Efficient and Secure Identity-Based Onion Routing

Junbeom Hur

School of Computer Science and Engineering, Chung-Ang University
84 Heukseok-ro, Dongjak-gu,Seoul 156-756, Korea
Email: jbhur@cau.ac.kr

Dong Kun Noh*

School of Electronic Engineering, Soongsil University
369 Sangdo-Ro, Dongjak-Gu, Seoul 156-743, Korea
Email: dnoh@ssu.ac.kr

Onion routing protocols achieve low-latency anonymous communication on public networks. 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 Classifications: *C.2.2 Network Protocols, D.4.6 Security and Protection*

1. INTRODUCTION

As we move to ubiquitous network environment, it has become apparent that our privacy is at stake (Vanhilst *et al*, 2009). These privacy concerns were recognized since the beginning of the internet age, and anonymous 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 an 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

* Corresponding author: Dong Kun Noh

Manuscript received: 14 August 2012
Communicating Editor: Seok-Hun Kim

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 *et al*, 2004; Dingledin and Mathewson, 2011).

Onion routing relies on public key cryptography. It allows the onion routing to encrypt layers of onions such that only intended recipients of each layer can decrypt it with their private keys. Each hop along the route only knows about the previous hop and the next hop. If the attackers compromise a host in the network of onion routers, they will only be able to see where the onion came from on the last hop, and where it should be sent to on the next hop. The data source and destination of the onion are hidden.

Specifically, 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. Figure 1 following shows the example of onion constructed by a user (source) for a service provider (destination), which is triple-encrypted (that is, onion A, B, and C).

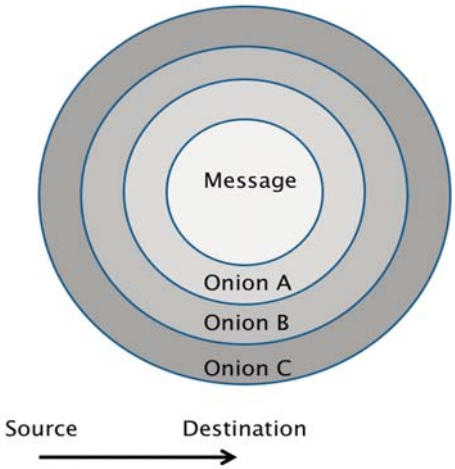

Figure 1: Onion example

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.

Each onion layer provides the material to generate the symmetric keys used for passing the data back and forth (Syverson, 1999). The final layer contains neither a destination address nor meaningful content to be transmitted. Thus, the onion for a three-hop route through R_1, R_2, R_3 could be abstractly constructed as follows:

$$E(PK_{R_1},[K_1,R_2,E(PK_{R_2},[K_2,R_3,E(PK_{R_3},[K_3,Pad])])])$$

where $E(K,M)$ is an encryption of message M under the key K. PK_{R_i} is a public key for R_i, and K_i is session key material to be shared between the route originator and R_i. This is the circuit construction process where R_1, R_2, R_3 are the onion routers that the sender (that is, route originator, or a source) selected during the anonymous communication.

It is important to note that what is encrypted for the last layer in the original onion routing protocol is just the symmetric key material to be used for passing data back and forth and then just empty padding. Thus, there is no message at the innermost core of the onion. The essential feature of the onion routing is that public keys are used to lay a cryptographic circuit of symmetric keys, which is then used to pass data.

Onion routing was designed to facilitate anonymous bidirectional communication with low-latency for remote login, web browsing, chat, and other interactive applications for example, (Syverson, 2011). By only using the public key cryptography to establish session keys, the onion routing can allow for throughput and latency that would be infeasible if public key operations were needed for each message or packet passing through the system. Following multihop free-route path selection through a network of independently managed onion routers prevents an adversary from observing traffic entering and leaving the system.

Figure 2 following shows the anonymous communication process in the onion routing protocol between the user and the service provider through a set of onion routers that the sender randomly selected (that is a circuit) before the communication.

