♦ Design and Correctness Proof of a Security Protocol for Mobile Banking

Dalton Li, David Lin, Grace Zhao, and Brian Huang

A strong security protocol is the cornerstone for the implementation of mobile banking services and is used to determine the security properties of the system. This paper proposes an application layer security protocol for mobile banking services—the mobile banking (MB) protocol—based on requirements from mobile banking systems. Our research provides an in-depth analysis of the design technologies used in the MB protocol, as well as a correctness proof of its security properties based on the strand space model. © 2009 Alcatel-Lucent.

Introduction

Most security for mobile banking (MB) systems today relies primarily on a transport layer security protocol known as Wireless Transport Layer Security (WTLS), which evolved from the Transport Layer Security (TLS) version 1.0 standard [1]. Prior to that, the Secure Electronic Transaction (SET) [11] protocol was also used, though SET in particular failed to win popular support due to its cost, complexity, and limited capabilities for mobile applications. Both mobile banking security protocols share the common drawback of being overly complicated with restricted performance requirements that are difficult to adapt for mobile handsets or devices. In addition, they both lack the flexibility and application layer capabilities necessary to fulfill security requirements for mobile banking systems.

Per our recent survey, no dedicated application layer security protocol has ever been defined or attempted to fill this security gap. In the same vein, we discovered there is a definite need to support certain security measures in the application layer in conjunction with the use of security capabilities in the lower layers of any protocol stack. Our observations for security capabilities in the application layer are as follows:

- The application layer protocol for mobile banking should guarantee the security of all transactions, i.e., secrecy, correspondence, integrity, and nonrepudiation. For example, the security protocol should have the ability to resist malicious attacks.
- For mobile banking applications especially, message exchange should be kept to a minimum, e.g., use only two messages, request and response, to complete a transaction.

In the following sections, we will provide more detailed descriptions of this MB protocol, along with a correctness proof based on the strand space model to certify its usefulness to this protocol.

Basic Message Exchange Scenario for Mobile Banking

A typical message exchange for a mobile banking transaction entails two major steps, the secure request procedure sent from the mobile customer and the secure response procedure received from the banking service.

Bell Labs Technical Journal 14(1), 259–266 (2009) © 2009 Alcatel-Lucent. Published by Wiley Periodicals, Inc.
Published online in Wiley InterScience (www.interscience.wiley.com) • DOI: 10.1002/bltj.20366

1. *Secure Request.* The secure transaction request includes customer authentication information to be used by the banking service to verify the customer's authenticity.
2. *Secure Response.* After authenticating/verifying the identity of the customer, the service returns a secure transaction response back to the customer.

The preceding steps only encompass the secure transfer procedure for a secure transaction. The handshake of exchanging security keys is predetermined between the two entities beforehand. Hence, the generation of private/public keys and the handshake procedure is outside the scope of this study.

MB Protocol

We begin our discussion of the MB protocol by postulating use assumptions and describing the protocol mechanisms in detail. **Panel 2** provides an explanation of the symbols used.

Assumptions

When a customer subscribes to a mobile banking service, he will create a pair of public/private keys, which are generated through some automated mechanism. The customer will then register his name and corresponding public key in the mobile banking system. At the same time, he will save the bank public key to his mobile device. So we have the following valid assumptions:

1. Only the customer knows his private key, K_C^{-1}
2. The customer also knows the public key of the bank, K_B
3. Only the bank knows its private key, K_B^{-1}
4. The bank also knows the public key of the customer, K_C

The MB Protocol Description

The mobile device saves the customer's private key, K_C^{-1}, and the bank's public key, K_B. When the customer initiates a transaction, the mobile device launches following series of steps:

1. Generate two new symmetric session keys, k_1 and k_2.
2. Create a transaction request message body. The message body presents the transaction indication, such as remittance or payment.

