# Secure Electronic Transaction (SET) Protocol

Secure Electronic Transaction or SET is a system which ensures security and integrity of electronic transactions done using credit cards in a scenario. SET is not some system that enables payment but it is a security protocol applied on those payments. It uses different encryption and hashing techniques to secure payments over internet done through credit cards.

SET protocol was supported in development by major organizations like Visa, Mastercard, Microsoft which provided its Secure Transaction Technology (STT) and NetScape which provided technology of Secure Socket Layer (SSL).

SET protocol restricts revealing of credit card details to merchants thus keeping hackers and thieves at bay. SET protocol includes Certification Authorities for making use of standard Digital Certificates like X.509 Certificate. Before discussing SET further, let's see a general scenario of electronic transaction, which includes client, payment gateway, client financial institution, merchant and merchant financial institution.
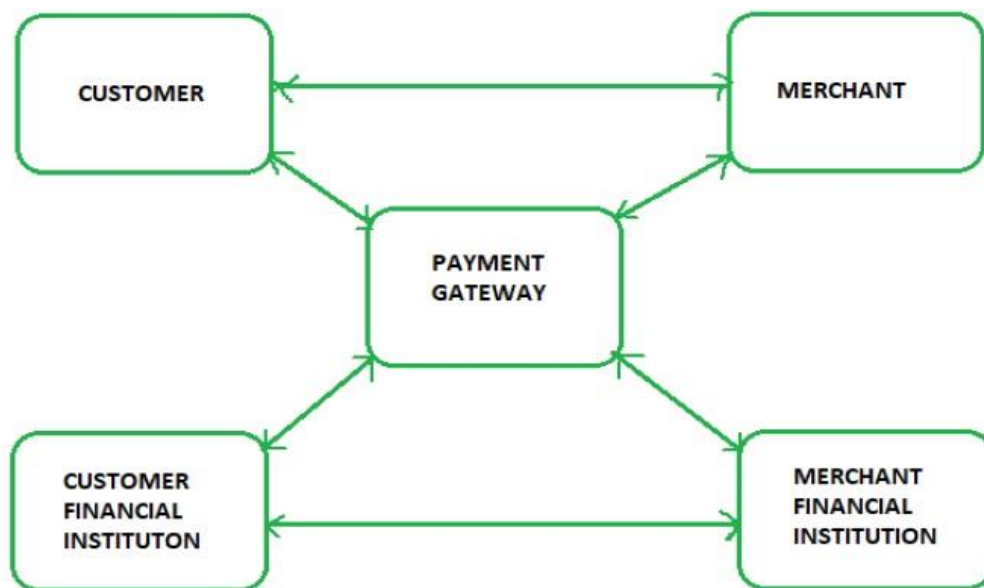
Requirements in SET :

SET protocol has some requirements to meet, some of the important requirements are :

- It has to provide mutual authentication i.e., customer (or cardholder) authentication by confirming if the customer is intended user or not and merchant authentication.
- It has to keep the PI (Payment Information) and OI (Order Information) con_dential by appropriate encryptions.
- It has to be resistive against message modi_cations i.e., no changes should be allowed in the content being transmitted.
- SET also needs to provide interoperability and make use of best security mechanisms.

Participants in SET :

In the general scenario of online transaction, SET includes similar participants:

1. Cardholder – customer
2. Issuer – customer financial institution

3. Merchant

4. Acquirer – Merchant financial

5. Certificate authority – Authority which follows certain
   standards and issues certificates(like X.509V3) to all
other
   participants.

SET functionalities :

❖ **Provide Authentication**

- **Merchant Authentication** – To prevent theft, SET allows
  customers to check previous relationships between
  merchant and financial institution. Standard X.509V3
  certificates are used for this verification.

- **Customer / Cardholder Authentication** – SET checks if use
  of credit card is done by an authorized user or not using
  X.509V3 certificates.

❖ **Provide Message Confidentiality** : Confidentiality refers
  to preventing unintended people from reading the message
  being transferred. SET implements con_dentiality by using
  encryption techniques. Traditionally DES is used for
  encryption purpose.

