DEPLOYMENTS OF ELLIPTIC CURVE CRYPTOGRAPHY

Scott Vanstone
University of Waterloo

September 19, 2005

Outline

- Introduction to ECC
- US government's "Suite B"
- Digital postal marks
- Fast ECDSA verification
- Web security (SSL/TLS)
- BlackBerry security
- Other deployments

ECC Parameters

- Let E be an elliptic curve defined over a finite field $\mathbb{F}_q$.
- Suppose $\#E(\mathbb{F}_q) = nh$, where n is prime and h is small. (By Hasse's Theorem, we have $n \approx q$.)
- Let $P \in E(\mathbb{F}_q)$ be a base point of order n.
- Key generation: Each user selects a random integer $d \in [0, n-1]$. The user's public key is $Q = dP$, and its private key is d.
- A necessary condition for the security of any ECC protocol is that the ECDLP be intractable:
  - Given E, n, P and Q, find d.

ECC/RSA Key Size Comparisons

(FIPS 186-2, Lenstra/Verheul, NESSIE)

Security level in bits	Block cipher	$\mathbb{F}_p$ $\|p\|$	$\mathbb{F}_{2^m}$ m	RSA $\|n\|$
80	SKIPJACK	192	163	1024
112	Triple-DES	224	233	2048
128	AES Small	256	283	3072
192	AES Medium	384	409	7680
256	AES Large	521	571	15360

Key Lifetimes

- FIPS 201: Personal Identity Verification (PIV) of Federal Employees and Contractors.
- SP 800-78: Cryptographic Algorithms and Key Sizes for PIV.
- Elliptic curves: P-224, K-233, B-233, P-256, K-283, B-283.

PIV authentication key	– 2010	RSA: 1024,2048,3072, ECDSA: 224-283
	2011–	RSA: 2048,3072, ECDSA: 224-283
Card authentication key	– 2010	RSA: 1024,2048, 3072, ECDSA 224-283
	2011 –	RSA: 2048,3072, ECDSA 224-283
Digital signature key	– 2008	RSA: 1024,2048, 3072, ECDSA 224-283
	2009 –	RSA: 2048,3072, ECDSA 224-283
Key management key	– 2008	RSA: 2048,3072, ECDSA 224-283
	2009 –	RSA: 2048,3072, ECDSA 224-283

Curve Selection

Popular choices are the 15 elliptic curves recommended by NIST in FIPS 186-2:

- One randomly-selected curve (with $h = 2$) over each of the binary fields $\mathbb{F}_{2^{163}}$, $\mathbb{F}_{2^{233}}$, $\mathbb{F}_{2^{283}}$, $\mathbb{F}_{2^{409}}$, $\mathbb{F}_{2^{571}}$.
- One Koblitz curve (with $h = 2$ or 4) over each of the binary fields $\mathbb{F}_{2^{163}}$, $\mathbb{F}_{2^{233}}$, $\mathbb{F}_{2^{283}}$, $\mathbb{F}_{2^{409}}$, $\mathbb{F}_{2^{571}}$.
- One randomly-selected curve (with $h = 1$) over each of the prime fields $\mathbb{F}_p$ for the following p;
  - $p = 2^{192} - 2^{64} - 1$.
  - $p = 2^{224} - 2^{96} + 1$.
  - $p = 2^{256} - 2^{224} + 2^{192} + 2^{96} - 1$.
  - $p = 2^{384} - 2^{128} - 2^{96} + 2^{32} - 1$.
  - $p = 2^{521} - 1$.

Elliptic Curve Digital Signature Algorithm

- The Elliptic Curve Digital Signature Algorithm (ECDSA) is an elliptic curve analogue of the DSA.
- Widely standardized:
  - 1999: ANSI X9.62.
  - 2000: NIST FIPS 186-2.
  - 2000: IEEE 1363-2000.
  - 2002: ISO 15946-2.
- Domain parameters: E, $\mathbb{F}_q$, n, h, P.
- Each user A has a private key $d \in_R [1, n-1]$, and a public key $Q = dP$.

ECDSA Signature Generation

To sign a message m, A does the following:

1. Select a random integer k, $1 \leq k \leq n-1$.
2. Compute $R = kP$ and $r = x(R) \bmod n$.
   If $r = 0$ then go to step 1.
3. Compute $k^{-1} \bmod n$.
4. Compute $e = H(m)$, where H is a hash function.
5. Compute $s = k^{-1}(e + dr) \bmod n$.
   If $s = 0$ then go to step 1.
6. A's signature for the message m is (r, s).