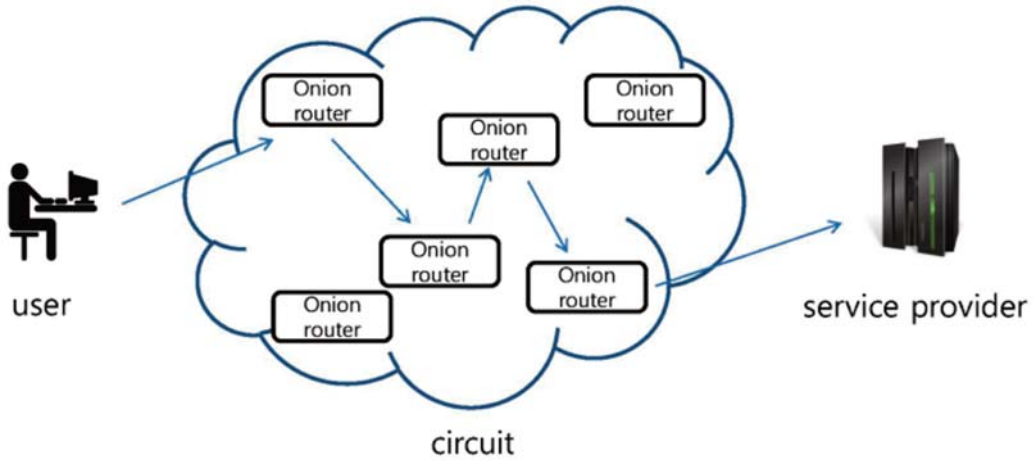

Figure 2: Onion routing protocol

1.1 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 decrypt 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 (Dingledin and Mathewson, 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 (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* (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 centre (KGC). PB-OR uses the original onion routing idea 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* (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 *et al* (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 (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* (2007) proposed formal models for rigorous analysis of anonymity in onion routing protocols.

Katti *et al* (2005) have shown it is possible to design anonymizing peer-to-peer overlays that do not need a public key infrastructure (PKI). Katti *et al* show how to perform onion routing without public key cryptography. Katti *et al*'s scheme is based on the idea of information slicing. Their information slicing protocol can hide the source, the destination, the path, and the content of the

message, even when the sender does not have the public keys of the nodes in the overlay. Specifically, to provide anonymous communication, each node along the path (including the destination node) needs a particular piece of information, which should be hidden from other nodes in the network. For example, the destination needs to learn the content of the message without revealing that content to other nodes, while each intermediate relay needs to learn its next hop without other nodes in the network knowing that information. They divide the information needed by a particular node into many small random pieces. These information pieces are then delivered along disjoint paths that meet only at the intended node. Thus, only the intended node has enough bits to decode the information content. They call this approach information slicing because it splits the information traditionally contained in an onion peel into multiple pieces/slices.

Usually onion routing is specified through asymmetric cipher and thus is inefficient. In Tor, it is suggested to employ symmetric cipher to encrypt the packets in onion routing. The suggestion in Tor is simple and it is a direct employment of Diffie-Hellman handshake to generate the secret keys for the routers' symmetric cipher. Recently, Peng (2011) shows that direct application of Diffie-Hellman handshake to implement key generation and exchange in onion routing is not efficient in communication as multiple instances of Diffie-Hellman handshake needs a lot of additional communication. Moreover, its efficiency improvement for the sender is not satisfactory. So Peng's scheme designs an advanced application of Diffie-Hellman key exchange technique, Diffie-Hellman chain. This scheme saves a sender's cost and needs less communication for Diffie-Hellman key exchange. With the efficiency improvement in this paper, Tor can be applied to communication networks with weaker computational capability and smaller communicational bandwidth.

1.2 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 (Dingledin, 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 *et al*, 2009; Catalano *et al*, 2009; Dingledin *et al*, 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.

1.3 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 also be avoided by identity-based encryption as in Kate *et al* (2009) and Catalano *et al* (2009). The proposed scheme could be utilized as an efficient solution to the anonymous non-interactive (one-way) communication such as voting.

1.4. 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.

