Birth of SET

SEPP
(Secure Electronic Payment Protocol)
Master Card, IBM, Netscape
GTE

+

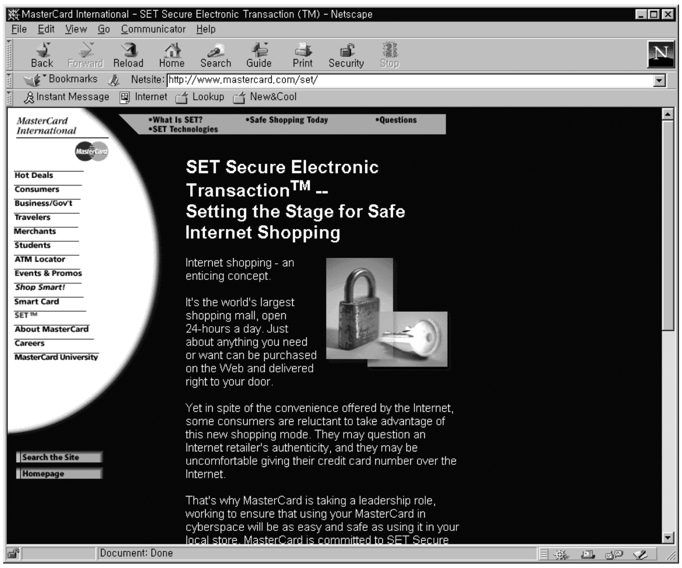
STT
(Secure Transaction Technology)
Visa, Microsoft

⇩

'96.2
SET(Secure Electronic Transaction)
Master Card, Visa, GTE, Microsoft,
Verisign, SAIC, IBM, Netscape, RSA
'96.5
Release SET 1.0

Background of SET

- Increase of Internet-based on-line transactions
- Support new way of EC by major banks
- Under way to implement electronic purse/ cash system
- Easy Web access
- Provide Multimedia information
- Change of customer's purchasing style
- Easy to provide cryptographic services to payment system



Role of SET in EC

- Browsing and Shopping
- Merchant Item Selection
- Negotiation and Ordering
- Payment Selection
- Payment Authorization and Transport
- Confirmation and Inquiry
- Delivery of Goods
- Merchant Reimbursement

Req't in Business side

1.Provide confidentiality of payment information (PI) and enable confidentiality of order information (OI) that is transmitted along the payment information.

2.Ensure the integrity of all transaction data.

3.Provide authentication that a cardholder is a legitimate user of a branded payment card account.

4. Provide authentication that a merchant can accept branded payment card transactions through its relationship with an acquiring financial institutions.

Req't in Business side(II)

5. Ensure the use of the best security practices and system design techniques to protect all legitimate parties in an electronic commerce transaction.

6. Create a protocol that neither depends on transport security mechanisms nor prevents their use.

7. Facilitate and encourage interoperability among software and network providers

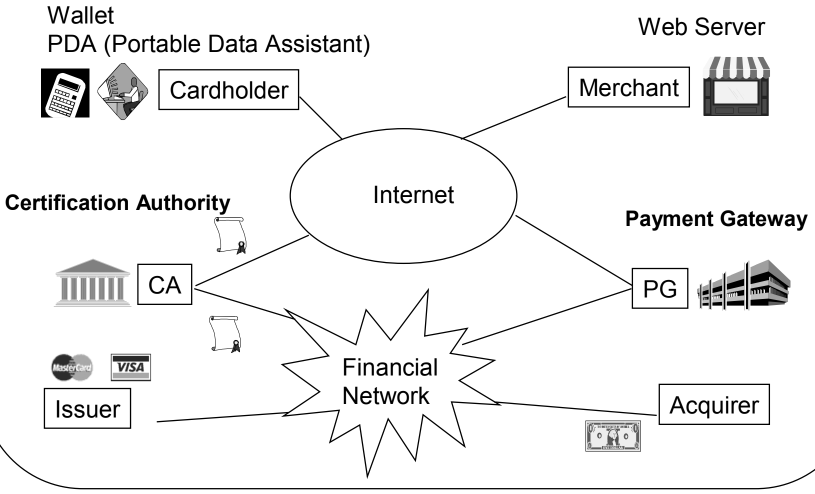
Properties of SET(I)

- Different parts of a SET transaction might require:
  - Confidentiality of information
  - Integrity of data
  - Cardholder and account authentication
  - Merchant authentication
  - Interoperability
- Trusted cryptographic techniques are used to provide these properties.

