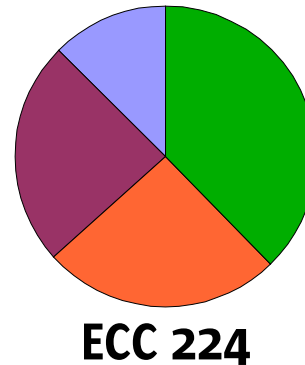
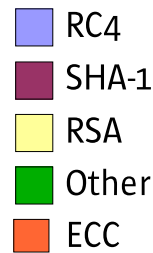
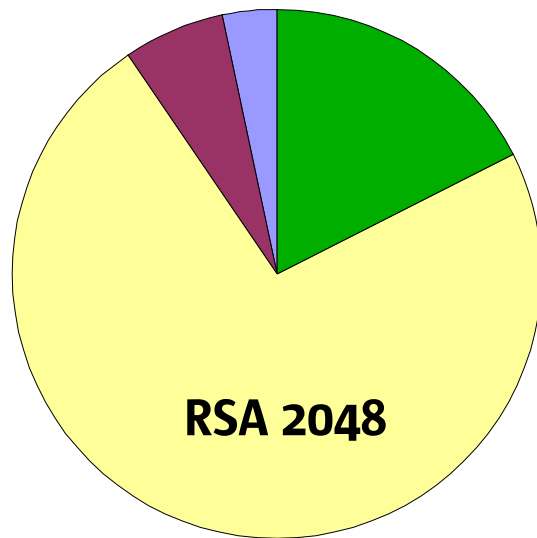
- **Java Card** based technology - digital identification to authenticate attendees in a secure manner
- Provides secure access to networked services, activities, and customized information during the show
- Conference planners - dynamically modify and add applications and services
- Deployed in more than 300 million smart cards worldwide – platform of choice for secure, mobile, and identity-based applications

IETF Standardization

- Ensure interoperability:
end-to-end client-server secure communication
- IETF draft specifying ECC cipher suites for TLS/SSL
<http://www.ietf.org/internet-drafts/draft-ietf-tls-ecc-03.txt>
- Key technology critical to wireless mobile industry
- Open source implementations - Reaffirms Sun's
commitment to advancing open source software and
Internet security infrastructure
- ECC ciphersuites: **ECDH-ECDSA, ECDH-RSA,
ECDHE-ECDSA, ECDHE-RSA**

Key exchange, Bulk encryption and MAC algorithm
ECDH_ECDSA_WITH_RC4_128_MD5

Understanding SSL Performance



Core operations

4X - current key size

14X - future key size

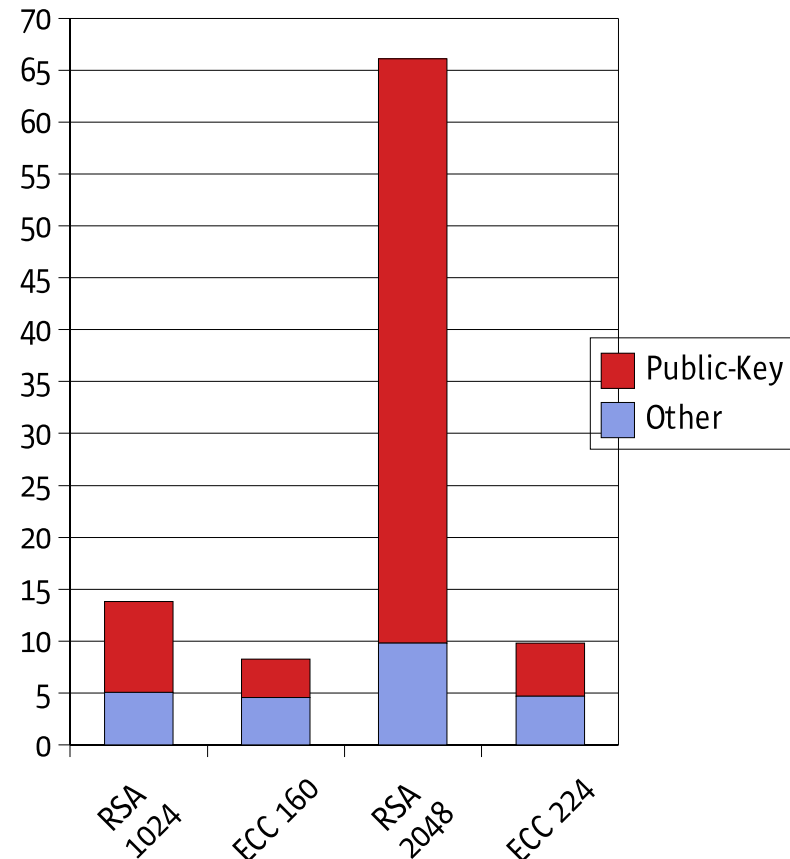
- SSL specifies combinations of algorithms
RC4-MD5, ECDH-ECDSA-AES128-SHA
- Public key, bulk encryption, and hash algorithms -
key size, file size, server software, platform
- Understand how each component affects overall
performance; analyze with Amdahl's law

