Certicom's Bulletin of Security and Cryptography

C0DE&C1PHER

Elliptic Curve Cryptography (ECC) is included in numerous standards around the world. This issue of Code and Cipher focuses on three key standards bodies that incorporate ECC: ANSI, IETF and NIST.

Development of Core ECC Standards by ANSI

The first ECC standard developed by an accredited standards body was ANSI X9.62: The Elliptic Curve Digital Signature Algorithm (ECDSA) in 1999. ANSI X9.63: Key Agreement and Key Transport Using Elliptic Curve Cryptography followed in 2001. ANSI was interested in Elliptic Curve Cryptography as far back as 1995 because of its potential for providing strong and efficient security for applications in the financial services industry.

The ANSI X9.62 and X9.63 standards provide detailed specifications of elliptic curve cryptographic protocols for the fundamental tasks of signatures, key agreement, and key transport. These core cryptographic standards are extremely important because they provide the reference base for other standards bodies that develop security and applications standards. For example, the FIPS 186-2 standard describes ECDSA by simply providing a pointer to ANSI X9.62. FIPS 186-2 in turn is very influential in dictating the cryptographic mechanisms deployed within the US federal government.

This article provides some background on ANSI, and summarizes the key elements of the ANSI X9.62 and X9.63 standards for elliptic curve signatures and key establishment.

ANSI: The Organization

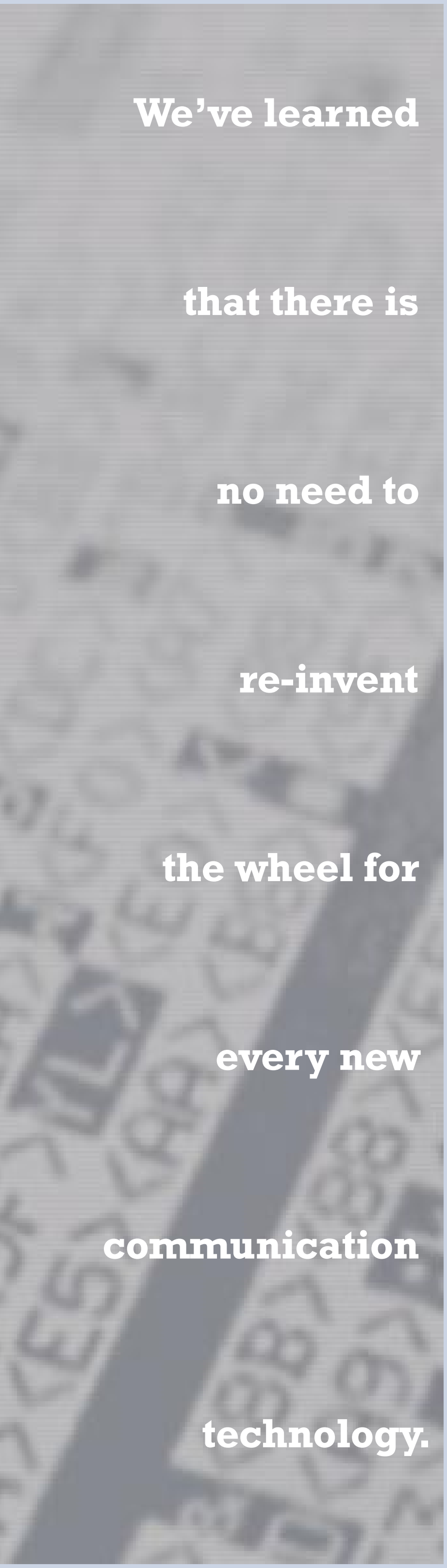
ANSI is a private, non-profit organization whose mission is to promote and facilitate voluntary consensus standards and conformity assessment systems, and safeguard their integrity. There are presently over 11,000 ANSI standards that specify ratings, test methods, performance and safety requirements, systems, and services for a diverse range of industries. The ANSI C standard, familiar to most software developers, is an example of a widely deployed ANSI standard.

ANSI does not itself create standards. Rather, it establishes the consensus procedures that are the basis for the development of a standard, accredits organizations that develop draft standards for a particular sector, and approves these draft standards provided that all procedural requirements have been met. ANSI also promotes the use of US standards internationally via their membership in the International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC), thus facilitating their widespread adoption.

