International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 6 (2017) pp. 1069-1074

Efficient and Secure Identity-Based Onion Routing

Junbeom Hur* and Dong Kun Noh**

** Department of Computer Science and Engineering, Korea University*
145 Anam-ro, Seongbuk-gu, Seoul 02841, Republic of Korea

*** Department of Smart Systems Software, Soongsil University*
369 Sangdo-ro, Dongjak-gu, Seoul 156-743, Korea

****ORCID : 0000-0003-2068-633X & Scopus Author ID: 25927459300***

Abstract

Onion routing protocols achieve low-latency anonymous communication on public networks. Up to date, many onion routing protocols have been proposed, such as Tor network, in order to implement the anonymous network connection in the public networks. Although the previous schemes' multi-pass cryptographic circuit construction appears satisfactory, their circuit construction protocols have some drawbacks with regard to the efficiency and security. This paper presents a novel identity-based onion routing protocol that allows users to establish anonymous channels over a public network. The proposed scheme eliminates iterative and interactive symmetric key agreement procedures between users and onion routers by embedding a circuit construction into the non-interactive message delivery process. It significantly improves the storage and communication costs required from each user and onion router compared to the previous onion routing protocols, while requiring comparable computation cost to them.

Keywords: onion-routing, anonymity, circuit construction, encryption, complexity analysis

ACM Classification: C.2.2 Network Protocols, D.4.6 Security and Protection

INTRODUCTION

As we move to ubiquitous network environment, it has become apparent that our privacy is at stake. These privacy concerns were recognized since the beginning of the Internet age, and anonumous communication was conceived as a possible approach to their solutions. Anonymity is the user's ability to hide not only his identity but also his network information, such as his network address. This is of utter importance in many real life applications, where a user's identity should be decoupled from his network activities, for example, voting, e-cash, anonymous credentials, and so forth.

Goldschlag et al. (1996) introduced the so-called onion routing approach which is based on Chaum's notion of a anonymous channel (1981). Onion routing is an infrastructure for private communication over a public network (1998). It is an efficient mechanism to achieve a one-way anonymous channel from anonymous users to non-anonymous service providers over a public network such as the Internet. Onion routing is a type of anonymous communication that creates cryptographic circuits along an unpredictable route through a network of nodes called onion routers, and passes traffic bidirectionally along those circuits with minimal latency (Dingledin, 2004),(1996),(1998).

An onion routing is defined by a set of users, a set of nodes called onion routers that relay traffic, and a service provider. A user constructs a circuit choosing a small ordered subset of the onion routers, where the chosen nodes route the user's traffic over the path formed. In the onion routing protocol, a user who wishes to send a message wraps the message with several layers of encryption (called an onion), one for each of randomly selected onion routers in the circuit, and sends it through a sequence of them. A user includes the identifier of the next node and a random symmetric session key in each onion router, and uses routers' public keys to encrypt their respective layers. When an onion router receives a message, it decrypts the message using its private key and obtains (1) the name of the next router in the circuit and (2) another ciphertext. Then it forwards the ciphertext to the next router, and uses the random symmetric session key for the rest of the session. Anonymity derives from the fact that the order in which the onion routers are selected is random and that each router should know nothing more than its two adjacent nodes in the sequence.

The following Fig.1 shows the anonymous communication process in the onion routing protocol between the user and the service provider.

***corresponding author*

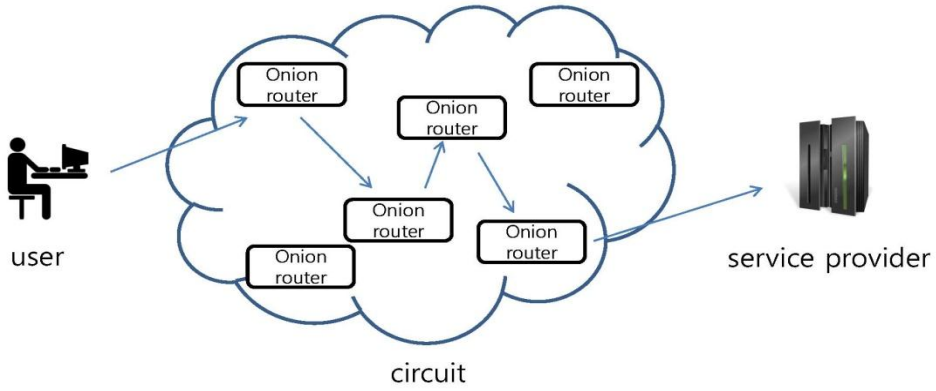

Figure 1: Onion routing protocol

Related Work :