How ECC Affects SSL Performance

- **SSL benchmark:**
number of connections per second
- Current key sizes: **1.7X**
ECC 160 vs. RSA 1024
- Future key sizes: **6.7X**
ECC 224 vs. RSA 2048

Apache 2 w/OpenSSL, 1 cpu, 10K file sizes, 900MHz UltraSPARC III

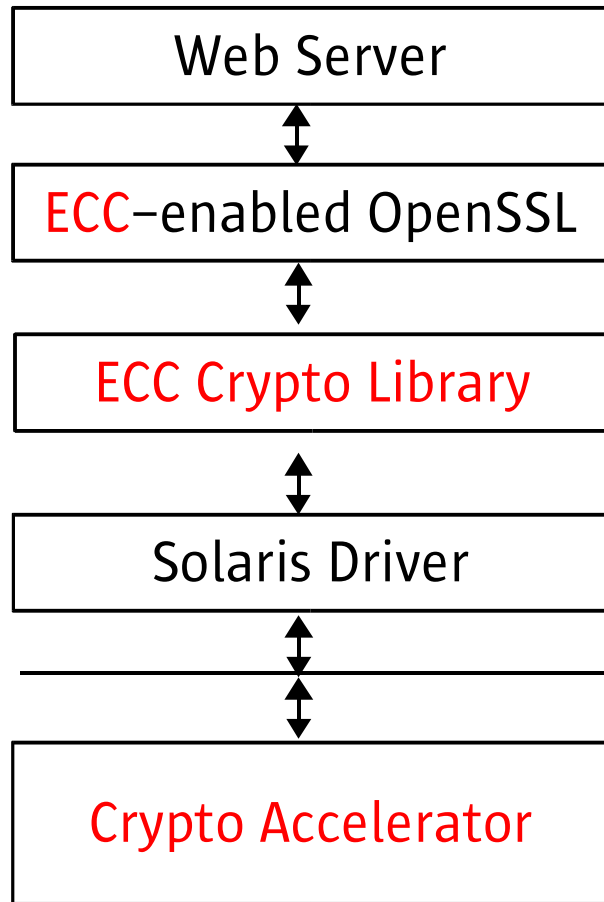
SSL Connection Time (ms)



Public Key Crypto Math

- RSA, DH, DSA are based on **modular exponentiation**
 $C = M^e \bmod n$
- ECC is based on **point multiplication**
 $Q = kP$
- **Modular multiplication** on large operands is the key underlying function
- ECC math is more complicated; also requires multi-word addition / subtraction / division

