On the Joint Security of Encryption and Signature in EMV

Jean Paul Degabriele, Anja Lehmann, Kenneth G. Paterson, Nigel P. Smart and Mario Strefler

CT-RSA 2012

29th February 2012

Outline

The EMV Standard

- EMV stands for Europay, Mastercard and VISA, and it is the **de facto global standard** for IC credit/debit cards – *Chip & PIN*.

- As of Q3 2011, there were more than 1.34 billion EMV cards in use worldwide.
- The standard specifies the inter-operation of IC cards with Point-Of-Sale terminals (POS) and Automated Teller Machines (ATM) .

The EMV Standard

- EMV stands for Europay, Mastercard and VISA, and it is the **de facto global standard** for IC credit/debit cards – *Chip & PIN*.

- As of Q3 2011, there were more than 1.34 billion EMV cards in use worldwide.
- The standard specifies the inter-operation of IC cards with Point-Of-Sale terminals (POS) and Automated Teller Machines (ATM) .

The EMV Standard

- EMV stands for Europay, Mastercard and VISA, and it is the **de facto global standard** for IC credit/debit cards – *Chip & PIN*.

- As of Q3 2011, there were more than 1.34 billion EMV cards in use worldwide.
- The standard specifies the inter-operation of IC cards with Point-Of-Sale terminals (POS) and Automated Teller Machines (ATM) .

EMV Cards

- EMV cards contain a ‘Chip’ which allows them to perform cryptographic computations.
- All EMV cards contain a **symmetric key** which they share with the Issuing Bank.
- Most cards are also equipped with **RSA keys** to compute signatures for card authentication and transaction authorization, and encrypt the PIN between the terminal and the card.

Transaction Flow

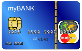
An EMV transaction progresses over three stages:

- **Card Authentication**: Static Data Authentication (SDA), Dynamic Data Authentication (DDA/CDA).
- **Cardholder Verification**: paper Signature, PIN – online/offline – cleartext/encrypted.
- **Transaction Authorization**: A successful transaction ends with the card producing a **Transaction Certificate (TC)** – a MAC computed over the transaction details.

  CDA cards additionally compute a **digital signature** over the transaction details and the TC.

Transaction Flow

An EMV transaction progresses over three stages:



Card Authentication: Static Data Authentication (SDA), Dynamic Data Authentication (DDA/CDA).

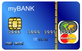
Cardholder Verification: paper Signature, PIN – online/offline – cleartext/encrypted.

Transaction Authorization: A successful transaction ends with the card producing a **Transaction Certificate (TC)** – a MAC computed over the transaction details.

CDA cards additionally compute a **digital signature** over the transaction details and the TC.

Transaction Flow

An EMV transaction progresses over three stages:



Card Authentication: Static Data Authentication (SDA), Dynamic Data Authentication (DDA/CDA).

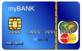
Cardholder Verification: paper Signature, PIN – online/offline – cleartext/encrypted.

Transaction Authorization: A successful transaction ends with the card producing a **Transaction Certificate (TC)** – a MAC computed over the transaction details.

CDA cards additionally compute a **digital signature** over the transaction details and the TC.

Transaction Flow

An EMV transaction progresses over three stages:



Card Authentication: Static Data Authentication (SDA), Dynamic Data Authentication (DDA/CDA).

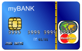
Cardholder Verification: paper Signature, PIN – online/offline – cleartext/encrypted.

Transaction Authorization: A successful transaction ends with the card producing a **Transaction Certificate (TC)** – a MAC computed over the transaction details.

CDA cards additionally compute a **digital signature** over the transaction details and the TC.

Transaction Flow

An EMV transaction progresses over three stages:



Card Authentication: Static Data Authentication (SDA), Dynamic Data Authentication (DDA/CDA).

Cardholder Verification: paper Signature, PIN – online/offline – cleartext/encrypted.

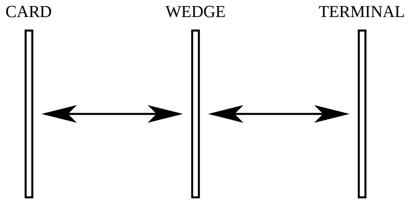
Transaction Authorization: A successful transaction ends with the card producing a **Transaction Certificate (TC)** – a MAC computed over the transaction details.