Properties of SET(II)

- While cryptographic algorithms are the focus here, they are only the starting point.
- Message handling is built around the PKCS#7 cryptographic message syntax standard.
- Certification is built around X509.v3 certificates
  - with both standard and custom extensions.
- Built around secure components
  - the basic algorithms have been standardized and are in widespread use.
- Uses some of the latest research results
  - novel and innovative features such as HMAC and OAEP are included.

Components of SET(I)



References

(1) *Answers to Frequently Asked Questions about Today's Cryptography,* Paul Fahn, RSA Laboratories, 1993. (http://www.rsa.com/rsalabs/faq/)

(2) *Applied Cryptography, Second Edition*, Bruce Schneier, John Wiley & Sons, Inc., 1996.

(3) "Asymmetric Encryption: Evolution and Enhancements," Don B. Johnson and Stephen M. Matyas, *CryptoBytes,* volume 2, number 1, Spring 1996

(4) *BSAFE 2.1*™, RSA Data Security, Inc., 1994. (http://www.rsa.com/rsa/prodspec/bsafe/rsa_bsaf.htm)

(5) *Data Encryption Standard*, Federal Information Processing Standards Publication 46, 1977.

(6) "The HMAC Construction," Mihir Bellare, Ran Canetti, and Hugo Krawczyk, *CryptoBytes,* volume 2, number 1, Spring 1996

(7) *HTML Sourcebook*, Ian S. Graham, John Wiley & Sons, Inc., 1995.

(8) *The Internet for Everyone: A Guide for Users and Providers*, Richard W. Wiggins, McGraw-Hill, Inc., 1995.

(9) *Optimal Asymmetric Encryption*, M. Bellare and P. Rogaway, Eurocrypt 94. (http://www-cse.ucsd.edu/users/mihir/papers/oae.ps.gz)

(10) *An Overview of the PKCS Standards*, Burton S. Kaliski, Jr., RSA Laboratories, 1993. (http://www.rsa.com/pub/pkcs/doc/ or http://www.rsa.com/pub/pkcs/ps/)

(11) *Public-Key Cryptography Standards (PKCS)*, RSA Data Security, Inc., Version 1.5, revised Nov. 1, 1993.

(12) *Extensions and Revisions to PKCS #7*, RSA Data Security, Inc., May 13, 1997.

(13) *ITU Rec. X.509 (1993) | ISO/IEC 9594-8: 1995*, including Draft Amendment 1: Certificate Extensions (Version 3 certificate).

(14) *RFC 1750, Randomness Recommendations for Security, D. Eastlake, S. Crocker, J. Schiller, December 1994*.

Algorithm

- Symmetric encryption
  - DES (Data Encryption Standard) : 56bit key, protect financial data
  - CDMF (Commercial Data Masking Facility) : 40 bit key, protect acquire-to-cardholder message
- Asymmetric encryption and digital signature : RSA
- Hash function : SHA-1
- Message Authentication Code : HMAC (based on SHA-1)

Asymmetric Cryptosystem(I)

- RSA can be used as both
  - the public-key component of the digital envelope, and as
  - a digital signature algorithm
- RSA is widely used and is well trusted.
- The RSA keys in SET will resist todays most dedicated attacker (even when allowing for some possible factoring improvements).