❖ **Provide Message Integrity** : SET doesn't allow message
  modification with the help of signatures. Messages are
  protected against unauthorized modi_cation using RSA
  digital
  signatures with SHA-1 and some using HMAC with SHA-1.

Dual Signature :

The dual signature is a concept introduced with SET, which aims
at connecting two information pieces meant for two different
receivers:

Order Information (OI) for merchant

Payment Information (PI) for bank

You might think sending them separately is an easy and more secure way, but sending them in am connected form resolves any future dispute possible. Here is the generation of dual signature:
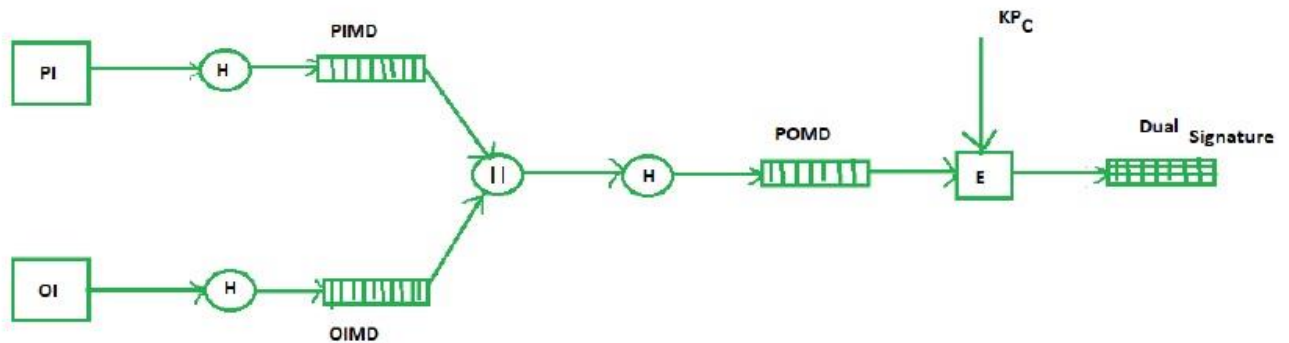
Where,

PI stands for payment information

OI stands for order information

PIMD stands for Payment Information Message Digest

OIMD stands for Order Information Message Digest

POMD stands for Payment Order Message Digest

H stands for Hashing

E stands for public key encryption

KPc is customer's private key
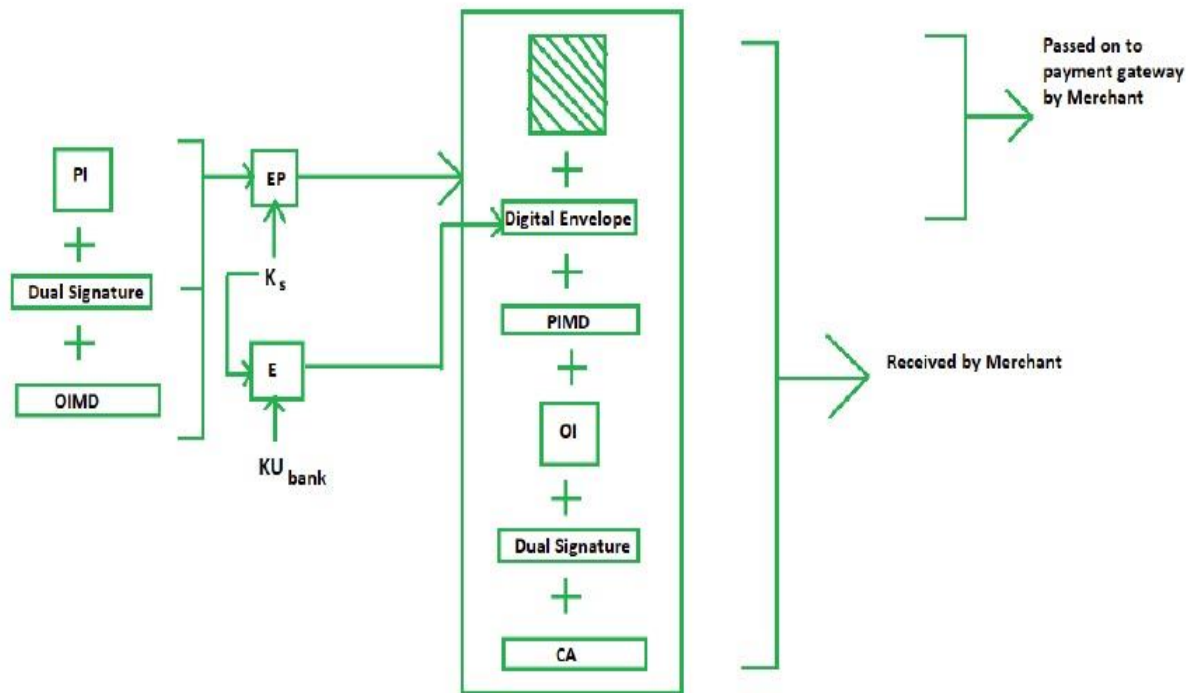
|| stands for append operation