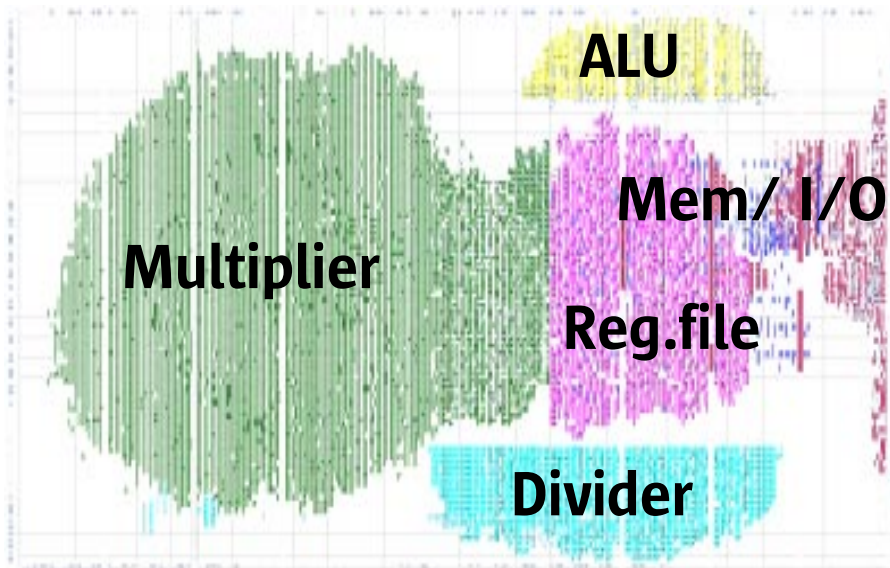
Web Server Acceleration



- Aggregation of secure connections on the server side
- Heterogeneous clients
- Fast impl. of named curves
 - Standardized curves (NIST, SECG)
 - Known curve parameters
- Support for generic curves
 - Infrequently used curves
 - Unknown curve parameters