However, this single-pass circuit construction is not forward secure; if an adversary corrupts an onion router and obtains its private key, then the adversary can derypt all of its past communication. The adversary could then successively compromise all the routers in a circuit to break the anonymity of a user's past communications. Although changing the public key and private key pairs for all onion routers after a predefined interval is a possible solution, it is not scalable. Every system user has to download a new set of public keys for all the onion routers at the start of every interval.

In Tor (2011), which is one of the largest onion routing systems to date, the circuit is constructed using the interactive and incremental telescoping technique in which a user establishes secure channels with the onion routers using a Diffie-Hellman (DH) key agreement. More precisely, the technique relies on using the routers' public keys to establish a temporary session key via an interactive DH key agreement. In Tor authentication protocol (TAP), which is used to negotiage the session keys in this multi-pass circuit construction, an onion router's public key is only used to initiate the construction and its compromise does not void the security of the session keys once the randomness used in the protocol is erased. Establishing a circuit of length l requires $O(l^2)$ network communications and $O(l^2)$ symmetric encryptions/decryptions in Tor. Overlier and Syverson (Overlier, 2007) improved the efficiency of Tor using a half-certified Diffie-Hellman (DH) key agreement, but the round complexity of telescoping is still quadratic.

Kate et al. (Kate, 2009) proposed a paring-based onion routing (PB-OR) protocol that builds a circuit with a single pass using identity-based anonymous key agreement. In identity-based cryptography, the parties' public keys are their identities, and the secret keys are provided to them by a trusted key generation center (KGC). PB-OR uses the original onion routing iea to encrypt messages using the public key of the routers, except that in this case the routers' public keys are their identities together with the validity period. Therefore, a router's corruption reveals only the messages encrypted during the particular period of the corruption. The circuit construction is non-interactive and requires O(l) messages to be exchanged. Based on this scheme, Catalano et al. (Catalano, 2009) proposed another circuit construction scheme using non-interactive certificate-less anonymous key agreement. This scheme implicitly involves identity-based key setting, and reduces the computational cost by replacing pairing operations with public cryptography operations. Certificateless encryption is a hybrid setting that lies between public key and identity-based cryptography: each user has an identity string ID with a matching secret key produced by the KGC and also a public/secret key pair, as in the traditional public key model but with the advantage that such key needs not be certified. Certificateless encryption does not suffer the problem of key escrow as the KGC cannot decrypt the message sent to a user. The CL-OR protocol in (Catalano, 2009) achieves eventual forward secrecy by having the routers periodically change their public keys. Compared to PB-OR, CL-OR requires the users to interact with the service provider at each time period to obtain the routers' new public keys (but with the advantage of not having to manage and verify certificates).

Johnson and Syverson (Johnson, 2009) proposed a model of trust in network nodes and use it to design path-selection strategies that minimize the probability that the adversary can successfully control the entrance to and exit from the network. This minimizes the chance that the adversary can observe and correlate patterns in the data flowing over the path and thereby deanonymize the user. Using the trust information, they improved the anonymity provided by onion routing networks. In another literature, Mauw et al. (2004) and Feigenbaum et al. (Feigenbaum, 2007) proposed formal models for rigorous analysis of anonymity in onion routing protocols.

Motivation :

A common realization of the previous onion routing protocols is that the onion routers are randomly chosen according to a given strategy and the user anonymously establishes with each of them a symmetric session key, which will be used to encrypt the layers of future onions. We noticed that the symmetric key establishment procedures result in efficiency and security problems.