CDA cards additionally compute a **digital signature** over the transaction details and the TC.

The Cambridge Attack

Royal Holloway University of London Information Security Group

- At Oakland ’10 the following **Wedge Attack** was presented, it allows an attacker to make transactions without the card’s PIN.
- The wedge manipulates the communication between the card and the terminal so that the terminal believes PIN verification was successful, while the card thinks that a paper signature was used instead.



The Cambridge Attack

- At Oakland '10 the following **Wedge Attack** was presented, it allows an attacker to make transactions without the card's PIN.
- The wedge manipulates the communication between the card and the terminal so that the terminal believes PIN verification was successful, while the card thinks that a paper signature was used instead.
- The card's view of the cardholder verification is transmitted to the terminal in a format which it may not comprehend, and the attack can go undetected even during **online** and **CDA** transactions.
- The attack can easily be prevented, by ensuring that the terminal inspects the card's view of the cardholder verification.

Our Contribution

Royal Holloway University of London Information Security Group

- The EMV standard allows the **same RSA key-pair** to be used for both encryption and signature.
- Folklore dictates key separation, but sharing keys reduces processing and storage costs.
- No formal analysis exists that shows whether this is detrimental for the security of EMV or not.
- This is exactly the aim of our paper, we present an attack that exploits key reuse in EMV, together with positive results about upcoming versions of the standards.

A New Attack on EMV

- Our attack exploits the reuse of RSA keys in an EMV card to allow an attacker to make transactions **without** the card's PIN.
- The attack is only applicable to a **CDA** card in an **offline** transaction.
- If the countermeasure against the Cambridge attack is in place our attack would still work!
- The attack builds on Bleichenbacher's attack against RSA with PKCS#1 encoding (CRYPTO '98).

The Bleichenbacher Attack

- PKCS#1 v1.5 specified that the plaintext be encoded as:

$$m = \mathbf{00} \,||\, \mathbf{02} \,||\, \textbf{Padding String} \,||\, \mathbf{00} \,||\, \textbf{Data}$$

- Assume access to a ciphertext-validity oracle **Valid**$(\cdot)$.
- If **Valid**(c) then $2B \leq m < 3B$, where $B = 2^{8(k-2)}$.
- Using the multiplicative homomorphism of RSA, it is possible to construct a sequence of related ciphertexts such that:
  a. Each ciphertext is valid with probability one half.
  b. Each valid ciphertext found, narrows down the range by half.
- For a 1024-bit RSA modulus, roughly a **million** oracle queries are required to recover m.

The Bleichenbacher Attack

- PKCS#1 v1.5 specified that the plaintext be encoded as:

  $$m = \mathbf{00} \,||\, \mathbf{02} \,||\, \textbf{Padding String} \,||\, \mathbf{00} \,||\, \textbf{Data}$$

- Assume access to a ciphertext-validity oracle **Valid**$(\cdot)$.

- If **Valid**(c) then $2B \leq m < 3B$, where $B = 2^{8(k-2)}$.

- Using the multiplicative homomorphism of RSA, it is possible to construct a sequence of related ciphertexts such that:
  a. Each ciphertext is valid with probability one half.
  b. Each valid ciphertext found, narrows down the range by half.

- For a 1024-bit RSA modulus, roughly a **million** oracle queries are required to recover m.

The Bleichenbacher Attack

- PKCS#1 v1.5 specified that the plaintext be encoded as:

$$m = \mathbf{00} \,||\, \mathbf{02} \,||\, \mathbf{Padding\ String} \,||\, \mathbf{00} \,||\, \mathbf{Data}$$

- Assume access to a ciphertext-validity oracle $\mathbf{Valid}(\cdot)$.
- If $\mathbf{Valid}(c)$ then $2B \leq m < 3B$, where $B = 2^{8(k-2)}$.
- Using the multiplicative homomorphism of RSA, it is possible to construct a sequence of related ciphertexts such that:
  a. Each ciphertext is valid with probability one half.
  b. Each valid ciphertext found, narrows down the range by half.
- For a 1024-bit RSA modulus, roughly a **million** oracle queries are required to recover m.