Panel 1. Abbreviations, Acronyms, and Terms

IP—Internet Protocol
IMS—IP Multimedia Subsystem
MB—Mobile Banking Protocol
PD-FE—Policy decision function entity
SET—Secure Electronic Transaction
TLS—Transport Layer Security
WTLS—Wireless Transport Layer Security

3. Build a message digest consisting of the customer name, the bank name, the request message body, and k_2 : $h(C, B, req, k_2)$.
4. Use the customer's private key, K_C^{-1}, to sign the result of step 3 to obtain the customer's digital signature: $\{h(C, B, req, k_2)\}_{K_C^{-1}}$.
5. Concatenate the request message body and the result of step 4. Then, use k_1 to encrypt it: $\{req, \{h(C, B, req, k_2)\}_{K^{-1}{}_C}\}_{k_1}$.
6. Use K_B to encrypt k_1 and k_2 to construct a digital envelope $\{k_1, k_2\}_{K_B}$.
7. Concatenate the result of step 5 and step 6 to build the complete message: $\{\{req, \{h(C, B, req, k_2)\}_{K_C^{-1}}\}_{k_1},$

Panel 2. Explanation of Symbols

C(Customer): Customer identifier
B(Bank): Bank identifier
T: Set of text representing the atomic messages
K: Set of cryptographic keys disjoint from T
k_1: Newly generated session key, k_1
k_2: Newly generated session key, k_2
K_C^{-1}: Private key of the customer used for signature
K_B^{-1}: Private key of the bank used for signature
K_C: Public key of the customer
K_B: Public key of the bank
K_P: A key set containing the keys initially known to a penetrator
$\{m\}_k$: Message m encrypted with k
$\{m\}_{k^{-1}}$: Message m signed with private key k^{-1}
{m1, m2}: The concatenation of message m1 and m2
h(m): The message digest of m
req: Transaction request message body
ans: Transaction answer, either successful or failed

$\{k_1, k_2\}_{K_B}$ and send it to the mobile banking system.

When the mobile banking system receives the request message, it will perform the following steps:

1. Use the private key of the bank, K_B^{-1}, to decrypt the digital envelope $\{k_1, k_2\}_{K_B}$ to get the session keys: k_1 and k_2.
2. Use k_1 to decrypt $\{req, \{h(C, B, req, k_2)\}_{K_C^{-1}}\}_{k_1}$ to get the request message body.
3. Create a message digest containing the customer name, the bank name, the request message body, and k_2: $h(C, B, req, k_2)$.
4. Use the public key of customer, K_C, to decrypt $\{h(C, B, req, k_2)\}_{K_C^{-1}}$ to get the message digest, and verify whether the digest equates to the result from step 3. If the verification is valid, that means the message has not been compromised, and the system continues with step 5. Otherwise, the message is dropped.
5. Handle the transaction request and generate the answer message body.
6. Create the message digest of the bank name, the customer name, and the answer message body : $h(B, C, ans)$.
7. Use the bank's private key, K_B^{-1}, to sign the result of step 6 to obtain the bank's digital signature: $\{h(B, C, ans)\}_{K_B^{-1}}$.
8. Use k_2 to encrypt the answer message body, k_1, and the result of step 7 to build a complete message: $\{ans, k_1, \{h(B, C, ans)\}_{K_B^{-1}}\}_{k_2}$ and send it to the customer.

When the customer device receives the answer message, it will perform the remaining steps:

1. Use k_2 to decrypt the answer message to get the answer message body.
2. Create the message digest of the bank name, the customer name, and the answer message body: $h(B, C, ans)$.
3. Use the public key of the bank, K_B, to decrypt $\{h(B, C, ans)\}_{K_B^{-1}}$, and verify whether the digest equates to the result from step 2. If the verification is valid, that means the message has not been compromised, and the mobile device can continue processing the answer message. Otherwise the transaction fails, and the message should be dropped.

As a summary, the formal description of the MB protocol is as follows:

$$C \rightarrow B: \{req, \{h(C, B, req, k_2)\}_{K_C^{-1}}\}_{k_1}, \{k_1, k_2\}_{K_B}$$

$$B \rightarrow C: \{ans, k_1, \{h(B, C, ans)\}_{K_B^{-1}}\}_{k_2}$$

The Essential Design Techniques

Essential design techniques and related procedures for the MB protocol are detailed in the following.