First, building a circuit of length l requires at least additional $O(l)$ messages to be exchanged for symmetric keys establishment; and requires each user to store $O(l)$ pseudonyms and symmetric keys, and each onion router to store at most $O(n)$

pseudonyms and symmetric keys, where *n* is the number of users in the system. This might degrade the scalability of the anonymous system especially in a large scaled network such as the Internet. For instance, Tor network has approximately thousands of onion routers and hundreds of thousands of users (2011). This puts emphasis on the demand for a more efficient and scalable circuit construction in a practical and pragmatic setting.

Second, to achieve forward secrecy (which means that a router's corruption should not reveal anything about communication prior to the corruption), the previous schemes require to frequently change the keys of routers (Kate, 2009),(Catalano, 2009),(Dingledin, 2004). This is due to the fact that each router establishes long-term symmetric keys with users, which is vulnerable to the router corruption attack. Therefore, if it is possible to build a circuit without establishing any symmetric key between users and onion routers, forward secrecy could be enhanced since onion routers are not enforced to store the long-term symmetric keys.

Contribution :

In this study, a novel identity-based onion routing protocol is proposed. The circuit construction is embedded into the message delivery process from a user to a service provider with a non-interactive single pass on the basis of the Boneh-Franklin identity-based setting (2001). Each onion router in the circuit just decrypts the received ciphertext with its own secret key and forwards it to the next router. As the symmetric keys do not need to be shared between users and each onion router, the communication cost for building a circuit is significantly reduced and the forward secrecy is enhanced as long as the primitive encryption scheme is secure. In addition, each onion router is only required to store $O(1)$ key in the proposed scheme, while requiring comparable computation overhead to the previous onion routing protocols. Certificates management can be also avoided by identity-based encryption as in (Kate, 2009) and (Catalano, 2009). The proposed scheme could be utilized as an efficient solution to the anonymous non-interactive (one-way) communication such as voting.

Organization :

The remainder of this paper is organized as follows. In Section 2, we introduce background information and preliminaries relevant to the identity-based cryptography and one-way anonymous key agreement protocol. In Section 3, we propose a novel identity-based onion routing protocol. In Section 4, we analyze the proposed scheme in terms of the efficiency. In Section 5, we conclude our paper.

PRELIMINARIES

Bilinear Pairing :

Let $\mathbb{G}_1$ be an additive cyclic group of prime order q and $\mathbb{G}_2$ be a multiplicative cyclic group of same order. A map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ is said to be bilinear if $\hat{e}(aP, bQ) = \hat{e}(P, b)^{ab}$ for all $P, Q \in \mathbb{G}_1$ and $a, b \in \mathbb{Z}_q^*$, and non-degenerate if $\hat{e}(P, P) \neq 1$ for the generator P of $\mathbb{G}_1$.

Then, our key distribution scheme can be built from any efficiently computable non-degenerate bilinear map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ between two groups $\mathbb{G}_1$, $\mathbb{G}_2$ as long as the bilinear Diffie-Hellman (BDH) problem is hard. The BDH problem is defined as a follow.

Definition 1 (Bilinear Diffie-Hellman Problem) The bilinear Diffie-Hellman problem is to compute $\hat{e}(P, P)^{abc} \in \mathbb{G}_2$, given a generator P of $\mathbb{G}_1$ and elements aP, bP, cP for $a, b, c \in \mathbb{Z}_q^*$.

Security Requirements :

We define security of onion routing protocols using the same properties stated in (Kate, 2009). We replace session key secrecy with confidentiality property since the proposed scheme does not need to establish a symmetric key between a user and onion routers. As we consider anonymous one-way communications from a user to a service provider, we will use the terms ‘service provider’ and ‘receiver’ interchangeably henceforth.