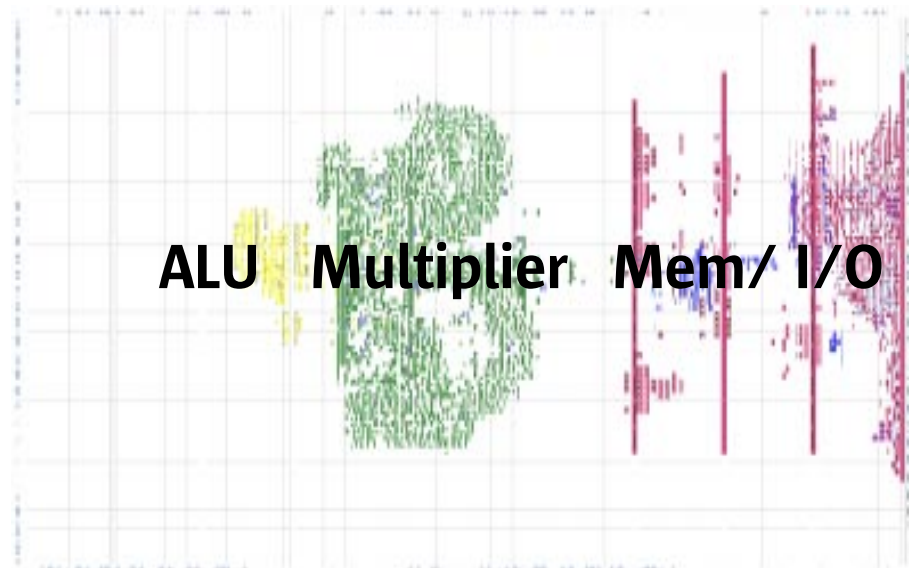
Crypto Accelerator Prototypes

1st Generation ECC Accelerator



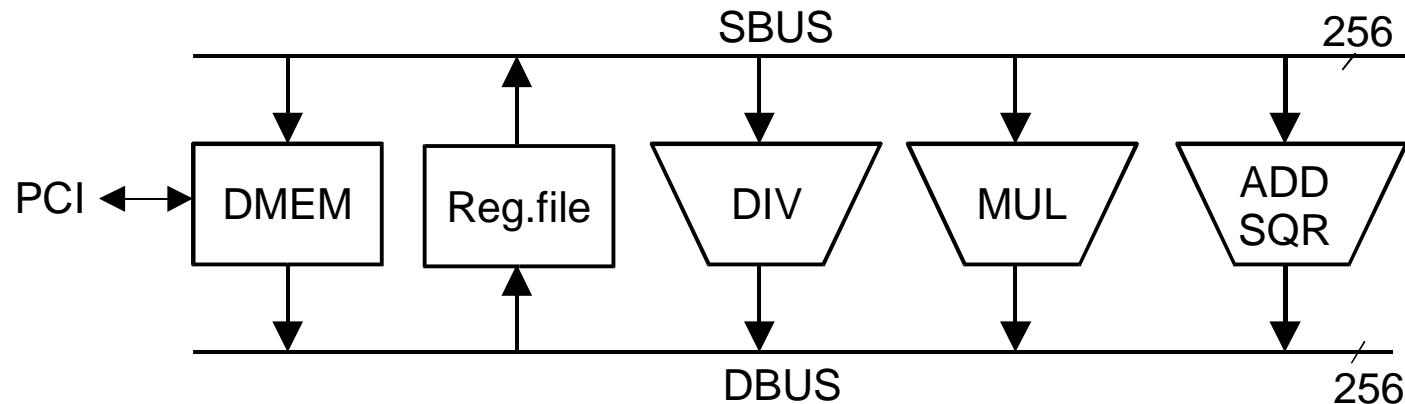
Dedicated 256-bit coprocessor
20,197 LUTs, 6,624 FFs
8kB data memory, 1kB instr. memory

2nd Generation ECC & RSA Accelerator



General-purpose 64-bit processor
4,155 LUTs, 2,387 FFs
16kB data memory, 12.5kB instr. memory

1st Generation ECC Accelerator



- **Dedicated crypto coprocessor for ECC**
 - Binary polynomial fields, key sizes ≤ 255 bits
 - Optimized performance for named curves
 - Support for generic curves
- Architecture
 - Microprogrammable 256-bit data path
 - Load/store architecture
 - ADD, SQR, MUL, DIV

1st Generation Highlights

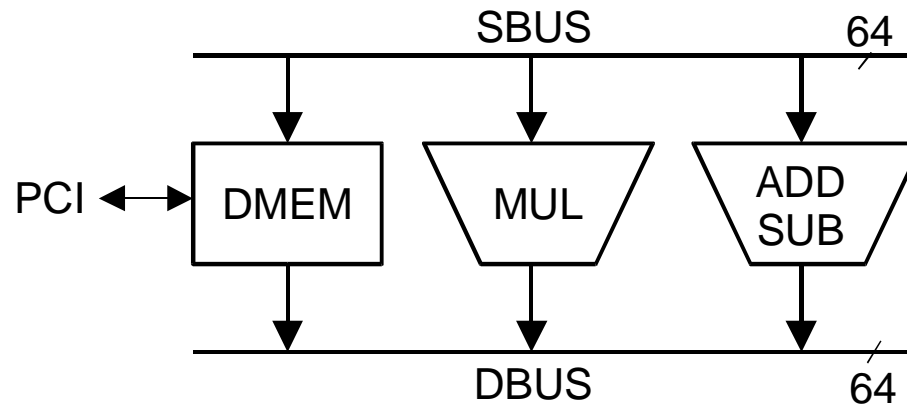
- **Fastest** reported ECC implementation
 - Fast modular multiplier (4 cycles for 256-bit mul)
 - 1-cycle squarer
 - Hardware divider
 - Parallel and overlapped instruction execution
- First ECC implementation supporting **multiple named curves** and **arbitrary generic curves**

Performance

	Hardware	Software*	Speedup
	66 MHz [op/s]	900 MHz USIII [op/s]	
Named Curves			
GF(2 ¹⁶³)	6987	322	21.7x
GF(2 ¹⁹³)	5333	294	18.1x
GF(2 ²³³)	4438	223	19.9x
Generic Curves			
GF(2 ¹⁶³)	3308		
GF(2 ¹⁹³)	2375		
GF(2 ²³³)	1980		