X9 is an ANSI-approved organization that creates standards for the financial services industry, which includes banks and credit card companies. Within X9, the X9F subcommittee deals with data and information security issues. X9F has five working groups:

- **X9F1 – Cryptographic Tools**
- **X9F3 – Protocols**
- **X9F4 – Cryptographic Applications**

continues on page 7



We've learned that there is no need to re-invent the wheel for every new communication technology.

Crypto Column by Dr. Scott Vanstone

The Importance of Good Crypto and Security Standards

This issue of *Code and Cipher* focuses on security standards & associated standards bodies. It takes several years to come up with a good security standard. When developing a new security standard, it is important to choose protocols that have withstood the test of time, such as TLS and IPSec. This way, you can ensure strong and reliable security. The cost of weak standards and implementations spreads far beyond reputation: they can result in the loss of hundreds of millions of dollars to the industry.

WiFi (802.11) is a high profile example of a weak security standard that did use standard cryptographic algorithms, but not existing, proven protocols. The initial release was flawed. What resulted is that the typical enterprise deployment for WiFi became to install wireless access points outside the enterprise network and outside the firewall. While this isn't necessarily bad, it does require the enterprise to deploy a VPN to gain access, and therefore adds more cost.

Now add Voice over IP (VoIP) into the mix. VoIP over the enterprise network is projected to save millions, because the same network lines that carry data can also be used to transmit voice calls.

VoIP transportation and signaling protocols are vulnerable to attack simply because they travel over the existing enterprise data network. If a hacker accesses the voice transport, they can quite easily listen to any call. Attacking the signaling protocol would allow someone to make unlimited international calls or re-direct inbound calls, all at the expense of the hacked organization. These security threats must be dealt with to prevent the associated losses. There are VoIP security standards evolving to address these issues.

Using VoIP over a wireless network would allow a user to be completely mobile in the enterprise for voice and data. However, VoIP and WiFi can't easily coexist in the enterprise today because the access points are outside the enterprise network. This means that you have to run yet another security protocol such as SSL or VPN on top of VoIP and WiFi security. This represents enormous cost to the enterprise–all because of poor choices made during WiFi security standards process.

To be fair, WiFi security standards are evolving; unfortunately the new standards require major hardware upgrades and the addition of authentication servers for the enterprise. It will certainly take some time to see how well these technologies co-exist in the emerging enterprise.

Much of this could have been easily avoided if industry first considered using well established core standards that are discussed in this issue of *Code and Cipher*. We have learned that there is no need to re-invent the wheel for every new communication protocol. In fact, many of the standards described here have modular components for encryption, key agreement and more, so they can be effectively used to address almost any situation.

How ECC Can Improve Internet Communications: ECC and the IETF

As has been demonstrated in past issues of Code and Cipher, ECC is the best option for public-key cryptography when performance is a concern. One area where good performance is important is Internet communication.

The body that looks after setting the standards for Internet communication protocols is the Internet Engineering Task Force (IETF). IETF has been very influential in the cryptographic standards industry by setting the widely used standards for IPSec, TLS and S/MIME.

ECC provides performance benefits for Internet communications, and indeed companies are using ECC today within these protocols. This article summarizes some of the main areas where ECC is being used for Internet communications.

How the IETF works

The IETF is a large, open and international software and community of network designers, operators, vendors, and researchers concerned with the evolution of the Internet architecture and the smooth operation of the Internet. The IETF holds meetings three times a year, and anyone can attend. The rest of the time participants communicate via mailing lists.

The actual technical work of the IETF is done in its working groups, which are organized by topic into eight different areas. Security is one of these groups.

IETF standards begin as Internet Drafts (I-D). To become an IETF standard, the document must be published as an I-D so that interested participants can provide comments and feedback. After a period of time, the draft is presented to the IESG (Internet Engineering Steering Group) for review and publishing, and is assigned an RFC (Request for Comments) number by the RFC Editor.

Elliptic Curve Cryptography can be found in RFCs and I-Ds for all the key Internet protocols: IPSec, TLS, PKIX, and S/MIME.