PIN Encryption in EMV

- The encoding used in EMV for PIN is encryption is as follows:

  7F || PIN Block || ICC Challenge || Random Padding

  where the PIN block and the ICC Challenge are 8 bytes long.

- Upon decryption the card performs 3 checks:
  a. Is the ICC Challenge equal to the one it produced?
  b. Is the Header byte equal to '7F'?
  c. Does the PIN in the PIN Block match the one stored in the card?

- If test **b** is carried out first, and its success or failure can be distinguished **(e.g. Timing or Power Analysis)**, then a Bleichenbacher-style attack is possible.

PIN Encryption in EMV

Royal Holloway University of London Information Security Group

- The encoding used in EMV for PIN is encryption is as follows:

  7F || PIN Block || ICC Challenge || Random Padding

  where the PIN block and the ICC Challenge are 8 bytes long.

- Upon decryption the card performs 3 checks:
  a Is the ICC Challenge equal to the one it produced?
  b Is the Header byte equal to '7F'?
  c Does the PIN in the PIN Block match the one stored in the card?

- If test **b** is carried out first, and its success or failure can be distinguished **(e.g. Timing or Power Analysis)**, then a Bleichenbacher-style attack is possible.

PIN Encryption in EMV

- The encoding used in EMV for PIN is encryption is as follows:

  7F || PIN Block || ICC Challenge || Random Padding

  where the PIN block and the ICC Challenge are 8 bytes long.

- Upon decryption the card performs 3 checks:
  - a Is the ICC Challenge equal to the one it produced?
  - b Is the Header byte equal to '7F'?
  - c Does the PIN in the PIN Block match the one stored in the card?

- If test **b** is carried out first, and its success or failure can be distinguished **(e.g. Timing or Power Analysis)**, then a Bleichenbacher-style attack is possible.

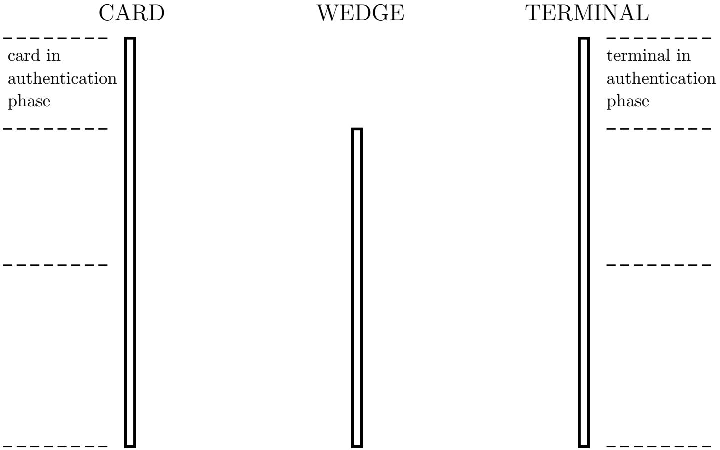
Bleichenbacher's Attack in EMV

Royal Holloway University of London Information Security Group

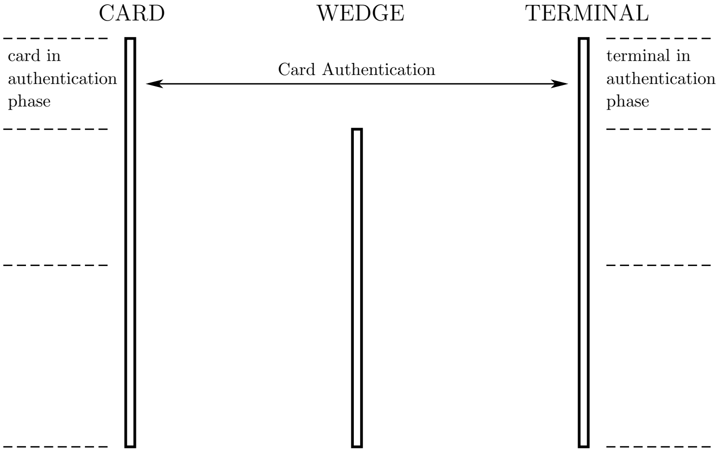
- View Bleichenbacher's attack as a black box, which when given a valid ciphertext c and access to a ciphertext-validity oracle recovers the underlying (encoded) message m.
- Alternatively we can view m as the signature of some message whose **encoding** is c, since $m = c^d \bmod N$.
- Thus when a single key pair is used, Bleichenbacher's attack allows us to sign messages whose encodings happen to be also valid ciphertexts.
- In order to sign an arbitrary encoded message μ, we blind it with an integer ρ such that $\rho^e\mu$ is a valid ciphertext.