* 64-bit OpenSSL 0.9.8

2nd Generation ECC & RSA Accelerator



- General-purpose proc. supporting ECC, RSA, DH, DSA
 - Binary polynomial fields and prime fields
 - Arbitrary key sizes, arbitrary curves
- Architecture
 - Microprogrammable 64-bit data path
 - Memory operands
 - Multi-word MUL, ADD, SUB
 - Dual-issue machine

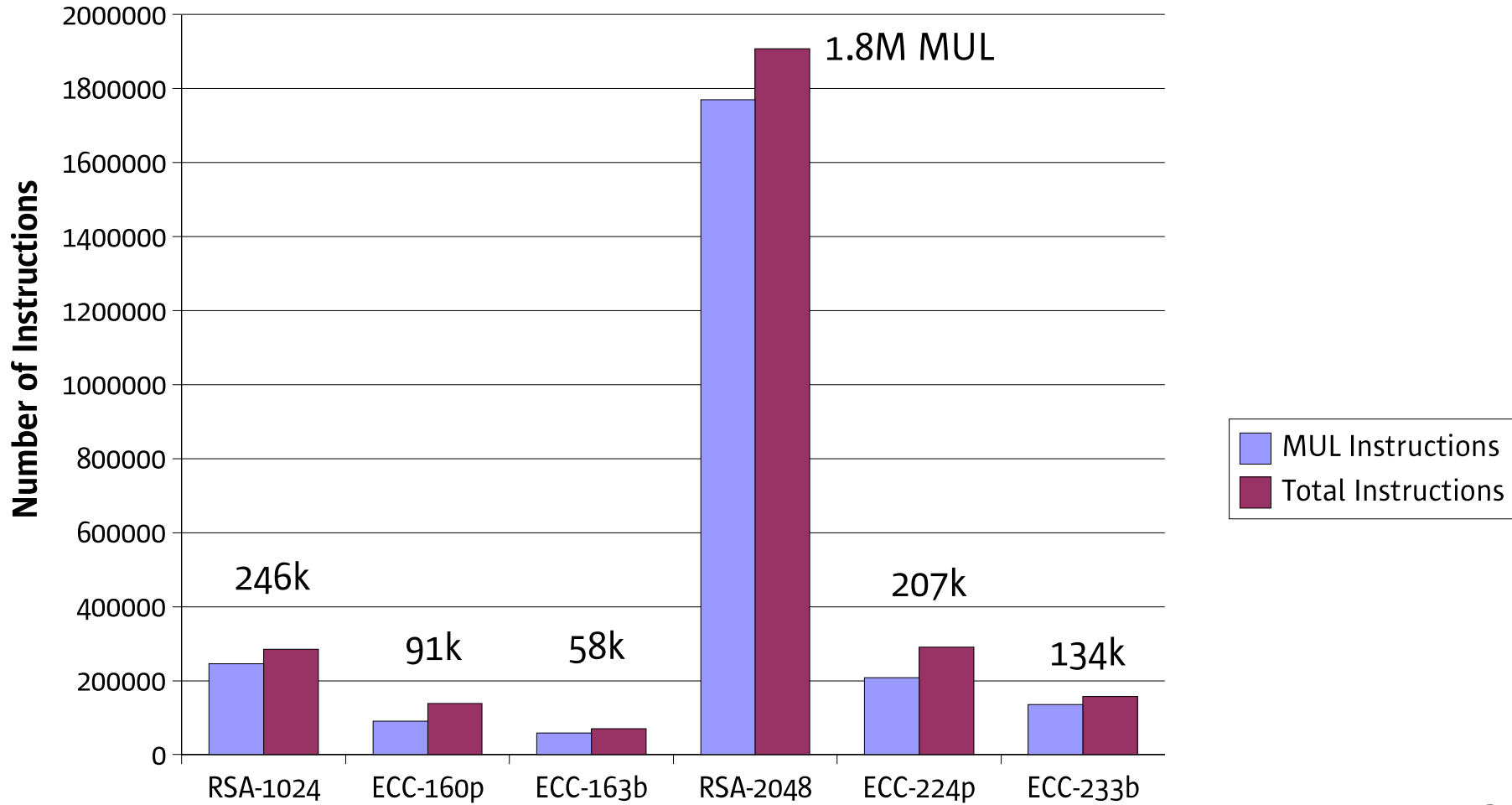
2nd Generation Highlights

- Optimized for **fast modular multiplication**
- Multi-word arithmetic for **arbitrary operand sizes**
