*i*KP: *i*-Key-Protocol

Christopher Hsu

What is *i*KP?

- ***i*-Key-Protocol, *i* = 1, 2, 3, …**
- **Family of protocols**
- **Secure electronic payment**
- **Based on credit card payments**
- **Can be extended to debit card and check payments**

History of *i*KP

- **1995, IBM Research Labs Zurich and Watson Research Centre**
- **Open industry standard**
- **Incorporated into SEPP, SET**
- **Z*i*P: fully operational prototype**
- **Did not become commercial product**
- **But has been deployed in some businesses**

Initial Assumptions

- **Only partial privacy is emphasized**
- **Encryption not used in protocol**
- **Can be implemented by other means**
- **Or protocol can be extended**
- ***i*KP emphasizes the payment**
- **Assumes purchase order is already known**

Parties and Attackers

- **Three parties: Acquirer (A), Merchant (M), Customer (C)**
- **Three attackers: eavesdropper, active attacker, insider**

Acquirer Requirements

- **A1. Proof of transaction authorization of customer**
- **A2. Proof of transaction authorization by merchant**

Merchant Requirements

- **M1. Proof of transaction authorization by acquirer**
- **M2. Proof of transaction authorization by customer**

3/12/2004 7

Customer Requirements

- **C1. Unauthorized payment is impossible**
- **C2. Proof of transaction authorization by acquirer**
- **C3. Certification and authentication of merchant**
- **C4. Receipt from merchant**

3/12/2004 8

Extra Customer Requirements

- **C5. Privacy**
- **C6. Anonymity**

3/12/2004 9

Components of the Protocol

- **Keys: PK_X, SK_X, $CERT_X$**
- **Cryptography: H(.), E_X(.), S_X(.)**
- **Quantities: $SALT_C$, PRICE, DATE, $NONCE_M$, ID_M, TID_M, DESC, CAN, R_C, CID, Y/N, PIN, V**
- **Discuss 1KP in detail, comparison with 2KP and 3KP**

3/12/2004 10

1KP Protocol Flow

Initiate: C→M	$SALT_C$, CID
Invoice: C←M	Clear
Payment: C→M	EncSlip
Auth-Request: M→A	Clear, H(DESC,$SALT_C$), EncSlip
Auth-Response: M←A	Y/N, Sig_A
Confirm: C←M	Y/N, Sig_A

Common: PRICE, ID_M, TID_M, DATE, $NONCE_M$, CID, H(DESC,$SALT_C$)
Clear: ID_M, TID_M, DATE, $NONCE_M$, H(Common)
SLIP: PRICE, H(Common), CAN, R_C, [PIN]
EncSlip: E_A(SLIP)
Sig_A: S_A(Y/N, H(Common))

3/12/2004 11

Removing Nonce and Date

Initiate: C→M	$SALT_C$, CID
Invoice: C←M	Clear
Payment: C→M	EncSlip*
Auth-Request: M→A	Clear, H(DESC,$SALT_C$), EncSlip*
Auth-Response: M←A	Y/N, Sig_A
Confirm: C←M	Y/N, Sig_A

Common: PRICE, ID_M, TID_M, DATE, $NONCE_M$, CID, H(DESC,$SALT_C$)
Clear: ID_M, TID_M, DATE, $NONCE_M$, H(Common)
SLIP: PRICE, H(Common), CAN, R_C, [PIN]
EncSlip: E_A(SLIP)
Sig_A: S_A(Y/N, H(Common))

3/12/2004 12

Removing $SALT_C$

Initiate: C→M	$SALT_C$, CID
Invoice: C←M	Clear
Payment: C→M	EncSlip
Auth-Request: M→A	Clear, H(DESC,$SALT_C$), EncSlip
Auth-Response: M←A	Y/N, Sig_A
Confirm: C←M	Y/N, Sig_A

Common: PRICE, ID_M, TID_M, DATE, $NONCE_M$, CID, H(DESC,$SALT_C$)
Clear: ID_M, TID_M, DATE, $NONCE_M$, H(Common)
SLIP: PRICE, H(Common), CAN, R_C, [PIN]
EncSlip: E_A(SLIP)
Sig_A: S_A(Y/N, H(Common))

Removing $SALT_C$: Invariant

```
-- Intruder never knows desc and salt
  from same customer
invariant "Intruder does not know DESC"
  forall i: IntruderId do
    forall j: CustomerId do
    !int[i].descs[j] |
    !int[i].salts[j]
  end
end;
```


What is not fulfilled?