Digital Envelope

The digital envelope is a security method which uses the public key of the receiver to encrypt messages. Since only the receiver knows the corresponding private key, he is the only party who can successfully decrypt the message, i.e., "open the envelope." In MB protocol, $\{k_1, k_2\}_{K_B}$ is a digital envelope. The customer encrypts two newly generated keys with the public key of the bank. Thus, only the bank can open this envelope and obtain these two keys.

Symmetric Approach Within Asymmetric Structure

This security proposal embeds a symmetric encryption and decryption approach using $\{k_1, k_2\}$ keys within the asymmetric algorithm. Since only the bank service has the capability to decrypt the pair, it is considerably safer to transmit this type of information as part of the message sent to the bank service. As a $\{k_1, k_2\}$ pair is newly generated for each transaction, this is another security measure to prevent Replay attacks.

Analysis of MB Protocol Based on Strand Space

Building on strand space model theory [2–10], the correctness of the MB protocol can be considered with respect to the aspects of correspondence and secrecy [10]:

1. *Correspondence* means that each time a principle B completes a run of the protocol as a responder using x—which to B appears to be a run with A—there is a unique run of the protocol with the principal A as initiator using x, which to A appears to be a run with B.
2. *Secrecy* means that messages protected by the protocol cannot be known by any penetrator.

Next, we will use the strand space model to prove these two properties of the MB protocol. One item to

note is that the strand space model theory defines the eight types of generalized attack behaviors currently known.

Definition 1: MB Strand Spaces

An infiltrated strand space $(\sum, P)$ is an MB space if $\sum$ is the union of three kinds of strands:

1. *Penetrator strands,* p ∈ P. The definition of penetrator strands can be found in [9] and in section 3.1 of [10].
2. *Customer strands,* s ∈ Customer[B,C, k_1, k_2, req, ans] with trace:

$$\langle + \{\{req, \{h(C, B, req, k_2)\}_{K_C^{-1}}\}_{k_1}, \{k_1, k_2\}_{K_B}\}, - \{ans, k_1, \{h(B, C, ans)\}_{K_B^{-1}}\}_{k_2}\rangle$$

   where C,B ∈ T, k_1, k_2 ∈ K, Customer[B, C, k_1, k_2, req, ans] denotes the set of all strands with the trace shown. The principle associated with this strand is C.
3. *Bank strands*, t ∈ Bank[B, C, k_1, k_2, req, ans], with trace:

$$\langle - \{\{req, \{h(C, B, req, k_2)\}_{K_C^{-1}}\}_{k_1}, \{k_1, k_2\}_{K_B}\}, + \{ans, k_1, \{h(B, C, res)\}_{K_B^{-1}}\}_{k_2}\rangle$$

   where C, B ∈ T, k_1, k_2 ∈ K, Bank[B, C, k_1, k_2, req, ans] denotes the set of all strands with the trace shown. The principle associated with this strand is B. Given any strand s in S, we can uniquely classify it as a penetrator strand, a customer's strand, or a bank's strand just by the form of its trace.

Proposition 1: The Proof of Correspondence

If:

1. $\sum$ is an MB space, ε is a bundle [8] in $\sum$, and s is a customer strand in Customer[B, C, k_1, k_2, req, ans] with ε-height 2.
2. K_B^{-1}, K_C^{-1}, k_1, $k_2 \notin K_P$.
3. $k_1 \neq k_2$ and k_1, k_2 are uniquely originating in $\sum$.

Then ε contains a bank strand, t ∈ Bank[B,C, k_1, k_2, req, ans] with ε-height 2.

The customer's strand is depicted in **Figure 1**. We will prove proposition 1 using a sequence of lemmas.

Lemma 1. The set $V = \{n \in \varepsilon : k_1 \subset uns\_term(n) \wedge \{k_1, k_2\}_{k_B} \not\subset uns\_term(n)\}$ has a ≤-minimal node n2. The node n2 is regular, and the sign of n2 is positive.