ECC and IPSec

IPSec is one of the protocols where ECC is in use today. Typically, IPSec uses DH (Diffie-Hellman) for key generation, which is fine for a desktop connection, but is too slow for smaller devices (*see* Figure 1). ECDH (Elliptic Curve Diffie Hellman) provides much faster performance. As more companies want to enable secure connections from constrained devices such as smart phones and handhelds, connection time becomes an important consideration.

	ECDH	DH
20 MhZ Dragonball	1 sec.	40 sec.
206 MhZ ARM	1 sec.	6 sec.

* benchmarking performed by Certicom using movianVPN and a Cisco VPN 3000 Concentrator series.

Figure 1: Relative Key Setup Times (IKE)

Cisco, Shiva and Nortel all support ECDH for key exchange in their gateways. RFC 2409 (*The Internet Key Exchange (IKE)*) describes the use of certain ECC curves for IKE. Cisco, Shiva and Nortel actually base their IKE implementations of ECDH on Internet Drafts that describe stronger ECC curves. They are using *Additional ECC Groups for IKE* (draft-ietf-ipsec-ike-ecc-groups-04.txt), which describes how to use some NIST-recommended curves with IKE.

ECC and SSL/TLS

SSL is short for Secure Sockets Layer, and is a protocol originally designed by Netscape. TLS (Transport Layer Security) is the IETF version of SSL. Version 3.0 of SSL was used as the basis for the IETF TLS standard, version 1.0.

Using ECC for secure transactions makes sense for a number of reasons. Transactions will need to be processed more efficiently – more and smaller devices are being connected to the Internet that require security, from onboard automotive computers to smart cards and process control sensors. ECC uses less bandwidth than alternative cryptographic algorithms for SSL/TLS.

Processing power itself is increasing and hackers have even more resources available to them than ever before. Although 1024-bit RSA keys are currently most often used today, use of 2048 bits is becoming more and more common. Figure 2 shows the impact that this will have in server response time.

According to this chart, in order to handle the same amount of web traffic, someone using RSA instead of ECC would have to purchase and maintain 3.5 times as many web servers in order to handle the same amount of traffic.

continues on page 4

continued from page 3

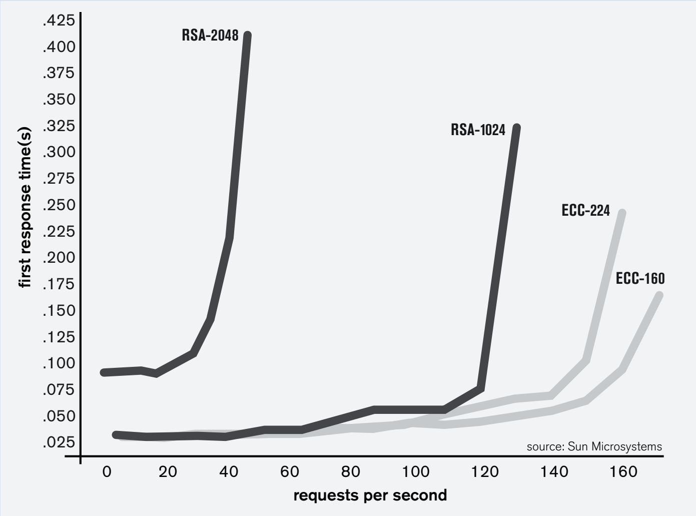


Figure 2: Server Response Times

Certicom, OpenSSL and Sun Microsystems are actively promoting the adoption of ECC into IETF standards, and have implemented ECC in TLS.

ECC Cipher Suites for TLS (draft-ietf-tls-ecc-06.txt) is the IETF draft that discusses new key exchange algorithms based on ECC for the TLS (Transport Layer Security) protocol. In particular, it specifies the use of Elliptic Curve Diffie-Hellman (ECDH) key agreement in a TLS handshake and the use of Elliptic Curve Digital Signature Algorithm (ECDSA) as an authentication mechanism.

S/MIME and ECC

S/MIME can be used to secure email, allowing them to be encrypted, which protects them from being scanned and read. Furthermore, emails can be authenticated using S/MIME, which prevents modification of their contents and forgery of their originating address. This is commonly done in spam email, as well as in the more nefarious form called “Phishing”, where the sender spoofs the address of a legitimate authority, such as a bank or company IT department, in order to induce the recipient to reply with sensitive information.