Asymmetric cryptosystem(II)

- For SET, the RSA modulus is 1024 bits in length.
- Using the latest factoring results it appears that factoring a 1024-bit modulus would require over 100,000,000,000 MY of computational effort.
  - note that to factor RSA-129 eight calendar months were required to accumulate 5,000 MY of computational effort
- While factoring the RSA modulus may be infeasible we still have to be careful to use RSA correctly.
- One of the innovations of SET is the use of the OAEP method of message formatting prior to RSA encryption.

Asymmetric cryptosystem(III)

Entity	Message Signature	Key-Exchange	Certificate Signing	CRL Signing
Cardholder	1024			
Merchant	1024	1024		
Payment Gateway	1024	1024		
Cardholder CA	1024	1024	1024	
Merchant CA	1024	1024	1024	
Payment Gateway CA	1024	1024	1024	1024
Brand Geo-political CA			1024	1024
Brand CA			1024	1024
Root CA			2048	2048

Electronic Envelope(I)

- Encrypt the long message with DES then encrypt the DES key with RSA.
- This combines the encryption speed of DES with the key management advantages of RSA public-key encryption.

Electronic Envelope(II)

- The sender
  1. Encrypt the message using a randomly generated symmetric encryption key
  2. Encrypts the symmetric key using the recipient's public key
  3. Sends the encrypted message and the digital envelope.
- The recipient
  1. Recovers the symmetric key by decrypting the digital envelope with his private key
  2. Obtains the original message by decrypting the encrypted message with the recovered symmetric key.

SHA-1

- SHA-1 (1994) is the hash function specified in FIPS 180-1.
- A good hash function with many interesting properties.
- SHA-1 is used within SET
  - to optimize digital signatures
  - as a building block in the HMAC construction
  - as a crucial component in the OAEP formatting used for RSA encryption

HMAC

- HMAC (1996) is a design for a message authentication code that builds on the properties of a hash function.
- Development was implementation driven:
  - reuse existing hash function code
  - offer good software performance
- Good theoretical basis for security.
- Increasingly popular and used widely.

Dual Signature(I)

- Suppose that C wants to send M_1 (OI) to M and M_2 (PI) to B in such a way that
  1. M can't see M_2 and B can't see M_1, but
  2. Two messages are linked together
- Then C first generates the signature for M = $H(M_1) \| H(M_2)$ as $Sig_M = S_C(H(M))$ and then sends $\{Sig_M, E_B(M_1), H(M_2)\}$ to M and $\{Sig_M, H(M_1), E_C(M_2)\}$ to B
- Application : used to link an payment order sent to the merchant with the payment instructions containing account information sent to the acquirer.

Semantic Security

- An Encryption Scheme [*G,E,D*] is said to be semantically secure if for every ensemble $X=\{X_n\}_{n\in N}$ of polynomial random variables, for every polynomial function *h*, for every function *f*, and for every probabilistic polynomial-time algorithm *A*' s.t. for every constant *c* >0 and for every sufficient large *n*,

$$\Pr[A\,(E_{G(1^n)}(X_n),h(X_n),1^n) = f(X_n)] \leq \Pr[A'(h(X_n),1^n) = f(X_n)] + 1/n^c$$

where the probability is taken over the coin tosses of *A* (resp. *A*'), *E* and *G*, and the distribution of *X*. Intuitively, given any *a-priori* information, $h(X_n)$, no algorithm *A* can obtain some information $f(X_n)$, from the ciphertext that could not have been efficiently computed by A' without the ciphertext.

Indistinguishability

- **An Encryption Scheme [*G,E,D*] is said to be secure in the sense of <u>indistinguishability</u> if, for every probabilistic polynomial time algorithm *F* (for fixed), for every probabilistic poly-time algorithm *A*, for every constant *c* >0,**

  and for every sufficiently large *n*,

$$Pr[F(1^n)= (\alpha,\beta,\gamma) \text{ s.t. } |Pr\{A(\gamma),E_{G(1^n)}(\alpha))=1] - Pr[A(\gamma, E_{G(1^n)}(\beta))=1\}| >1/n^c] < 1/n^c$$

  where the probability is taken over the coin tosses of *F*,*A*, *E* and *G.*
- **Indistinguishable enough ==> semantically secure.**

Non-malleability

- Requires that it is infeasible, <u>given a ciphertext, to create a different ciphertext s.t., their plaintext are related</u>.
- Extension of chosen ciphertext security in that it considers security and self-protection of sender in the context of a network of users, and not simply between a sender and a receiver.