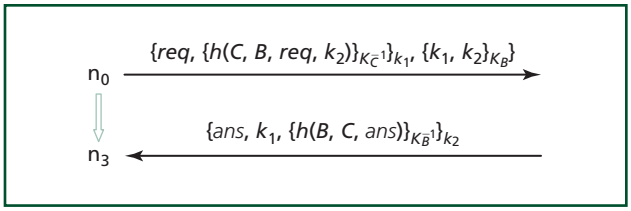


Figure 1.
Customer strand.

Proof. Because $k_1 \subset term\langle s, 1\rangle$ = term(n0), so k_1 originates on n0. From Figure 1, we know n3 ∈ ε, n3 ∈ V, V is non-empty. Hence V has at least one ≤-minimal element n2. The sign of n2 is positive.

Can n2 lie on a penetrator of strand p? Let us examine the possible cases for positive penetrator nodes, according to the form of the trace of p [10].

M. *Text message*. The trace tr(p) has the form ⟨+t⟩ where t ∈ T . But $T \cap K = \phi$, and k_1 ∈ K, so t ≠ k_1. Thus, this case is invalid.

F. *Flushing*. The trace tr(p) has the form ⟨−g⟩ and thus lacks any positive nodes.

T. *Tee*. The trace tr(p) has the form ⟨−g,+g,+g⟩, so the positive nodes are not minimal occurrences.

C. *Concatenation*. The trace tr(p) has the form ⟨−g,−h,+gh⟩, ∵ k_1 is simple, ∴ $k_1 \subset$ g or $k_1 \subset$ h. So the positive node is not a minimal occurrence.

K. *Key*. The trace tr(p) has the form ⟨+ k⟩ where k ∈ K_P. But $k_1 \notin K_P$, so this case does not apply.

E. *Encryption*. The trace tr(p) has the form ⟨−K,−h, +{h}K⟩. Suppose $k_1 \subset \{h\}_K \wedge \{k_1, k_2\}_{K_B} \not\subset \{h\}_K$, ∵ $k_1 \subset \{h\}_K$, $k_1 \neq \{h\}_K$ ∴ $k_1 \subset$ h. But $\{k_1, k_2\}_{K_B} \not\subset h$, so the positive node is not minimal.

D. *Decryption*. The trace tr(p) has the form $\langle -K^{-1}, -\{h\}_K, +h\rangle$. If $k_1 \subset h \wedge \{k_1, k_2\}_{K_B} \not\subset h$, according to the minimality of h, we can suppose $\{k_1, k_2\}_{K_B} \subset gh$. Hence (using the assumption of free encryption) h = {k_1, k_2}, K = K_B. Thus there exists a node m with $term(m) = K_B^{-1}$. Since by assumption, $K_B^{-1} \notin K_P$, we can infer that K_B^{-1} originates only on a regular node. However, no customer strand or bank strand originates K_B^{-1}.

S. *Separation into components*. The trace tr(p) has the form ⟨−gh,+g,+h⟩. Assume term(n2) = g, there is a symmetrical case if term(n2) = h. ∵ $k_1 \subset g \wedge \{k_1, k_2\}_{K_B} \not\subset g$, according to the minimality of g, we can suppose

$\{k_1, k_2\}_{K_B} \subset gh$. But $\{k_1, k_2\}_{K_B}$ is simple, so $\{k_1, k_2\}_{K_B} \subset h$. Let $U = \{m \in \varepsilon : m \prec n_2 \wedge gh \subset unsterm(m)\}$. Because term(<p, 1>) = −gh, <p, 1> ∈ U, U is non-empty. Hence U has at least one ⩽-minimal element m1.

For M, F, T, K, clearly a minimal member of U cannot lie on these strands.

S. *Separation into components*. If gh ⊂ term(m1), where m1 is a positive node on a strand p′ of kind S, then gh ⊂ term(<p′, 1>), <p′, 1><m1, contradicting the minimality of m in U.