1. **Anonymity:** It should be infeasible for an attacker to recognize a link between an anonymous sender and a (non-anonymous) receiver.
2. **Integrity:** It should be possible to recognize those onions that are longer than a pre-specified upper-bound. (Let n be a pre-specified upper bound for the number of routers in a circuit. Then we say that an onion routing protocol satisfies integrity if its is possible to recognize an onion ciphertext which is intended for more than n rounters.)
3. **Correctness:** The recipient receives the original message prepared by the sender if all routers in the circuit correctly execute the protocol.
4. **Confidentiality:** It should be infeasible for anyone other than the intended receiver to obtain any information of the message forwarded in the circuit.

PROPOSED SCHEME

In this section, a novel onion routing protocol is proposed. The proposed scheme exploits an identity-based encryption, especially Boneh-Franklin's basic scheme (2001), as a primitive trapdoor permutation to encapsulate the onion. Since onion routers in the circuit perform decryptions only with their own secret keys, the symmetric key agreement procedures are eliminated in the circuit construction.

In the proposed scheme, a circuit construction is embedded into a non-interactive message delivery process. More precisely, a user encrypts a message for a receiver, and adds several layers of encryption to it with different pseudonyms for each onion router in the circuit. Then, the message is delivered to the

intended receiver as a result of the circuit construction protocol.

System Description :

The onion system consists of (anonymous) users, onion routers, and (non-anonymous) service providers. Let $\mathbb{O} = \{O_1, \ldots, O_l\}$ be the universe of onion routers. As in previous schemes (Kate, 2009), (Catalano, 2009), the service provider acts as a key generation center (KGC) in the proposed scheme[1]. $ID_x \in \mathbb{G}_1$ represents the unique identity of an entity x (users, onion routers, and service providers).

Construction :

The proposed protocol consists of the following three phases: (1) setup, (2) key generation, and (3) circuit construction phases.

Setup: KGC selects a prime q, an additive group $\mathbb{G}_1$ and a multiplicative group $\mathbb{G}_2$ of order q, and generates a bilinear map group system $(q, \mathbb{G}_1, \mathbb{G}_2, \hat{e}(\cdot,\cdot))$. It randomly selects a generator $P \in \mathbb{G}_1$, a random $s \in \mathbb{Z}_q^*$, and computes sP. It chooses a cryptographic hash function $H : \mathbb{G}_2 \to \{0,1\}^*$. KGC publishes all of these values except the master secret key s.

Key generation: For every entity with public identity ID_i, the KGC generates a private key sID_i and sends it to the entity with ID_i securely.

Circuit construction: When a user wants to send a message σ to a service provider, he constructs a circuit and sends the message in the following sequence.

1. The user chooses an ordered sequence of l onion routers $O_1, \ldots, O_l$ at random. For each onion router and service provider, he selects $r_1, \ldots, r_{l+1} \in \mathbb{Z}_q^*$ at random, and generates l+1 pseudonyms $r_1P, \ldots, r_{l+1}P$. (We will denote the service provider O_{l+1} for simple description.) Then, the user constructs a circuit as follows:

- Computes
$$C_{l+1} = \left\langle C_{l+1}^1, C_{l+1}^2 \right\rangle = \left\langle r_{l+1}P, \sigma \oplus H(\hat{e}(r_{l+1}ID_{O_{l+1}}, sP)) \right\rangle$$

- Computes
$$C_i = \left\langle C_i^1, C_i^2 \right\rangle = \left\langle r_iP, (O_{i+1} \middle| C_{i+1}^1 \middle| C_{i+1}^2) \oplus H(\hat{e}(r_iID_{O_i}, sP)) \right\rangle$$

for $l \geq i \geq 1$,
$$C_i = \left\langle C_i^1, C_i^2 \right\rangle = \left\langle r_iP, (O_{i+1} \middle| C_{i+1}^1 \middle| C_{i+1}^2 \oplus H(\hat{e}(r_iID_{O_i}, sP)) \right\rangle$$

where $a|b$ represents the concatenation of strings a and b.

- Finally, sends the onion C_1 to the first onion router O_1 in the circuit.

2. For $1 \leq i \leq l$, upon receipt of the onion C_i by O_i, O_i computes $C_i^2 \oplus H(\hat{e}(C_i^1, sID_{O_i}))$ with its secret key sID_{O_i} and obtains $O_{i+1}, C_{i+1}^1, C_{i+1}^2$. Then, it sends the onion $C_{i+1} = \left\langle C_{i+1}^1, C_{i+1}^2 \right\rangle$ to O_{i+1}.