- **Dual-field** multiplier
- Projected performance
(1.5 GHz, 2-cycle 64x64 mul)
 - 10,000 ECC-163 op/s
 - 2,500 RSA-1024 op/s

Complexity Analysis

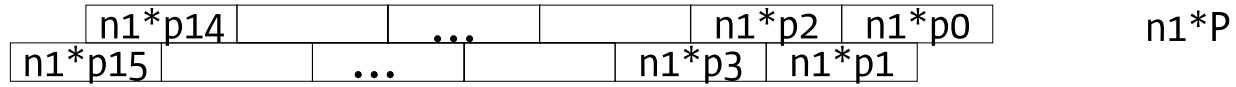
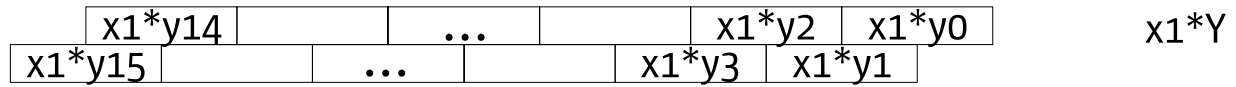
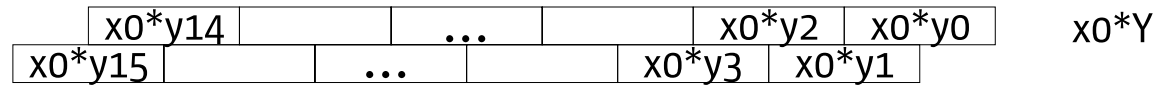
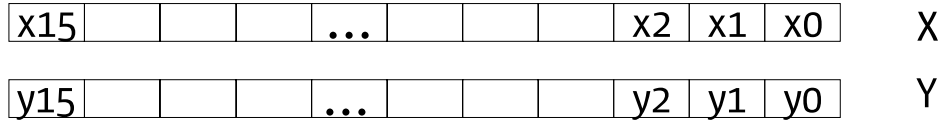
Today 4x

Tomorrow 13x

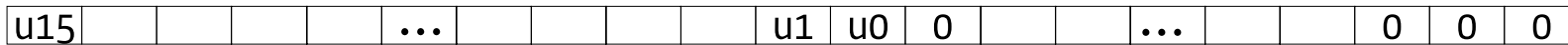
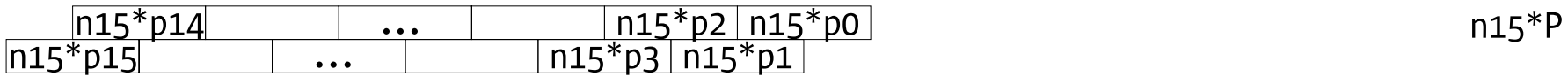
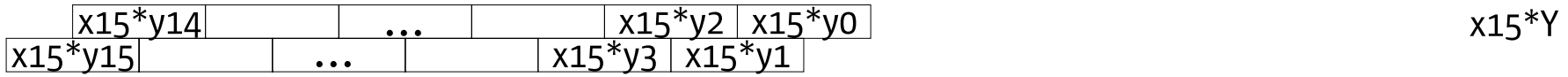