E. *Encryption*. If gh ⊂ term(m1), where m1 is a positive node on a strand p′ of kind E, then gh ⊂ term(<p′, 2>), <p′, 2 > <m1, contradicting the minimality of m in U.

D. *Decryption*. If gh ⊂ term(m1), where m1 is a positive node on a strand p′ of kind D, then gh ⊂ term(<p′, 2>), <p′, 2><m1, contradicting the minimality of m in U.

C. *Concatenation*. Suppose gh ⊂ term(m1), m1 is a positive node on a strand p′ of kind C, then gh= term(m1), term(<p′, 1>)=g=term(n2). Hence < p′, 1><<p′,3> = m1 ≺ n2, contradicting the minimality of n2 in V.

Therefore n2 does not lie on a penetrator strand, but must lie on a regular strand instead.

Lemma 2. A node n1 precedes n2, and $\{k_1, k_2\}_{K_B} \subset term(n_1)$, as shown in **Figure 2**.

Proof. k_1 originates at n0, and originates uniquely in $\sum$. Moreover, we have $\{k_1, k_2\}_{K_B} \subset term(n_0)$ and $\{k_1, k_2\}_{K_B} \not\subset term(n_2)$, so n0 ≠ n2. Hence, k_1 does not originate at n2. So there is a node n1 preceding n2 on the same strand such that $k_1 \subset term(n_1)$. By the minimality property of n2, we can infer that $\{k_1, k_2\}_{K_B} \subset term(n_1)$.

Lemma 3. The regular strand t containing n1 and n2 is a bank strand, and is contained in ε.

Proof. Node n2 is a positive regular node and follows a node (namely, n1) of the form $\{xy\}_k$. Hence t is a bank strand; if it were a customer strand, it would contain only a negative node after one of that form. Thus, n1 and n2 are the first and second nodes of t, respectively. Since the last node of t is contained in ε, it must have ε-height of 2.

Proposition 2: The Proof of Secrecy

We may use the same methods to show that the customer's nonce k_1 remains secret in the protocol.

If:

1. $\sum$ is an MB space, ε is a bundle in $\sum$, and s is a customer strand in Customer[B,C,k_1,k_2,req,ans] with ε-height 2.
2. K_B^{-1}, K_C^{-1}, k_1, $k_2 \notin K_P$.
3. $k_1 \neq k_2$, and k_1, k_2 are uniquely originating in $\sum$.