RFC 3278 (*Use of ECC Algorithms in Cryptographic Message Syntax (CMS)*) defines a profile for the use of Elliptic Curve Cryptography (ECC) public key algorithms such as ECDSA, ECDH and MQV to secure email.

As with SSL, ECC uses less bandwidth and requires less computing power than alternative cryptographic algorithms for S/MIME, which can be critical for devices like PDAs and smartphones, which are beginning to support email. In addition, many of the security benefits of MQV that have been highlighted in previous issues of *Code and Cipher* will apply when MQV is used in S/MIME. In the context of email, using MQV in S/MIME allows a sender to authenticate email to a recipient rather the sender digitally signing the email for anybody to see.

Currently, deployment of S/MIME is still in its early days. As S/MIME adoption increases, use of ECC-based algorithms within S/MIME should also increase.

PKIX and Other IETF References to ECC

RFC 3279 (*Algorithms and Identifiers for the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*) describes the encoding formats for the digital signatures and public keys for ECDSA and ECDH).

This RFC is important not only to PKIX, but also to IPSec, SSL/TLS and S/MIME as well because it describes the certificates that are used for all of these Internet protocols.

There are many other Internet drafts and standards that include ECC:

- ***Elliptic Curve Digital Signature Algorithm (ECDSA) for XML Digital Signatures*** **(draft-blake-wilson-xmldsig-ecdsa-09.txt) specifies how to use ECDSA with XML Signatures.**
- ***Elliptic-Curve Diffie-Hellman Key Exchange for the SSH Transport Level Protocol*** **(draft-stebila-secsh-ecdh-01) describes new key exchange algorithms based on ECC for the Secure Shell (SSH) protocol.**

The Future of ECC and the IETF

There are a number of ongoing efforts to have ECC incorporated into IETF standards. Moreover, companies are already using ECC-based algorithms today in their Internet communications. Elliptic Curve Cryptography is emerging as the best option where performance is a concern, and as such, it is important to ensure that it continues to be written into drafts and adopted as RFCs. ■

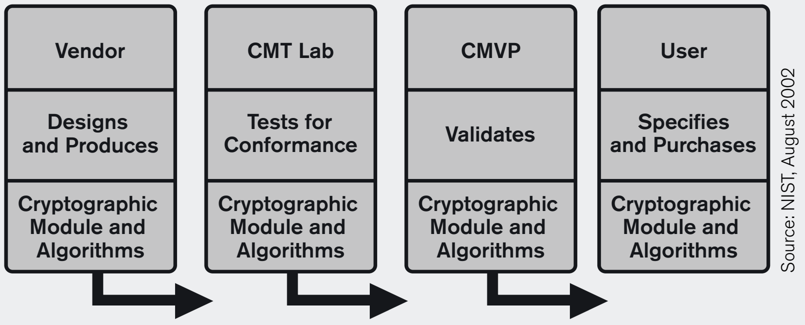
For more information:
The IETF – **www.ietf.org**
IPSec RFC: **http://www.faqs.org/rfcs/rfc2409.html**
SMIME RFC: **http://www.faqs.org/rfcs/rfc3278.html**
PKIX RFC: **http://www.faqs.org/rfcs/rfc3279.html**
ECC Cipher Suites for TLS (draft-ietf-tls-ecc-06.txt)
ftp://ftp.rfc-editor.org/in-notes/internet-drafts/draft-ietf-tls-ecc-06.txt

The Growing Importance of ECC in FIPS 140-2

In the US, requirements for government security are regulated by FIPS (Federal Information Processing Standard) publications, which are developed by NIST (National Institute of Standards for Technology) for use government-wide. NIST develops FIPS when there are compelling federal government requirements for security and interoperability and there are no acceptable industry standards or solutions. Other countries such as the United Kingdom and Canada are also starting to refer more to FIPS standards. Outside of government, other industries, such as financial and postal, also refer to FIPS.