3. If the service provider receives the onion $C_{l+1} = \left\langle C_{l+1}^1, C_{l+1}^2 \right\rangle$ from the exit onion router O_l, it computes $C_{l+1}^2 \oplus H(\hat{e}(C_{l+1}^1, sID_{O_{l+1}}))$ and obtains a message σ.

The following Fig. 2 shows the example of the circuit construction and message delivery process.

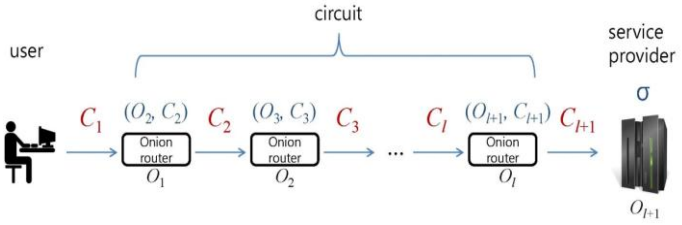

Figure 2: Circuit construction and message delivery process

Security Analysis :

The proposed scheme exploits the Boneh-Franklin's basic encryption scheme as an encryption primitive, which is proved to be secure against chosen plaintext attack (IND-CPA) (2001). However, the security of the proposed scheme can be extended to chosen ciphertext attack (IND-CCA) efficiently by applying a random oracle technique such as Fujisaki-Okamoto transformation (1999).

The proposed scheme trivially achieves correctness and integrity. In addition, the confidentiality is also achieved by the CPA-security of the primitive identity-based encryption as long as the BDH problem is hard (proof can be found in (2001)). Due to the CPA-security, any entity other than the intended onion router or service provider can by no means decrypt the outside layer of encryption. For an anonymous user, the pseudonym r_iP is the only parameter exposed to an onion router O_l in the circuit, and $r_iP \neq r_jP$ for $i \neq j$. It perfectly blinds the identity of the user and guarantees the anonymity of the user during the protocol.

[1] In fact, this assumes the existence of a secure channel of communications between a user and a ser-vice provider. Therefore, another setting, in which the system-wide trusted authority (other than the ordinary service provider) plays the role of the KGC, would likely to be more suitable to practical ap-plications in anonymous networks.

In the previous schemes, the forward secrecy is achieved in a course-grained level. The network system frequently changes the keys of each onion router in order to minimize the exposed period of symmetric keys (referred to as 'windows of vulnerability') to attackers who captured and compromised any onion routers. This incurs a significant large communication overhead for users to contact KGC or onion routers to obtain any updated keys. However, in the proposed scheme, forward secrecy can be simply enhanced compared to the previous schemes since a user does not need to establish session keys with each onion router. This resolves the problem of `windows of vulnerability' of the previous schemes.

Message Authentication and Confirmation :

In most one-way anonymous communications, it is also required to authenticate the non-anonymous service provider. With the proposed scheme, the message delivery is implicitly confirmed; the sender is assured that only the service provider can decrypt the message. However, the explicit confirmation can be also achieved by incorporating any symmetric-key based challenge-response protocol and replacing the message σ with any symmetric key.

ANALYSIS

In this section, efficiency of the proposed scheme is analyzed and compared to the previous schemes, that is Tor (2011), PB-OR (Kate, 2009), and CL-OR (Catalano, 2009), in terms of the communication, computation, and storage overhead needed for n users to construct each circuit of length l. Communication overhead represents the number of messages exchanged to build a circuit. Computation overhead represents the amount of operations and computing time required to build a circuit. Storage overhead represents the amount of secret keys needed to store for a user and an onion router in the circuit.

Efficiency :

Table 1 shows the efficiency comparison result among the schemes in terms of the storage and communication overhead required during the circuit construction.