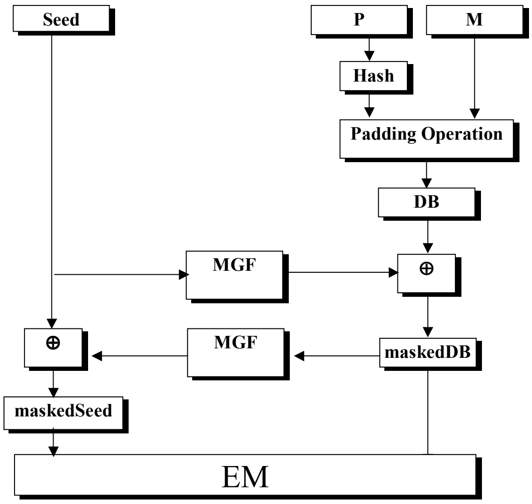
OAEP(I)

- The use of OAEP (1994) moves us on from more *ad hoc* methods of formatting blocks prior to RSA encryption.
- OAEP ties the security of RSA encryption closely to that of the basic RSA operation.
- The version of OAEP used in SET is a more advanced version of the original scheme.
- While existing message formatting methods for RSA encryption have no known flaw, the provable security aspects of OAEP are very appealing.
- OAEP is very new but already it is a part of the IEEE P1363 standards effort.

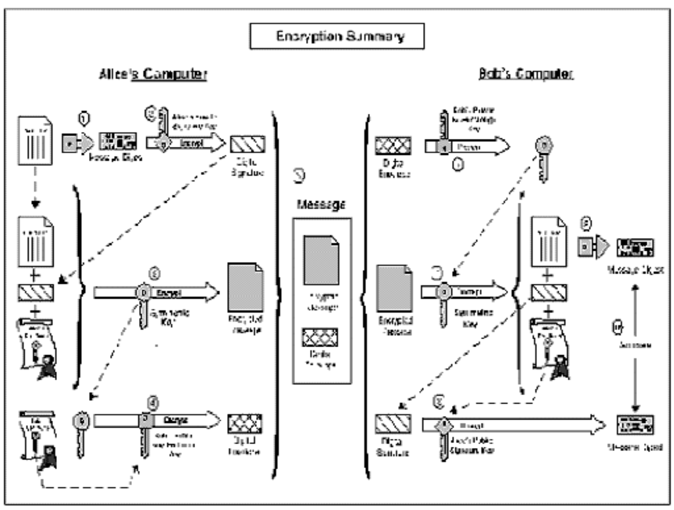
OAEP(II)

- Let $n=k-k_0-k_1$ and f,G,H be such that
  - f : $\{0,1\}^k \rightarrow \{0,1\}^k$; trapdoor permutation,
  - G : $\{0,1\}^{k_0} \rightarrow \{0,1\}^{n+k_1}$; random generator,
  - H : $\{0,1\}^{n+k_1} \rightarrow \{0,1\}^{k_0}$; random hash function
- To encrypt $x \in \{0,1\}^n$, choose a random k_0-bit r and compute the ciphertext y as $y=f(x0^{k_1} \oplus G(r) \,||\, r \oplus H(x0^{k_1} \oplus G(r)))$
- The above encryption scheme achieves non-malleabibility and chosen-ciphertext security assuming that G, H are ideal.
- From theory to practice : derive G,H from some standard cryptographic hash function.