One of the key standards is FIPS 140-2, which describes US federal government requirements that software and hardware products must meet for sensitive but unclassified use. FIPS 140-2 evaluation is currently a requirement for sale of products implementing cryptography to the federal government.

The FIPS 140-2 Validation Process



Vendor
Designs and Produces
Cryptographic Module and Algorithms
CMT Lab
Tests for Conformance
Cryptographic Module and Algorithms
CMVP
Validates
Cryptographic Module and Algorithms
User
Specifies and Purchases
Cryptographic Module and Algorithms

Source: NIST, August 2002

Key Points

- **The applicant cannot work directly with NIST**
- **The lab submits an application on behalf of applicant**
- **The process takes approximately 12 months**

Along with RSA and DSA, ECDSA is one of three FIPS-approved methods for asymmetric key functions within FIPS 140-2. Currently, however, unlike RSA and DSA, which have validation systems, ECDSA implementations can only be listed as vendor affirmed. Vendor affirmation represents a commitment on the part of the vendor to have implemented the algorithm correctly.

With the growing interest in ECDSA and other ECC-based algorithms for government and financial use, however, a validation system for ECDSA would ensure consistent and secure implementations.

The Process

For algorithms that do have a validation system, the process works as follows:

- **the algorithm implementation is submitted to third party lab who test on behalf of NIST;**
- **the implementation is tested for conformance and assigned an algorithm certificate number;**
- **the numbering of these certificates is done independently for each algorithm, in the order the certificates are issued.**

When dealing with FIPS 140-2 validation, all algorithms must be submitted and shown that they work properly within the module, either with an algorithm certificate or with vendor affirmation if a validation system is unavailable.

Risk versus return: Vendor Affirmed versus Validated

An incorrectly implemented algorithm can lead to security weaknesses and interoperability problems between products of different vendors (e.g. a browser and a CA (Certification Authority)).

A validation system provides the following benefits:

1. **By putting the algorithm through a series of tests in specifically defined areas, the validation system can detect significant implementation errors.**
2. **A common validation system helps ensure a consistent implementation of a particular algorithm on a specific platform.**

Combined, the above two benefits provide an adequate level of security and interoperability.

Although ECDSA is only vendor affirmed, it is in fact approved for use in FIPS 140-2 when implemented as per FIPS 186-2 (Digital Signature Standard).

continues on page 6

continued from page 5

To trust a "vendor affirmed" implementation of ECDSA (or any other algorithm that does not have a validation system), you have to ensure that the vendor has the expertise to implement the algorithm correctly. ECC-based algorithms can be difficult to implement – and today, that expertise is in the hands of a select few companies. A validation system from NIST could provide the third party validation needed to encourage wider use of ECDSA in other implementations.

MQV and FIPS

In 2003, the National Security Agency (NSA) selected ECC, and in particular MQV (Menezes-Qu-Vanstone), as a crucial technology for protecting mission critical national security information. This further validates the important role that ECC will be playing in the future for the government.

MQV is also moving towards FIPS adoption through Special Publication 800-56. While a Special Publication is not binding the same way that a FIPS publication is, it is a key step on the road to being incorporated into a FIPS. MQV was also one of the pieces of intellectual property recently licensed from Certicom by the NSA.

Along with ECDSA, the key establishment algorithms MQV and ECDH (Elliptic Curve Diffie-Hellman) are approved for use in FIPS 140-2 modules in FIPS-approved mode for key establishment.

Time for an ECC-based Validation system

FIPS 140-2 is required for sale of products implementing cryptography to the federal government. Because of the high level of security ensured by FIPS, the financial and healthcare industries are also starting to mandate FIPS 140-2 to secure their transactions.

The benefit of having validation systems for ECC algorithms is that accredited third parties would test implementations. This would speed up the related but separate FIPS Validation process (which is on its own lengthy and costly) and ensure interoperability across systems. Both of these benefits would make it easier for ECDSA and MQV to be embraced in other areas.

ECDSA has been widely accepted for use in financial and postal industries – it is specified in ANSI X9F (X9.62) and USPS Postal standards. With the recent signing of Check 21 (Check Cashing Act for the 21st Century), ECDSA will become more important, as it is the only algorithm that can handle the requirement for check image verification systems that must process a high volume (10,000 +) of images per minute.