$$\text{Signature} = \rho^{-1} m \bmod N$$

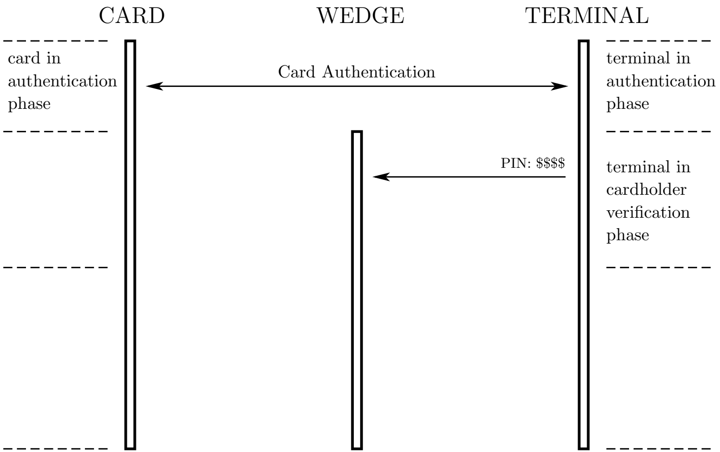
The Attack on a CDA Transaction



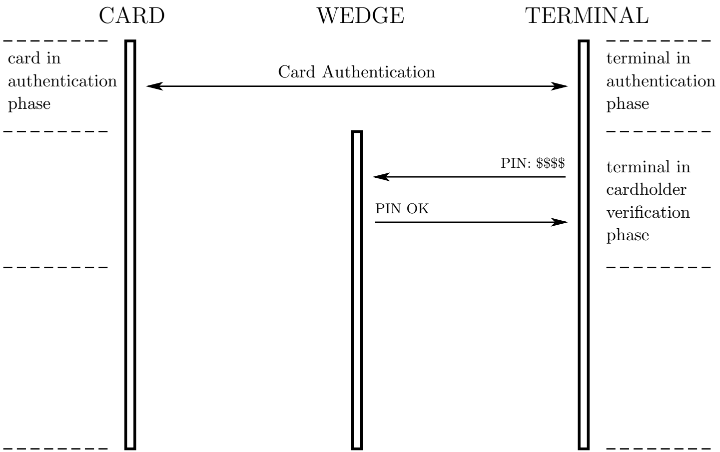
The Attack on a CDA Transaction



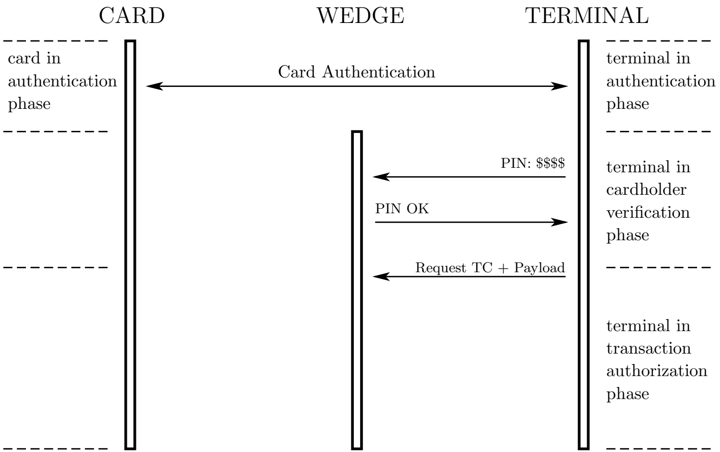
The Attack on a CDA Transaction



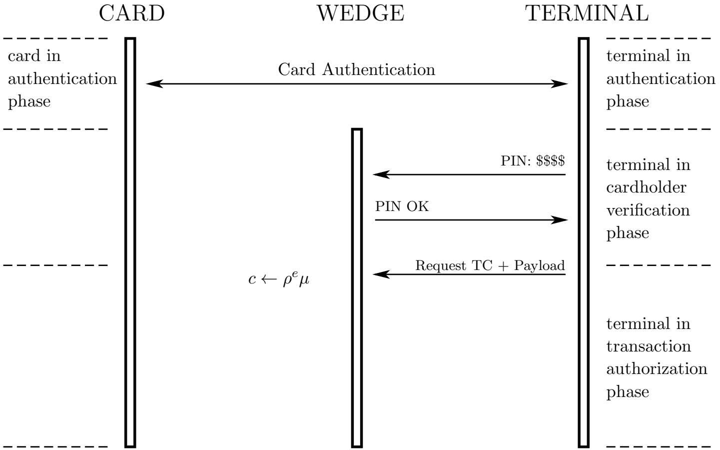
The Attack on a CDA Transaction



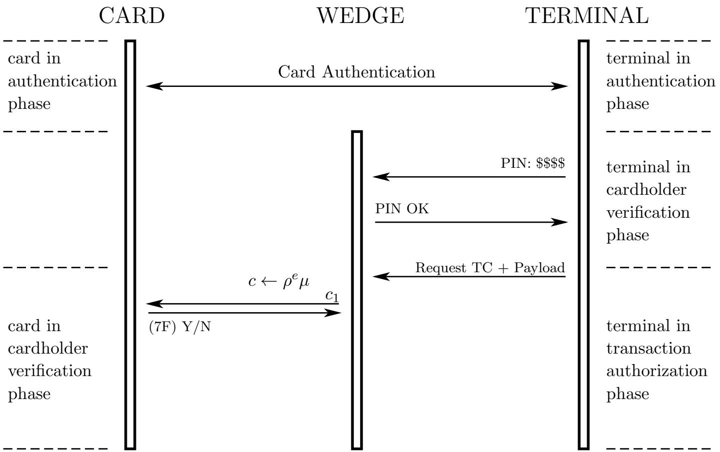
The Attack on a CDA Transaction



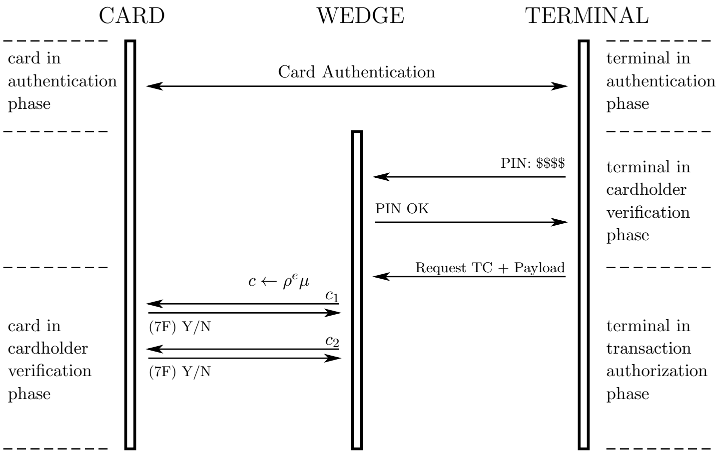
The Attack on a CDA Transaction



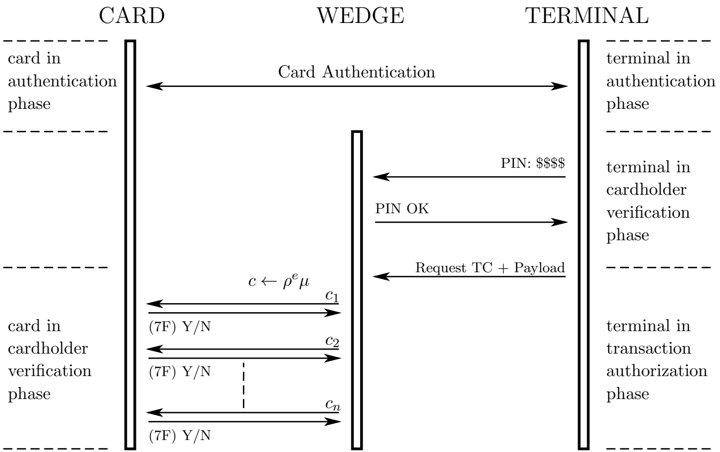
The Attack on a CDA Transaction



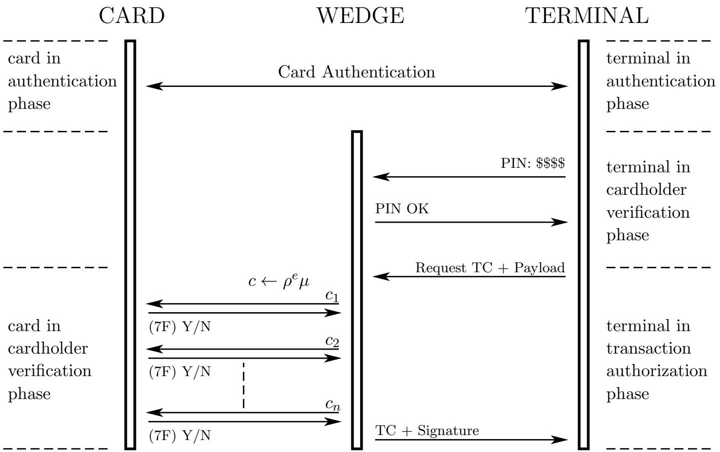
The Attack on a CDA Transaction



The Attack on a CDA Transaction



The Attack on a CDA Transaction



Practical Considerations

- We stress that we did not implement the attack in practice.
- Because the header is only 1 byte long, for a 1024-bit RSA modulus we need roughly **2000** queries to forge a signature.
- EMV cards may maintain both a **PIN try** counter and a **decryption failure** counter. Our attack would not affect the PIN try counter. In the EMV CPA specification the latter is specified to be a 2-byte counter.