2. Preliminaries

2.1 Bilinear Pairing

Let G_1 be an additive cyclic group of prime order q and G_2 be a multiplicative cyclic group of same order. A map $\hat{e} : G_1 \times G_1 \rightarrow G_2$ is said to be bilinear if $\hat{e}(aP,bQ) = \hat{e}(P,b)^{ab}$ for all $P,Q \in G_1$ and $a,b \in Z_q^*$, and non-degenerate if $\hat{e}(P,P) \neq 1$ for the generator P of G_1.

Then, our key distribution scheme can be built from any efficiently computable non-degenerate bilinear map $\hat{e} : G_1 \times G_1 \rightarrow G_2$ between two groups G_1, G_2 as long as the bilinear Diffie-Hellman (BDH) problem is hard. The BDH problem is defined as follows:

Definition 1 (Bilinear Diffie-Hellman Problem): The bilinear Diffie-Hellman problem is to compute $\hat{e}(P,P)^{abc} \in G_2$, given a generator P of G_1 and elements aP, bP, cP for $a,b,c \in Z_q^*$.

Weil pairing (2001) or Tate pairing (2002) on elliptic curves can be used as an efficiently computable nondegenerate bilinear map. There are efficient and practical ways to find such maps; see for example Miller (2004) and Choie and Lee (2004).

2.2 Security Requirements

We define security of onion routing protocols using the same properties stated in Kate *et al* (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 it is possible to recognize an onion ciphertext which is intended for more than n routers.)
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.

We refer the reader to the work in Camenisch (2005) and (2003) for formal security definitions for the problem of onion routing.

3. Proposed Schemes

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 trap-door 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.

3.1 System Description

The onion system consists of (anonymous) users, onion routers, and (non-anonymous) service providers. Let $\mathbf{O}=\{O_1,\ldots O_i\}$ be the universe of onion routers. As in previous schemes (Kate *et al*, 2009; Catalano *et al*, 2009), the service provider acts as a key generation centre (KGC) in the proposed scheme. (In fact, this assumes the existence of a secure channel of communications between a user and a service 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 be more suitable to practical applications in anonymous networks.) $ID_x \in \mathbb{G}_1$ represents the unique identity of an entity x (users, onion routers, and service providers).

3.2 Construction

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

1. **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 as a public key. It chooses a cryptographic hash function $H : \mathbb{G}_2 \rightarrow \{0,1\}^*$. KGC publishes all of these values except the master secret key s.
2. **Key generation:** For every entity with public identity ID_i, the KGC generates a private key sID and sends it to the entity with ID_i securely.
3. **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 $r_1,\ldots,r_{l+1} \in Z_q^*$ 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:
   I. Computes $C_{l+1} = \left\langle C_{l+1}^1, C_{l+1}^2 \right\rangle = \left\langle r_{l+1}P, \sigma \oplus H(\hat{e}(r_{l+1}ID_{O_{l+1}}, sP)) \right\rangle$
   II. Computes for $C_i = \left\langle C_i^1, C_i^2 \right\rangle = \left\langle r_iP, (O_{i+1} \| C_{i+1}^1 \| C_{i+1}^2) \oplus H(\hat{e}(r_iID_{O_i}, sP)) \right\rangle$ for $l \geq i \geq 1$,

   $C_i = \left\langle C_i^1, C_i^2 \right\rangle = \left\langle r_iP, (O_{i+1} \| C_{i+1}^1 \| C_{i+1}^2 \oplus H(\hat{e}(r_iID_{O_i}, sP)) \right\rangle$ where $a\|b$ represents the concatenation of strings a and b.
   III. Finally, sends the onion C_1 to the first onion router O_1 in the circuit.
2. For $1 \leq i \leq 1$, 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 $C_{l+1} = \left\langle C_{l+1}^1, C_{l+1}^2 \right.$ 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.

Figure 3 following shows the example of the circuit construction and message delivery process.