Table 1: Efficiency Comparison

Schemes	Tor (2011)		PB-OR (Kate, 2009)		CL-OR (Catalano, 2009)		Proposed scheme	
	user	router	user	router	user	router	User	router
Storage	l	$n+1$	$2l$	$2n+1$	l	$n+2$	0	1
Communication	$l(l+1)^{\S}$		$2l^{\S}$		$3l^{\S}$		l	

§: Encrypted by AES

As shown in Table 1, the analysis result indicates that the proposed scheme significantly reduces the storage overhead for the circuit construction. In the proposed scheme, a user does not need to store any secret key, since the circuit construction is done with non-interactive key agreement protocol between the user and each onion router. The only task the user need to do in the circuit construction is encrypting the messages with each identity of the selected onion routers in the circuit. In addition, each onion router is required to store just a single secret key of its own. Therefore, the proposed scheme is the most efficient among the scheme in terms of the storage overhead.

When it comes to the communciation overhead, the proposed scheme needs l communications during the circuit construction, which is the least amount of communication cost among the scheme. Additionally, the l communications between the user and the onion routers in the circuit do not need to be encrypted with AES in the proposed scheme. As opposed to the proposed scheme, however, the other schemes require each communication to be encrypted with AES block cipher in order for securing circuit construction messages. Therefore, additional symmetric encryption overhead is also eliminated in the proposed scheme.

Implementation :

Next, we analyze and measure the computation cost for encryption and decrypting a message during the circuit consturction by a user and each router in the circuit. We used a Type A curve (in the pairing-based cryptography (PBC) library (2010)) providing groups in which a bilinear map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ is defined. Although such curves provide good computational efficiency (especially for pairing computation), the same does not hold from the point of view of the space required to represent group elements. Indeed each element of $\mathbb{G}_1$ needs 512 bits at an 80-bit security level and 1536 bits when 128-bit of security are chosen.

TABLE 2 shows the computational time result. For each operation, we include a benchmark timing. Each cryptographic operation was implemented using the pairing-based cryptography (PBC) library ver. 0.4.18 (2010) on a 3.0 GHZ processor PC. The public key parameters were selected to provide 80-bit security level. The implementation uses a 160-bit elliptic curve group based on the supersingular curve $y^2 = x^3 + x$ over a 512-bit finite field. The computational cost is analyzed in terms of the pairing, exponentiation operations in $\mathbb{G}_1$ and $\mathbb{G}_2$. The comparatively negligible hash and exclusive-or operations are ignored in the result.

Table 2: Comparison of Computation Cost

Schemes	Time (ms)	Tor (2011)		PB-OR (Kate, 2009)		CL-OR (Catalano, 2009)		Proposed scheme	
		user	router	user	router	user	router	User	router
RSA encryption	0.1	l	0	0	0	0	*0*	0	0
RSA decryption	2.7	0	n	0	0	0	*0*	0	0
Modular exponentiation	1.5	$2l$	$2n$	0	0	$3l$	$2n$	0	0
Multiplication in $\mathbb{G}_1$	1.0	0	0	$2l$	0	0	*0*	l	0
Pairing	2.0	0	0	l	n	0	*0*	l	n
Computation (ms)		$3.1l$	$5.7n$	$4.9l$	$2.9n$	$4.5l$	$3n$	$3.9l$	$2.9n$

The computation cost is also improved compared to the other identity-based schemes (Kate, 2009), (Catalano, 2009). In the proposed scheme, a user is required to perform l multiplication operations in $\mathbb{G}_1$ and l pairing operations. Even if the computation overhead for a user is slightly larger than that of Tor (2011), it is the most efficient in the identity-based cryptography setting. In addition, the computation overhead of an onion router is the least among the schemes. These properties suggest that the proposed scheme could be a practical and efficient way to allow anonymity networks to scale gracefully.

CONCLUSIONS

Onion routing protocols achieve low-latency anonymous communication on public networks. Up to date, many onion routing protocols have been proposed, such as Tor network, in order to implement the anonymous network connection in the public networks. Although the previous schemes' multi-pass cryptographic circuit construction appears satisfactory, their circuit construction protocols have some drawbacks with regard to the efficiency and security.