Dual signature, DS= E(KPc, [H(H(PI)||H(OI))])

Purchase Request Generation :

The process of purchase request generation requires three inputs:

- Payment Information (PI)
- Dual Signature
- Order Information Message Digest (OIMD)

The purchase request is generated as follows:

Here,

PI, OIMD, OI all have the same meanings as before.

The new things are :

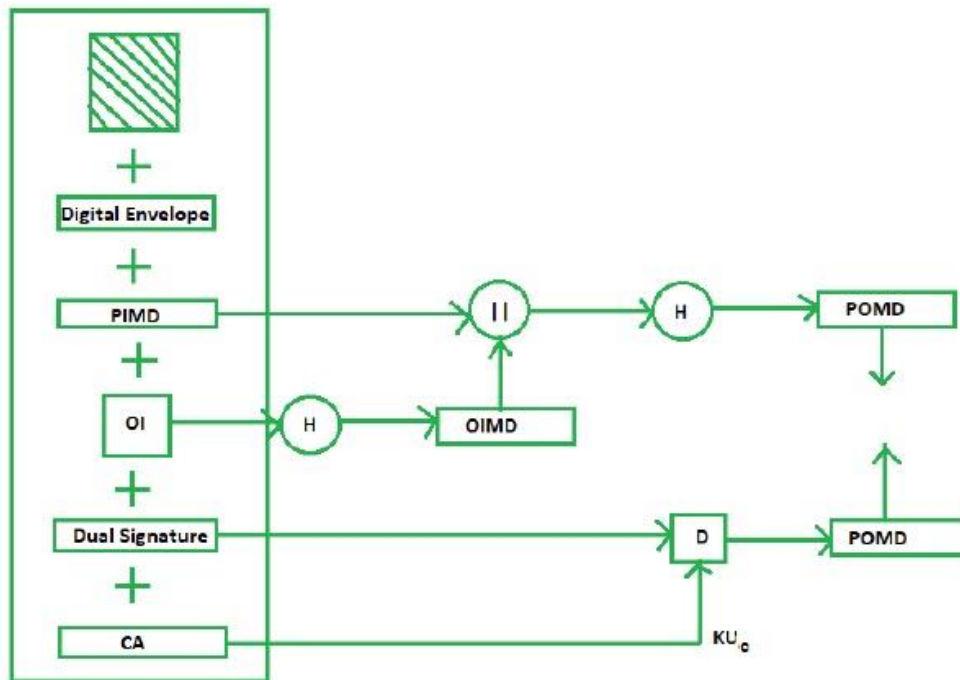EP which is symmetric key encryption

Ks is a temporary symmetric key

KUbank is public key of bank

CA is Cardholder or customer Certificate

Digital Envelope = E(KUbank, Ks)

Purchase Request Validation on Merchant Side :

The Merchant veri_es by comparing POMD generated through PIMD hashing with POMD generated through decryption of Dual Signature as follows:

Since we used Customer private key in encryption here we use KUc which is public key of customer or cardholder for decryption 'D'.

Payment Authorization and Payment Capture :

Payment authorization as the name suggests is the authorization of payment information by merchant which ensures payment will be received by merchant. Payment capture is the process by which merchant receives payment which includes again generating some request blocks to gateway and payment gateway in turn issues payment to merchant.