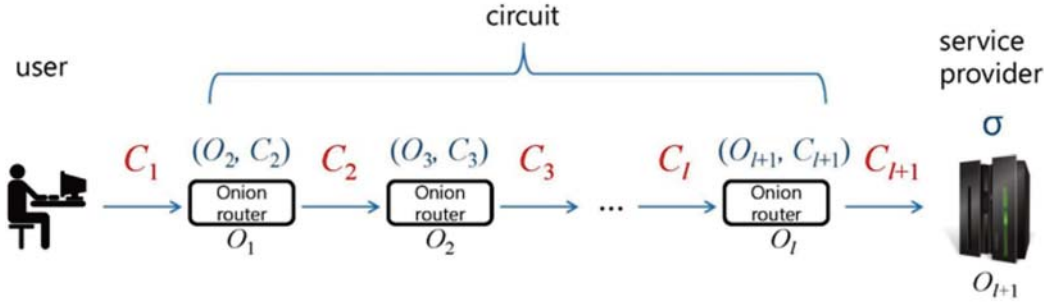

Figure 3: Circuit construction and message delivery process

3.3 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 (Fujisaki and Okamoto, 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 IND-CPA (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.

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.

3.4 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 with any symmetric key.

4. Analysis

In this section, efficiency of the proposed scheme is analyzed and compared to the previous schemes, that is Tor (Dingledin and Mathewson, 2011), PB-OR (Kate *et al*, 2009), and CL-OR (Catalano *et al*, 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.

4.1 Efficiency

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

Schemes	**Tor (Dingledin & *Mathewson*, 2011)**		**PB-OR (Kate *et al*, 2009)**		**CL-OR (Catalano *et al*, 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

Table 1: Efficiency Comparison

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 needs 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 communication 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 to secure circuit construction messages. Therefore, additional symmetric encryption overhead is also eliminated in the proposed scheme.

Figure 4 following shows the simulation result. The horizontal axis represents the number of communications (l) and the vertical axis represents the total communication cost in logarithm scale with different number of communications. As the simulation result shows, the proposed scheme requires the least number of communications as l increases. Therefore, the proposed scheme is the most efficient in terms of the communication overhead. It is important to note that the communications are not encrypted in the proposed scheme as opposed to the other schemes where all of the communications are encrypted by AES block cipher algorithm. Thus, the proposed scheme is also more efficient than the other schemes in terms of the computation cost.

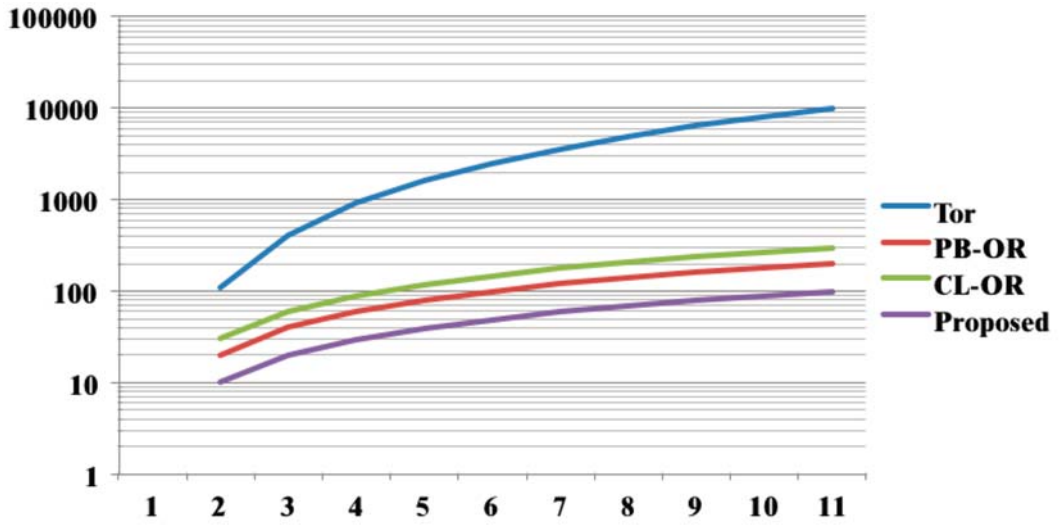

Figure 4: Communication cost

4.2 Implementation

Next, we analyze and measure the computation cost for encryption and decrypting a message during the circuit construction 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 $\hat{e} : G_1 \times G_1 \rightarrow G_2$ 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 G1 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 G1 and G1. The comparatively negligible hash and exclusive-or operations are ignored in the result.