ECDSA Signature Verification

To verify A's signature (r, s) on m, B should do the following:

1. Verify that r and s are integers in the interval $[1, n-1]$.
2. Compute $e = H(m)$.
3. Compute $u_1 = es^{-1} \bmod n$ and $u_2 = rs^{-1} \bmod n$.
4. Compute $R = u_1 P + u_2 Q$ and $v = x(R) \bmod n$.
5. Accept the signature if and only if $v = r$.

ECDSA versus RSA

- A primary comparison point is performance:
  - Signature generation: ECDSA is faster than RSA ($s = H(m)^d \bmod n$), and especially so if k, kP, k^{-1} are precomputed.
  - Signature verification: RSA ($s^e \equiv H(m) \pmod n$) is generally faster than ECDSA, and especially so if $e = 3$.
- Other issues include: signature and key size, code size, memory requirements, power consumption, requirement for an arithmetic processor, suitability for hardware implementation, standardization.

RSA versus ECDSA Verification

- Certicom Security Builder
- Device: Ipaq 3950, Intel PXA250 Processor, 400 MHz.
- RSA: $e = 2^{16} + 1$, decryption with CRT.
- ECDSA: NIST elliptic curves over prime fields (without fast verify technique).

RSA key size	verify (ms)	ECDSA key size	verify (ms)
1024	1.4	160	4.0
2048	5.2	224	7.7
3072	11.0	256	11.8
7680	65.8	384	32.9
15360	285.0	521	73.2

NSA Suite B

- To be used in equipment that will be fielded under crypto modernization initiatives.
  - About 1.3 million units of equipment will be replaced over the next 10 years.
  - Equipment will be used for the next 20-50 years.
- Two different levels of security:
  - Unclassified but mission critical data.
  - Classified and sensitive command and control information.
- The only public-key scheme is ECC.
- Also used in Canada, UK, and other NATO countries.

www.nsa.gov/ia/industry/crypto_elliptic_curve.cfm?MenuID=10.2.7

www.nsa.gov/ia/industry/crypto_suite_b.cfm?MenuID=10.2.7

Algorithms

Purpose	Algorithm	Unclassified	Classified
Encryption	AES	128-bit	256-bit
Hashing	SHA	SHA-256	SHA-384
Signatures	ECDSA	P-256	P-384
Key Agreement	ECDH or ECMQV	P-256	P-384

Standards

- AES: FIPS 197.
- SHA: FIPS 180-2
- ECDSA: FIPS 186-2 (ANSI X9.62)
- ECDH/ECMQV: SP 800-56 (ANSI X9.63)

Digital Postal Marks (DPMs)

- Purpose: provide evidence of payment to the post office.
- In traditional postage meters, the printing mechanism is securely coupled to an accounting unit.
- In the new generation of postage meters, there is no such coupling – the DPM may be generated by an individual user's desktop printer.
- The DPM should allow the post office to detect fraud (including duplication), and produce evidence of fraud.
- DPM should be small (< 128 bytes) and efficiently verified.
- DPMs are used in many countries, including Canada.

DPMs Using Digital Signatures

- The DPM should contain the mailer's digital signature on some postal data (PD) that includes:
  - An identification number for the mailer.
  - A serial mail piece count.
  - The postal value.
- Other desirable data includes:
  - The date and time the DPM was generated.
  - Delivery address information.
  - Sender postal code.
- Some importants issues:
  - Size of the public key and signature.
  - Retrieval of the public key (for signature verification).
  - Strategy for detecting duplicates.

Canada Post's DPM

- Mandates the use of ECDSA as the signing mechanism.
- Effective Dec 31 2006, all postal meters must be in compliance with the new standard.
- US Postal Service's Information-Based Indicia Program (IBIP) permits RSA, DSA or ECDSA.

Fast ECDSA Verification

(Joint work with A. Antipa, D. Brown, R. Lambert & R. Struik)

- E : Elliptic curve defined over a prime field $\mathbb{F}_p$.
- $\#E(\mathbb{F}_p) = n$, where $n > p$ is prime.
- Recall: $R = kP$, $r = x(R) \bmod n$.
- Since $n > p$, we have $r = x(R)$.
- Suppose that a single bit is appended to the signature so that $y(R)$ can be efficiently determined.
  (Note: One can always compute $R = u_1P + u_2Q$.)
- Then verification is equivalent to:

$$R \stackrel{?}{=} u_1P + u_2Q,$$

  where $u_1 = es^{-1} \bmod n$ and $u_2 = rs^{-1} \bmod n$.