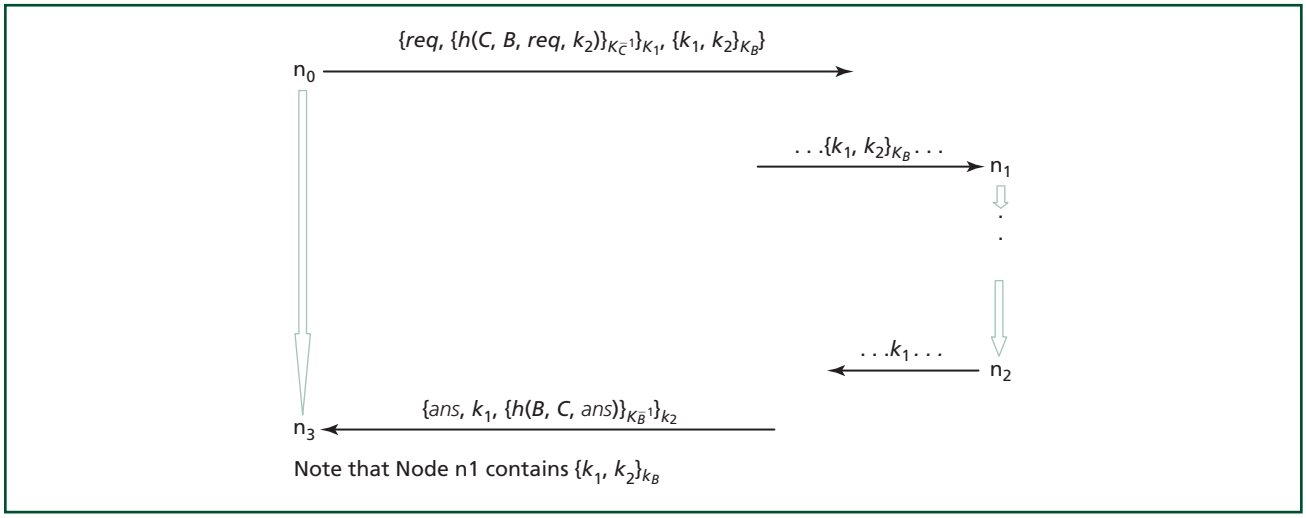


Figure 2.
Node n1 precedes n2.

Then for all nodes m $\in \varepsilon$ such that $k_1 \subset term(m)$, either $\{k_1, k_2\}_{K_B} \subset term(m)$ or $\{ans, k_1, \{h(B, C, ans)\}_{K_B^{-1}}\}_{k_2} \subset term(m)$.

Proof. Let $\{ans, k_1, \{h(B, C, ans)\}_{K^{-1}{}_B}\}_{k_2} = v_3$.

Consider the set $F = \{n \in \varepsilon : k_1 \subset term(n) \wedge \{k_1, k_2\}_{K_B} \not\subset term(n) \wedge v_3 \not\subset term(n)\}$. Suppose F is non-empty, then F has at least one $\leq$-minimal element. We show first that such nodes are not regular. We next show that they are not penetrator nodes. Therefore F is empty, and the theorem holds.

Suppose instead that m $\in$ F being minimal and a regular node, the sign of m is positive. Node m cannot lie on s: Only n0 is positive, and $\{k_1, k_2\}_{K_B} \subset term(n_0)$, so m is not in s. Moreover k_1 originates uniquely in n_0, so m cannot exist in other regular strands, $s' \neq s$. Thus m isn't a regular node.

The next proof is similar to the proof of lemma 1. The only significant difference is that when the penetrator strand is of type D, we must consider another case. In that case, $h = \{ans, k_1, \{h(B, C, ans)\}_{K_B^{-1}}\}_{k_2}$ and $K = k_2$. Thus there must be a node n with term(n) = k_2. But $k_2 \notin K_P$, so k_2 can only be sent from a regular node. However, no customer strand or bank strand originates k_2.

So we can draw a conclusion that F actually is empty and the occurrence of k_1 can only take the encrypted form prescribed by the MB protocol. That is to say k_1 remains secret in the MB protocol.

Thus the proof of secrecy of k_2 and req is similar to that of k_1.

Analysis of Computational Effort and Areas for Enhancement

To compose the request message, the customer's mobile device is only required to perform one hashing calculation and three encryption calculations. Similarly, to read from response messages, two decryptions and one hash are needed. So there are a total of five encryption/decryption operations and two hash operations needed for one transaction. We believe this is acceptable for a mobile device as this overhead can easily be solved at the hardware or software level.

In this protocol, the selection of fresh session keys, k_1 and k_2, can be strengthened by adopting a stateful authentication method by generating a new and unique key pair and keeping track of all key pairs used by the mobile unit.

Conclusion

This paper proposes a security protocol for mobile banking services, the MB protocol at the application level. We discuss the essential design technologies used in the MB protocol and provide an MB protocol correctness proof based on the strand space model.

Our study has found that the MB protocol achieves its security goals and has the ability to resist common attacks with high performance, proving itself a worthy addition to the application layer in conjunction with other security measures offered in the lower layer of the protocol stack.

Acknowledgements

The authors would like to acknowledge the contributions of the former and present members of the IP Multimedia Subsystem (IMS) security team for their contributions and suggestions to this research.

References

[1] T. Dierks and C. Allen, "The TLS Protocol, Version 1.0," IETF RFC 2246, Jan. 1999, <http://www.ietf.org/rfc/rfc2246.txt>.