- Other factors such as transaction time-outs and the inability to reproduce the '7F' oracle may limit the practicality of our attack.

On the Positive Side

- EMV Co is considering to adopt elliptic curve based algorithms in future versions of the EMV standards.
- More specifically, to use:
  - ECIES (ISO/IEC 18033-2) for PIN encryption.
  - EC-DSA or EC-Schnorr (ISO/IEC 14888-3:2006) to compute digital signatures.
- We show that the two resulting configurations are **jointly secure**, meaning that the security of the individual constituent schemes still holds when they share the same key pair.

Joint Security

- We define a **combined scheme**:

$$(\mathsf{KGen}, \mathsf{Sign}, \mathsf{Verify}, \mathsf{KEM.Enc}, \mathsf{KEM.Dec})$$

- EUF-CMA security is augmented by giving the adversary additional access to a decapsulation oracle.
- Similarly IND-CCA security is extended by giving the adversary additional access to a signing oracle.
- A combined scheme is jointly secure if it is **both** EUF-CMA secure in the presence of a decapsulation oracle, and IND-CCA secure in the presence of a signing oracle.

ECIES + EC-Schnorr

Royal Holloway University of London Information Security Group

In the Random Oracle Model:

Result	Scheme	Security	Assumptions
1	ECIES-KEM	IND-gCCA	gap-DH
2	EC-Schnorr	EUF-CMA	DLP
New	Combined Scheme	Joint Security	gap-DH, gap-DLP

[1] Abdalla, Bellare and Rogaway. *CT-RSA 2001*

[2] Pointcheval and Stern. *J. Cryptology 2000*

ECIES + EC-DSA

Assuming the group is ideal (Generic Group Model):

Result	Scheme	Security	Assumptions
3	ECIES-KEM	IND-CCA	DDH, KDF†
4	EC-DSA	EUF-CMA	$f_{conv}{}^{\ddagger}$, Hash†§
New	Combined Scheme	Joint Security	DDH, $f_{conv}{}^{\ddagger}$, Hash†§

[3] Smart. *Coding and Cryptography 2001*

[4] Brown. *Advances in Elliptic Curve Cryptography 2005*

†Uniform
‡Almost Invertible
§Collision Resistant and Zero-Finder Resistant

Conclusions

- Our attack illustrates the problems in reusing the same key-pair for encryption and signature in the current EMV standards.
- We show that the security of the individual EC-based schemes extends to the **joint setting** under the same assumptions.
- Thus for the elliptic curve based schemes under consideration, one can ‘reuse keys’ and gain substantial efficiency benefits while retaining a similar security margin.