Fast ECDSA Verification (2)

- $u_1P + u_2Q$ can be computed by first finding the joint sparse form (JSF) for (u_1, u_2), and then using 'Shamir's trick.'

1	0	0	-1	0	-1	-1	P
1	1	0	1	0	-1	0	Q

- If t is the bitlength of n, then the expected work factor is t point doublings, and $t/2$ point additions.

Fast ECDSA Verification (3)

- Using the extended Euclidean algorithm, we can write $u_2 = v_1/v_2$, where $v_1, v_2 \approx \sqrt{n}$.
- The verification equation $R = u_1 P + u_2 Q$ is equivalent to:

$$v_1 Q - v_2 R + u_1 v_2 P = \infty.$$

- This can be written as

$$v_1 Q - v_2 R + v_3 P + v_4 P' = \infty, \qquad (*)$$

  where $P' = 2^{\lceil t/2 \rceil} P$ is precomputed and $v_3, v_4 \approx \sqrt{n}$.
- The left side of (*) can be computed by determining the JSFs for (v_1, v_2) and (v_3, v_4) and then using Shamir's trick.
- The expected work factor is $t/2$ doublings and $t/2$ additions.

Fast Verification (4)

Analysis:

- If Jacobian coordinates are used, then a point addition takes 8M+3S and a point doubling takes 4M+4S.
- We assume that $S \approx 0.8M$.
- Then traditional verification takes $12.4t$ multiplications, and fast verification takes $8.8t$, which is 40% faster.
- Experiments also yielded this speedup.

Shorter Signatures

Pintsov-Vanstone signatures:

- An adaptation of the Nyberg-Rueppel signature scheme that provides partial message recovery (and thus smaller DPMs).
- Standardization: ANSI X9.92 (draft), postal standards UPU S36-4, CEN EN 14615.
- Formal security analysis (Brown & Johnson).

PV Signature Generation

Entity A with key pair (Q, d) signs a message m as follows:

1. Divide m into two parts, R and N, where R is recoverable from the signature, and N is not recoverable. R should contain 'sufficient' redundancy.
2. Select random $u \in [1, n-1]$.
3. Compute $U = uP$ and $k = \mathsf{KDF}(U)$.
4. Compute $c = \mathsf{Enc}_k(R)$, $h = H(c, N)$, $s = dh + u \bmod n$.

The signed message is (N, c, s).

PV Signature Verification

To verify (N, c, s) and recover m, B does:

1. Compute $h = H(c, N)$.
2. Compute $U = sP - hQ$, $k = \mathsf{KDF}(U)$, $R = \mathsf{Dec}_k(c)$.
3. Verify that R has the necessary redundancy.
4. Accept the signature for the message $m = (R, N)$.

DPM Size

Assumptions. 160-bit elliptic curve; PD: 20 bytes.

- Size of PD + ECDSA signature: 60 bytes.
- Size of PD + PV signature: 40-50 bytes (depending on amount of natural redundancy in R).
- Ideally, the DPM should contain all information required for verification, in particular a certificate consisting of the public key and the CA's signature on it.
  - For ECDSA-signed certificates, the certificate size is 60 bytes.
  - An implicit certificate allows a verifier to reconstruct A's public key using the certificate, A's identity, and the CA's public key. The authenticity of a reconstructed public key is only established after it is successfully used to verify a signature.
  - Optimal mail certificate is an elliptic curve point (20 bytes in size).

Web Security (SSL/TLS)

- Secure Sockets Layer (SSL) enables the secure hyper-text transfer protocol (HTTPS).
- Transport Layer Security (TLS) is the IETF version of SSL.
- Main components of SSL/TLS are:
  - Handshake protocol: Allows server and client to authenticate each other and to negotiate cryptographic keys.
  - Record protocol: Used to encrypt and authenticate transmitted data.

  The expensive crypto operations occur in the handshake.
- Sun has integrated ECC support into OpenSSL and the Netscape Security Services (NSS). Apache web server uses OpenSSL; NSS used with Mozilla and Netscape browsers.