As interest in ECC-based algorithms grows, it will become more important to have validation systems for them. Given the widespread use of ECDSA today, it only makes sense to start there. ■

A complete list of validated algorithms can be found at: **http://csrc.nist.gov/cryptval/vallists.htm**

To read more about implementing a FIPS 140-2 cryptographic module, visit **http://www.certicom.com/fips**

NIST Special Publication 800-56
http://csrc.nist.gov/CryptoToolkit/kms/keyschemes-Jan03.pdf

FIPS 140-2: Security Requirements for Cryptographic Modules
http://csrc.nist.gov/publications/fips/fips140-2/fips1402.pdf

NIST is currently developing a validation system for ECDSA. They are using a validation system submitted by Certicom as a reference.

Development of Core ECC Standards by ANSI

continued from page 1

- **X9F5 – Digital Signature and Certificate Policy**
- **X9F6 – Cardholder Authentication and ICC's**

The X9F1 working group is responsible for developing the core cryptography standards that specify symmetric-key and public-key algorithms for encryption and authentication.

The time between when a cryptographic algorithm is first proposed to the working group to when it eventually is included in an official ANSI standard can be as long as five years. During this time, the algorithms are thoroughly scrutinized by experts, while details such as parameter choices and data formatting are debated by the members of the working group until a consensus is reached. Working group members typically consult with the experts and implementers at their home institutions who provide valuable feedback during the evolution of the standard. An approved standard is reviewed every five years and updated as necessary.

Some of the important cryptographic standards that have been produced by the X9F1 working group are the following:

■ **X9.30:1**	**The Digital Signature Algorithm**
■ **X9.30:2**	**The Secure Hash Algorithm**
■ **X9.31**	**Digital Signatures Using Reversible Public Key Cryptography**
■ **X9.42**	**Agreement of Symmetric Keys Using Discrete Logarithm Cryptography**
■ **X9.52**	**Triple Data Encryption Algorithm Modes of Operation**
■ **X9.62**	**Elliptic Curve Digital Signature Algorithm (ECDSA)**
■ **X9.63**	**Key Agreement and Key Transport Using Elliptic Curve Cryptography**
■ **X9.80**	**Prime Number Generation, Primality Testing, and Primality Certificate**

X9.30 specifies the Digital Signature Algorithm (DSA), while X9.31 describes a particular variant of the RSA signature scheme.

ANSI X9.62: Elliptic Curve Digital Signature Algorithm (ECDSA)

X9.62, which was finalized in 1999, is notable for being the first standard in the world to specify an elliptic curve cryptographic protocol. It includes a detailed description of the finite field and elliptic curve parameters, and how they are to be represented. There are many possible representations for these parameters, and therefore fixing a particular representation is crucial for interoperability. Also included are detailed descriptions of the key generation and validation procedures, and the ECDSA signature generation and verification routines. There are extensive appendices that provide the relevant mathematics background, present algorithms for implementing finite field and elliptic curve operations, generation of elliptic curve parameters, and discuss various issues regarding the secure use of ECDSA. These appendices are a valuable resource to security engineers wishing to understand and implement elliptic curve cryptography.

A primary objective of the X9.62 standard was to achieve high degrees of security and interoperability. For this reason the finite fields were restricted to being either a prime field or a binary field. Moreover, preferred representations for the elements of these fields are given. A minimum field size of 160 bits is mandated, which translates to an 80-bit security level (equivalent to the security afforded by 1024-bit RSA or 160-bit ECC).

The early adoption of X9.62 had a strong influence on the progression of other standards for elliptic curve signature schemes. Many other standards for ECDSA were subsequently developed which, with a judicious choice of parameters, are compliant with X9.62. These include IEEE 1363-2000, FIPS 186-2, and ISO 15946-2. In fact, the FIPS 186-2 standard, which is part of the suite of cryptographic standards crafted by the US government's National Institute for Standards and Technology (NIST), specifies ECDSA simply by providing a pointer to ANSI X9.62. In addition, it lists a set of 15 elliptic curves that are recommended (but not mandated) for US federal government use.