OAEP(III)



Seed
P
M
Hash
Padding Operation
DB
MGF
⊕
MGF
⊕
maskedDB
maskedSeed
EM



Encryption Summary
Alice's Computer
Bob's Computer
Message

Role of CA

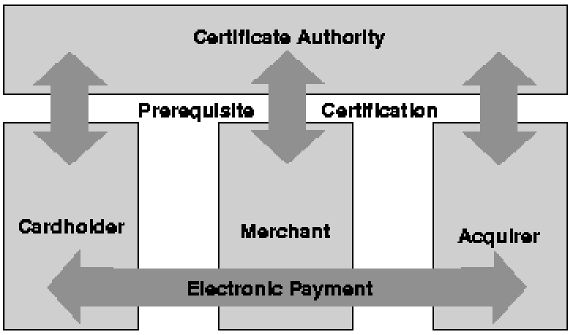
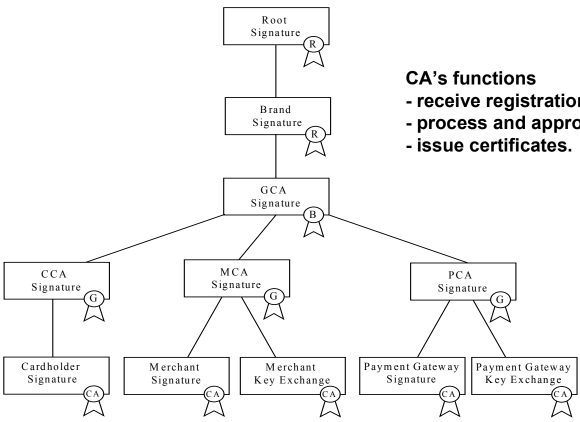


Figure 1: Payment System Participants

Hierarchy of Trust

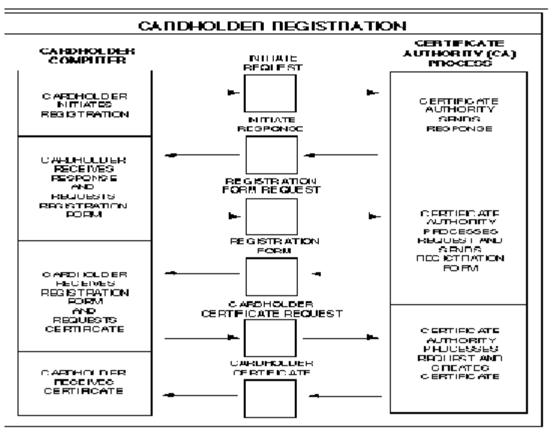


CA's functions
- **receive registration requests**
- **process and approve/decline requests**
- **issue certificates.**

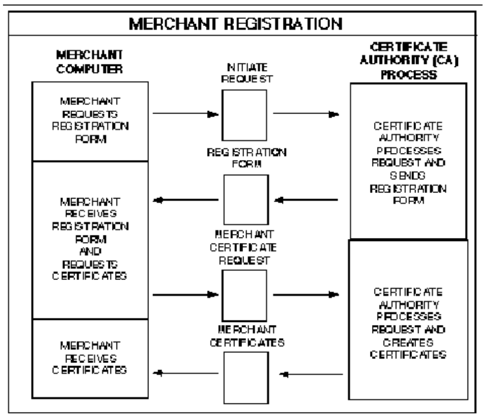
Certificates Types

Certificate Types	Digital Signature	KeyEncryption	Certificate & CRL Signing
Cardholder	X		
Merchant	X	X	
Payment Gateway	X	X	
Cardholder CA	X	X	X
Merchant CA	X	X	X
Payment Gateway CA	X	X	X
Brand Geo-political CA	X		X
Brand CA			X
Root CA			X