The proposed scheme enhances the efficiency and security of the onion routing protocol by eliminating the necessity of interactive and iterative symmetric key agreement procedures between users and onion routers. Considering the importance of scalability in large scaled networks such as the Internet, the proposed scheme could be exploited as an efficient solution to anonymous networks.

ACKNOWLEDGEMENT

This work was supported by the research fund of Signal Intelligence Research Center supervised by the Defense Acquisition Program Administration and Agency for Defense Development of Korea.

REFERENCES

[1] Boneh, D., Franklin, M. (2001): Identity-Based Encryption from the Weil Pairing, *Proc. Crypto*. LNCS 2139: 213-229.

[2] Catalano, D., Fiore, D., Gennaro, R. (2009): Certificateless Onion Routing. *Proc. ACM Conference on Computer and Communications Security*: 151-160.

[3] Chaum, D. (1981): Untraceable Electronic Mail, Return Address and Digital Pseudonyms, *Communications of the ACM* 24(2): 84-88.

[4] Dingledin, R., Mathewson, N., Syverson, P. (2004): Tor: The Second-Generation Onion Router. *Proc. USENIX Security Symposium*: 302-320.

[5] Dingledin, R., Mathewson, N. (2011): Tor Protocol Specification. http://www.torproject.org/svn /trunk/doc /spec/tor-spec.txt. Accessed 04-Jan-2011.

[6] Feigenbaum, J., Johnson, A., Syverson, P. (2007): A Model of Onion Routing with Provable Anonymity, *Proc. 11th Financial Cryptography and Data Security Conference*.

[7] Fujisaki, E., Okamoto, T. (1999): Secure Integration of Asymmetric and Symmetric Encryption Schemes. *Proc. Crypto*. LNCS 2139: 537-554.

[8] Goldschlag, D., Reed, M., Syverson, P. (1996): Hiding Routing Information. *Proc. Information Hiding*. LNCS 1174: 137-150.

[9] Johnson, A., Syverson, P. (2009): More Anonymous Onion routing Through Trust, *Proc. IEEE Computer Security Foundations Symposium*.

[10] Kate, A., Zaverucha, G., Goldberg, I. (2009): Pairing-Based Onion Routing with Improved Forward Secrecy. *ACM Transactions on Information and System Security*.

[11] Mauw, S., Verschuren, J.H.S., Vink, E.P.De (2004): A Formalization of Anonymity and Onion Routing, *Proc. 9th European Symposium on Research in Computer Security (ESORICS)*: 109-124.

[12] Overlier, L., Syverson, P. (2007): Improving Efficiency and Simplicity of Tor Circuit Establishment and Hidden Services. *Proc. PETS*: 134-152.

[13] Reed, M. G., Syverson, P. F., Goldschlag, D. M. (1998): Anonymous Connections and Onion routing. *IEEE Journal on Selected Areas in Communications* 16: 482-494.

[14] The Pairing-Based Cryptography Library (2010): http://crypto.stanford.edu/pbc/. Accessed 11-Sep-2010.

BIOGRAPHICAL NOTES

Junbeom Hur received the BS degree from Korea University, Seoul, South Korea, in 2001, and the MS and PhD degrees from KAIST in 2005 and 2009, respectively, in computer science. He was with the University of Illinois at Urbana-Champaign as a postdoctoral researcher from 2009 to 2011 and the School of Computer Science and Engineering at the Chung-Ang University, South Korea, as an assistant professor from 2011 to 2015. He is currently an associate professor in the Department of Computer Science and Engineering, Korea University, South Korea. His research interests include information security, mobile security, and applied cryptography.

Dong Kun Noh received BS, MS, and PhD degrees in EECS from Seoul National University in 2000, 2002, and 2007, respectively. He has been in University of Illinois at Urbana-Champaign as a postdoctoral researcher from 2007 to 2010. He is currently an associate professor in Dept. of Smart Systems Software at the Soongsil Univ. in Korea. His research interests include mobile computing, cyber-physical system, information security, and wireless sensor network.