Cryptographic Operations in Handshake

Server authentication only

	RSA	ECC
Client	RSA verify + RSA encrypt	ECDSA verify + ECDH
Server	RSA decrypt	ECDH

With client authentication

	RSA	ECC
Client	verify +encrypt +sign	ECDSA verify + ECDSA sign + ECDH, or ECDSA verify + ECDH
Server	2 verify + decrypt	2 ECDSA verify + ECDH, or ECDSA verify + ECDH

ECC alternatives correspond to ECDSA or ECDH certificate.

Throughput and Latency

Source: Sun Labs (http://research.sun.com/projects/crypto/)

Throughput: rate at which server can perform crypto ops in handshake.

Latency: total time on crypto ops on the client and server.

Environments: 450 MHZ Sun Ultra-80 (UltraSPARC II) and a Linux PDA with 200 MHz StrongARM.

No-Client-Authentication

- Server throughput for ECC-160 was more the five times better than with RSA-1024.
- Latency comparison more complicated if server is not loaded.
  - If client and server are on same type platform, then ECC is twice as fast.
  - RSA wins (by factor 4/3) in PDA-client-to-Sun-server. (Expensive private key ops in RSA done only by server.)

Throughput and Latency (2/4)

ECC for SSL/TLS a significant advantage for web servers.

- SPECweb99: 85% of fetches are under 10KB, and the time for RSA operations is 63%–88% of overall time.
- For RSA decryption and ECDH (the main computational costs in SSL handshake), ECC outperformed RSA by factor of 2.4 (RSA-1024/ECC-160) to 11 (RSA-2048/ECC-224).
- Two models: shopping cart with 66% session re-use and financial institution with 87.5% re-use. 30KB files.
  - 66% re-use: ECC-160 allows 31% more requests compared to RSA-1024 and 279% for RSA-2048/ECC-224.
  - 87% re-use: ECC-160 allows 13% more requests than RSA-1024 (and 120% more for RSA-2048/ECC-224).

Throughput and Latency (3/4)

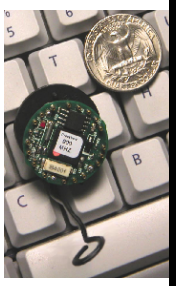
- Performance advantage of ECC increases with smaller pages.
  - If file size is 1 MB, then other factors are more important than the public-key ops.
- If latency is the measurement of interest, then low server loads mean that ECC-160 and RSA-1024 seem comparable to the client.
  - But server is saturated much earlier with RSA, at which point clients experience increased latency.

Throughput and Latency (4/4)

Client-authentication case

- ECC wins in all the comparisons.
- Server throughput comparison is not as dramatic when client authentication is added, but still more than factor 1.5 difference if client has ECDH certificate.
- Case for ECC is compelling (for both latency and throughput) as key size increases.

Sizzle (Slim SSL)



Mica2dot mote

Sun implemented "world's smallest secure web server" on Mica2 motes where "traditional alternatives like RSA are impractical".

- Berkeley/Crossbow motes: battery-powered wireless devices with limited processing power and memory.
- 8-bit Atmel ATmega 128L, 128KB flash, 4KB of EEPROM, 4KB of RAM, up to 16 MHz.
- Useful in industrial monitoring, tracking, security, and military applications (sensor networks).
- Extends internet to remote devices via wireless gateway.

Source: Sun Labs (http://research.sun.com/projects/crypto/)

Average ECC and RSA Execution Times

Atmel ATmega 128L at 8MHz

Algorithm	time s	data bytes	code bytes
ECC P-160	0.81	282	3682
ECC P-192	1.24	336	3979
ECC P-224	2.19	422	4812
Mod. exp. 512	5.37	328	1071
RSA-1024 pub $e{=}2^{16}{+}1$	0.43	542	1073
RSA-1024 priv w. CRT	10.99	930	6292
RSA-2048 pub $e{=}2^{16}{+}1$	1.94	1332	2854
RSA-2048 priv w. CRT	83.26	1853	7736

Handshake Performance

- Authentication (server to client) and secrecy essential. Desirable to use an end-to-end approach that does not require trusted gateway.
- Uses 60KB of program memory and 3/4 of the RAM.
  - Complete web server is below 31KB on Intel x86.
- Employs handshake optimizations to reduce traffic.
  - Handshake exchanges less than 600 bytes; "RSA certificate alone is typically more than 600 bytes." No client authentication.
- Possible to use gateway to accelerate.
  - Eg: if gateway possesses device certificates, then sensor can identify desired certificate by fingerprint and gateway forwards certificate.
- Single curve: ECC P-160. Full handshake on Mica2 (7.4 MHz) in 4 seconds, and 5.6 seconds on Mica2dot (4 MHz).