- Acquirer's proof of transaction authorization by merchant
- Merchant's proof of transaction authorization by customer
- Customer's certification and authentication of merchant
- Customer's receipt from merchant

2KP and 3KP

- Satisfies deficiencies of 1KP
- Differs in the number of public keys available
- Guarantees more undeniable receipts for the transaction

2KP Protocol Flow

Initiate: C→M	$SALT_C$, CID
Invoice: C←M	Clear, $\underline{Sig_M}$, $\underline{CERT_M}$
Payment: C→M	EncSlip
Auth-Request: M→A	Clear, H(DESC,$SALT_C$), EncSlip, $\underline{Sig_M}$, $\underline{CERT_M}$
Auth-Response: M←A	Y/N, Sig_A
Confirm: C←M	Y/N, $\underline{V}$, Sig_A

Common: PRICE, ID_M, TID_M, DATE, $NONCE_M$, CID, H(DESC,$SALT_C$), $\underline{H(V)}$
Clear: ID_M, TID_M, DATE, $NONCE_M$, $\underline{H(V)}$, H(Common)
SLIP: PRICE, H(Common), CAN, R_C
EncSlip: E_A(SLIP)
Sig_A: S_A(Y/N, H(Common))
$\underline{Sig_M}$: S_M(H(Common), H(V))

2KP Additions

- Merchant has public/private key and certificate
- Acquirer has certification and authentication of merchant
- Customer has certification and authentication of merchant
- Customer has receipt from merchant

3KP Protocol Flow

Initiate: C→M	$SALT_C$, CID, **$CERT_C$**
Invoice: C←M	Clear, Sig_M
Payment: C→M	EncSlip, **Sig_C**
Auth-Request: M→A	Clear, H(DESC,$SALT_C$), EncSlip, Sig_M, **Sig_C**
Auth-Response: M←A	Y/N, Sig_A
Confirm: C←M	Y/N, V, Sig_A

Common: PRICE, ID_M, TID_M, DATE, $NONCE_M$, CID, H(DESC,$SALT_C$), H(V)
Clear: ID_M, TID_M, DATE, $NONCE_M$, H(V), H(Common)
SLIP: PRICE, H(Common), CAN, R_C
EncSlip: E_A(SLIP)
Sig_A: S_A(Y/N, H(Common))
Sig_M: S_M(H(Common), H(V))
Sig_C: S_C(EncSlip, H(Common))

3/12/2004 19

3KP Additions

- **Customer has public/private key and certificate**
- **Merchant has proof of transaction authorization by customer**

3/12/2004 20

*i*KP Comparison Overview

Requirements / Protocols	1KP	2KP	3KP
A1. Proof of Transaction Authorization by Customer	*	*	**
A2. Proof of Transaction Authorization by Merchant		**	**
M1. Proof of Transaction Authorization by Acquirer	**	**	**
M2. Proof of Transaction Authorization by Customer			**
C1. Unauthorized Payment is Impossible	*	*	**
C2. Proof of transaction Authorization by Acquirer	*	*	**
C3. Certification and Authentication of Merchant		**	**
C4. Receipt from Merchant		**	**

3/12/2004 21

*i*KP Summary

- **1KP: simple, merchant is not authenticated, order and receipt are deniable**
- **2KP: merchant is authenticated**
- **3KP: non-repudiation for all messages**
- **Intended for gradual deployment**

3/12/2004 22

Bibliography

1. Bellare, Mihir *et al.* Design, Implementation and Deployment of the *i*KP Secure Electronic Payment System. IEEE Journal of Selected Areas in Communications, VOL 18, NO. 4, April 2000.
2. Bellare, Mihir *et al.* *i*KP – A Family of Secure Electronic Payment Protocols. 1995.

3/12/2004 23