Montgomery Multiplication

$$u = X * Y * 2^{-k} \text{ mod } P$$



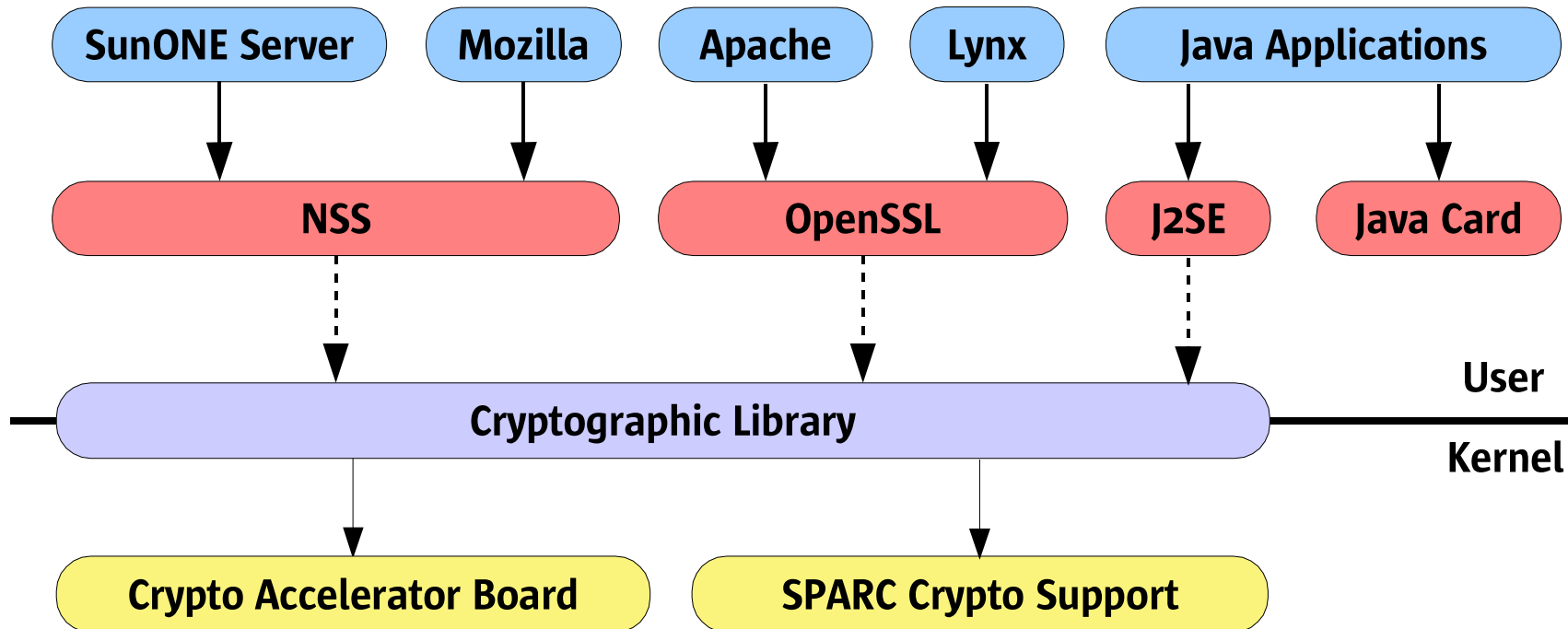
...



Sun's Competitive Advantages

- End-to-end solution for client and server systems based on industry standards (OpenSSL, NSS, TLS)
- Common architecture for RSA and ECC
 - Dual-field multiplier
 - Optimized multi-word multiplication
- Support for arbitrary ECC curves
 - Prime integer and binary polynomial fields
 - Partial reduction for non-standard curves

Towards a Coherent Crypto Architecture



Security is no longer an option, it is becoming an **integral part** of a system architecture.



Contact information:

Sheueling Chang (sheueling.chang@sun.com)

Hans Eberle (hans.eberle@sun.com)