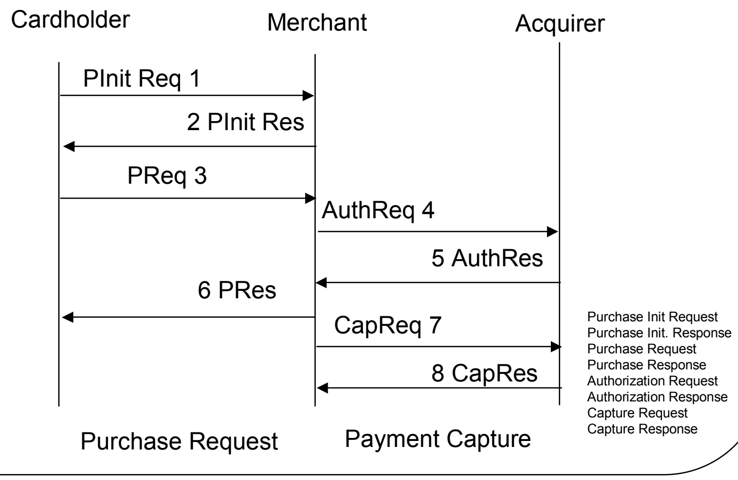
Cardholder Registration



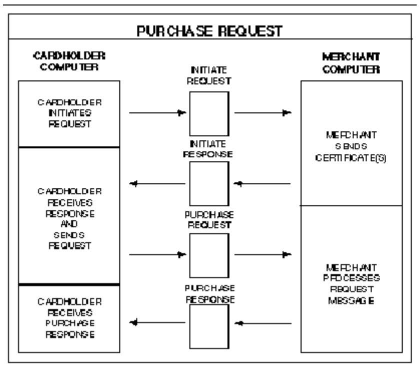
Merchant Registration



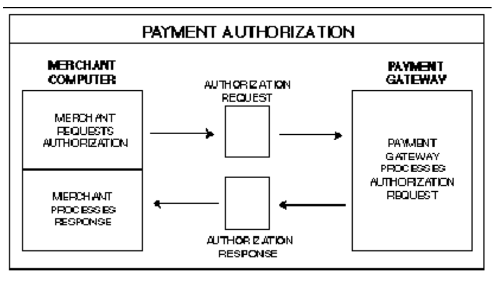
Basic purchase protocol



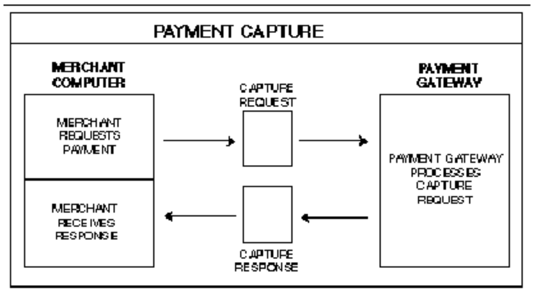
Purchase Request



Payment Authorization



Payment Capture



SET V.2*(I)

- Functionality Enhancements
  - Chip Cards
    - ✓ EMV
    - ✓ Multi-application
    - ✓ Other non-EMV
  - DEBIT
    - ✓ PIN for on-line Debit
    - ✓ PIN pads (PIN entry not from keyboard)

*'98.7.18.

SET V.2(II)

- Encryption Alternatives
  - Algorithm Independence
  - Hardware Vendor Support : Back Key Data
  - Separate symmetric key from Account Information
- Certificate Enhancements
  - Certificates with fewer bytes
  - Formatted Registration Forms (HTML)

 37

Algorithms

Algorithm	Now	Near-Future	Future
Symmetric (encrypts order instruction)	DES	Triple DES ?	(AES)
Hash (digests message)	SHA-1	?	?
Asymmetric (data integrity for authentication; key management)	RSA	ECC (ElGamal+Diffie Hellman+DSA)	?

 38

SET V.2(III)

- Order Enhancements
  - Multiple Payment Instruction on a single order
  - Order Cancellation
  - Re negotiation of Order Description
  - Delivery Receipt for electronic delivery
- Payment Enhancements
  - Payment Negotiation
  - Funds Transfer
  - Purchasing Card Support
  - Travel Agent Business Model