Energy Consumption

Experimental data with Atmega128L at 4 MHz.

- Energy cost of signing increases by a factor more than 7 when moving from RSA-1024 to RSA-2048, while ECDSA-224 is less than 3 times cost of ECDSA-160.
- Cost equivalents:

Sign operation	Bytes transmitted
RSA-1024	5132
ECDSA-160	385

- For given amount of energy, can perform 4 times as many key exchange operations with ECDSA-160 compared with RSA-1024.

Energy Consumption (2/2)

- Cost of public-key operations EC-160 and RSA-1024 dominates authentication.
- Relative cost depends on bytes transmitted between handshakes and on duty cycle (receive to sleep time).
  - With a duty cycle of .1%, five handshakes with ECC-160 use less than 11% of energy consumed per day; the corresponding consumption for RSA-1024 is approx 30%.
- Execution times and memory requirements favor ECC:

Operation	Time (s)	Data mem (bytes)
ECC-160 point mult	1.6	282
RSA-1024 private key op	22	930

RIM and BlackBerry

Features

- Wireless access to email.
- Phone and SMS text messaging.
- Calendar and other applications.
- Access to HTML and WAP web pages.

Security considerations

- Device will contain sensitive information.
- Wireless communication must be protected.
- Loss, theft, physical tampering.
- ARM-based device is relatively powerful, but battery life is a factor. Delays (e.g., for crypto ops) unacceptable to users.

Source: Herb Little, Research in Motion

Security Level

- Marketing demands 256-bit AES.
- Equivalent security level for public-key ops: 512-bit ECC or 15360 RSA/DH.
- ECC especially attractive here.
- Timings on BlackBerry 7230 for 128-bit security.

	ECC (256)	RSA (3072)	DH (3072)
Key generation	166 ms	Too long	38 s
Encrypt or verify	150 ms	52 ms	74 s
Decrypt or sign	168 ms	8 s	74 s

Source: Herb Little, RIM.

- RSA-1024 used for code signing (due to verify speed).
  - Bootstrapping process; each level verifies the integrity of next level.

OTA Provisioning

- Previously, initial keying of the device required physical connection to desktop computer.
  - OTA eliminates the desktop requirement.
- SPEKE (simple password-authenticated exponential key exchange) bootstraps strong secret from short password.
  - SPEKE is DH with password used to obtain group elt (P1363).
  - Password is known to both sides.
- OTA is a rare event, but multi-minute delays unacceptable.
  - Provisioning is the first thing a user experiences.
- BlackBerry uses ECC-521 with ECSPEKE.

OTA Re-Key (Key Rollover)

- Establishes a new master key
- Can be initiated by either side, occurs automatically after 30 days.
- BlackBerry uses ECMQV with 521-bit curve.

IT Policy Authentication

- Administrators control devices remotely by IT commands and policies.
  - If the device is lost or stolen, admin can erase all data.
  - Defines password life, encryption methods, browser options.
- BlackBerry signs policy packets with ECDSA.

Content Protection

- Encrypts items stored on device using AES, and messages received while locked using ECC.
- AES and ECC private keys encrypted by 256-bit key derived from password and stored in flash.
  - Decrypted forms never stored in flash while locked.
- At unlock, the AES and ECC private keys are recovered.
- Access to items (message body, subject, recipient) received while locked require ECC operations.
  - 160-, 283-, or 571-bit, depending on policy.
- Content protection uses ECDH for decryption speed.

Other Deployments (present and future)

- Check 21
  - US federal law designed to let banks handle more checks electronically.
  - Electronic cheques can be signed with ECDSA.
  - http://www.federalreserve.gov/paymentsystems/truncation/
- Freescale MPC190 Security Processor
  - Elliptic curve operations in $\mathbb{F}_p$ and $\mathbb{F}_{2^m}$.
  - Programmable field size from 55 to 511 bits.
- US government's Federal Aviation Administration (FAA)
  - Protecting the integrity of communications between air traffic controllers and planes.
- E-passport

Other Deployments (present and future)

- Digital rights management (DRM):
  - Consumer electronics (DTCP)
    www.dtcp.com
  - Advanced access content system (AACS)
    www.aacsla.com
  - Microsoft DRM 2.0 (used in Windows media player)
- Identity-based encryption
  (using low embedding degree elliptic curves).