[2] J. D. Guttman, "Security Protocol Design via Authentication Tests," Proc. 15th IEEE Comput. Security Foundations Workshop (CSFW '02) (Cape Breton, N.S., Can., 2002), pp. 92–103.

[3] J. D. Guttman, F. J. Thayer, J. A. Carlson, J. C. Herzog, J. D. Ramsdell, and B. T. Sniffen, "Trust Management in Strand Spaces: A Rely-Guarantee Method," Proc. 13th Eur. Symposium on Programming (ESOP '04) (Barcelona, Sp., 2004), published in Programming Languages and Systems, Lecture Notes in Comput. Sci. (LNCS 2986) (D. Schmidt, ed.), Springer, Berlin, Heidelberg, New York, 2004, pp. 325–339.

[4] J. D. Guttman and F. J. Thayer Fábrega, "Authentication Tests and the Structure of Bundles," Theoret. Comput. Sci., 283:2 (2002), 333–380.

[5] S. Lukell and A. C. M. Hutchison, "Attack Analysis of Cryptographic Protocols Using Strand Spaces," South African Comput. J., 31 (2003), 25–32.

[6] A. Mukhamedov, S. Kremer, and E. Ritter, "Analysis of a Multi-Party Fair Exchange

Protocol and Formal Proof of Correctness in the Strand Space Model," Proc. 9th Internat. Conf. on Financial Cryptography and Data Security (FC '05) (Roseau, Dominica, 2005), published in Lecture Notes in Comput. Sci. (LNCS 3570) (A. S. Patrick and M. Yung, eds.), Springer, Berlin, Heidelberg, New York, 2005, pp. 255–269.

[7] F. J. Thayer Fábrega, J. C. Herzog, and J. D. Guttman, "Honest Ideals on Strand Spaces," Proc. 11th IEEE Comput. Security Foundations Workshop (CSFW '98) (Rockport, MA, 1998), pp. 66–77.

[8] F. J. Thayer Fábrega, J. C. Herzog, and J. D. Guttman, "Strand Space Pictures," Proc. Workshop on Formal Methods and Security Protocols (Co-located with LICS '98) (Indianapolis, IN, 1998).

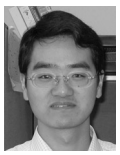
[9] F. J. Thayer Fábrega, J. C. Herzog, and J. D. Guttman, "Strand Space: Why Is a Security Protocol Correct?," Proc. IEEE Symposium on Security and Privacy (SP '98) (Oakland, CA, 1998), pp. 160–171.

[10] F. J. Thayer Fábrega, J. C. Herzog, and J. D. Guttman, "Strand Spaces: Proving Security Protocols Correct," J. Comput. Security, 7:2/3 (1999), 191–230.

[11] Visa and MasterCard, SET Secure Electronic Transaction Specification, Books 1-3: Business Description, Programmer's Guide, Formal Protocol Specification, York University, Ontario, Can., 1996.

(Manuscript approved December 2008)

DALTON LI is a member of technical staff in the IMS



Research and Development department at Alcatel-Lucent in Qingdao, China. He received a master's degree in computer science from Hohai University in China. He is currently working on IMS quality of service (QoS)-related features. His professional interests include IMS QoS and security.

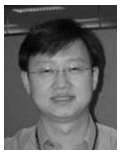
DAVID LIN is a member of technical staff in the IMS

Research and Development department at Alcatel-Lucent in Qingdao, China. He received a master's degree in computer applications from Ocean University of China. His professional interests are related to IMS QoS/PD-FE.

GRACE ZHAO is a technical manager in the IMS

Research and Development department at Alcatel-Lucent in Qingdao, China. She received a master's degree in software engineering from XI'an Jiaotong University, China. Her professional interests are IMS QoS/PD-FE and IMS solutions.

BRIAN HUANG is the head of the IMS Research and



Development department at Alcatel-Lucent in Qingdao, China. He received a master's degree in mechanics software development from Northwestern Polytechnical University, China. His professional interests are IMS next-generation network (NGN) solutions. ◆