ANSI X9.62 is presently going through the last stages of its first five-year review. The major changes proposed for the revision are (i) the inclusion of the SHA-224, SHA-256, SHA-384 and SHA-512 variants of the hash function SHA-1 that provide greater levels of security; (ii) the exclusion of binary fields of order 2^m for composite m to circumvent potential attacks on the elliptic curve discrete logarithm problem in this case; and (iii) the inclusion of the 15 elliptic curves from FIPS 186-2 as sample parameter sets. The revision to ANSI X9.62 is expected to be ratified in late 2004.

continues on page 8

Development of Core ECC Standards by ANSI

continued from page 7

ANSI X9.63: Key Agreement and Key Transport Using Elliptic Curve Cryptography

Another elliptic curve cryptographic standard, ANSI X9.63, was finalized in 2001 and specifies several protocols for key establishment. There are protocols for key transport and key agreement, including one-pass, two-pass and three-pass variants of the MQV key agreement protocol. There is an extensive treatment of the security attributes possessed by each protocol so that users can make an informed choice of the protocol most suitable for their application. The representations of data elements in X9.63 are consistent with those in X9.62. The description of MQV is consistent with other standards including ISO 15946-3, IEEE 1363-2000, and NIST's Special Publication 800-56 (Recommendation on key establishment schemes).

Other ANSI ECC Standards

The X9F1 working group is finalizing drafts of two other standards that use elliptic curve cryptography. X9.82 specifies several methods for generating pseudorandom numbers, including techniques that use elliptic curve operations. X9.92 describes the Pintsov-Vanstone signature scheme – an elliptic curve-based digital signature mechanism that has short signatures and is especially well suited for environments such as digital postal marks where bandwidth is severely limited. ■

For further information:

To purchase ANSI standards: **http://www.ansi.org**

X9 committee: **http://www.x9.org**

NIST's cryptographic toolkit: **http://csrc.nist.gov/CryptoToolkit**

Code and Cipher

Code and Cipher, published quarterly by Certicom Corp., is an educational newsletter that covers the security and cryptography industry. In each issue we will examine security issues and cryptography trends in an objective manner. We welcome your thoughts, opinions and comments on anything that affects the industry. Please send us your feedback on this issue and what you'd like to see in upcoming ones:

5520 Explorer Drive, 4th Floor
Mississauga, Ontario, L4W 5L1
Canada

T (905) 507-4220
F (905) 507-4230
E codeandcipher@certicom.com

Crypto News

Colossus Rebuilt

The Colossus Mk2, a wartime code-breaker that helped to turn the course of World War II, has successfully been rebuilt and is now on display at Bletchley Park, the hub of British code operations. The original Colossus played a key role in deciphering the Lorenz code used by Hitler. It was destroyed along with nine other Colossus machines after the end of the war.

The Colossus rebuild project began in 1993 using photos and illegally kept diagrams. The working replica is capable of breaking the same ciphers it cracked during the war and was unveiled to veteran coders in June, 2004 to mark the 60th anniversary of the first time the switch was flipped. ■

For more information on the project itself, visit
http://www.codesandciphers.org.uk/lorenz/rebuild.htm

William Tutte, one of the researchers whose work led to the development of the Colossus, was profiled in the first issue of Code and Cipher.

Crypto Challenges Solved

On April 27, 2004, RSA Security Inc. and Certicom Corp. both announced winners of cryptographic challenges. The challenges were each started as a way to stimulate further research in the security analysis of cryptosystems and increase industry understanding and appreciation for the strength of a variety of key size levels.

The RSA-576 Factoring Challenge was to determine the two prime factors of a number that is 576 bits in length. The security of RSA with a 576-bit modulus is based on the difficulty of factoring such numbers. A multinational team of eight experts used about 100 workstations to factor this number in about 3 months.

The Certicom (ECC)2-109 bit challenge was solved by Chris Monico, an assistant professor at Texas Tech University and a team of mathematicians. Their effort required 2600 computers and took 17 months.

A prize of $10,000 US was awarded in each of the challenges. Both of these strengths are well below the crypto that is used commercially today. ■

For more information about the Certicom ECC challenge, visit:
http://www.certicom.com/ecc

For more information about the RSA Challenge, visit:
http://www.rsasecurity.com/rsalabs/node.asp?id=2091