Schemes	Time (ms)	Tor (Dingledin & Mathewson, 2011)		PB-OR (Kate *et al*, 2009)		CL-OR (Catalano *et al*, 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 G1	1.0	0	0	$2l$	0	0	0	l	0
Pairing	2.0	0	0	l	n	0	0	l	n
Computation (ms)		$3.1l$	$5.7n$	$4.9l$	$2.9n$	$4.5l$	$3n$	$3.9l$	$2.9n$

Table 2: Comparison of Computation Cost

The computation cost is also improved compared to the other identity-based schemes (Kate *et al*, 2009; Catalano *et al*, 2009). In the proposed scheme, a user is required to perform l multiplication operations in G1 and l pairing operations. Even if the computation overhead for a user is slightly larger than that of Tor (Dingledin and Mathewson, 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.

5. Conclusions and Future Work

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.

As the proposed scheme is constructed on the basis of the identity-based cryptography, the key escrow problem is inherent. The key escrow problem is that the key generation centre is able to decrypt all the ciphertext exchanged in the system by determining every key of nodes using its master secret key. Thus, the security of the entire system is totally dependent on the trustworthiness of the key generation centre.

As a future work, we will study the key escrow problem in the identity-based onion routing protocol. Removing the key escrow from the identity-based encryption would be a very

challenging problem since most of the public crypto algorithms without the public key infrastructure such as identity-based encryption require the trusted key generation centre. Therefore, we observe that removing the key escrow problem in the identity-based onion routing protocol would be a promising and interesting research topic as a future work.

Acknowledgements

This work was supported by the Chung-Ang University Research Grants in 2011, and the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2012R1A1A1001835).

References

BONEH, D. and FRANKLIN, M. (2001): Identity-Based Encryption from the Weil Pairing, *Proc. Crypto*. LNCS 2139: 213–229.

CAMENISCH, J. and LYSYANSKAYA, A. (2005): A Formal Treatment of Onion Routing, *Proc. Crypto*. LNCS 3621: 169–187.

CATALANO, D., FIORE, D. and GENNARO, R. (2009): Certificateless Onion Routing. *Proc. ACM Conference on Computer and Communications Security*: 151–160.

CHAUM, D. (1981): Untraceable Electronic Mail, Return Address and Digital Pseudonyms, *Communications of the ACM* 24(2): 84–88.

CHOIE, Y. J. and LEE, E. (2004): Implementation of Tate Pairing on Hyperelliptic Curves of Genus 2, *Proc. International Conference on Information Security and Cryptology*, LNCS 2971: 97–111.

DINGLEDIN, R., MATHEWSON, N. and SYVERSON, P. (2004): Tor: The Second-Generation Onion Router. *Proc. USENIX Security Symposium*: 302–320.

DINGLEDIN, R. and MATHEWSON, N. (2011): Tor Protocol Specification. http://www.torproject.org/svn /trunk/doc /spec/tor-spec.txt. Accessed 04-Jan-2011.

FEIGENBAUM, J., JOHNSON, A. and SYVERSON, P. (2007): A Model of Onion Routing with Provable Anonymity, *Proc. 11th Financial Cryptography and Data Security Conference*.

FUJISAKI, E. and OKAMOTO, T. (1999): Secure Integration of Asymmetric and Symmetric Encryption Schemes. *Proc. Crypto*. LNCS 2139: 537–554.

GALBRAITH, S. D., HARRISON, K. and SOLDERA, D. (2002): Implementing the Tate Pairing, *Proc. 5th International Symposium on Algorithmic Number Theory*, LNCS 2369: 324–337.

GOLDSCHLAG, D., REED, M. and SYVERSON, P. (1996): Hiding Routing Information. *Proc. Information Hiding*. LNCS 1174: 137–150.

HWU, J., CHEN, R. and LIN, Y. (2006): An Efficient Identity-based Cryptosystem for End-to-end Mobile Security, *IEEE Transactions on Wireless Communications* 5: 2586–2593.

JOHNSON, A. and SYVERSON, P. (2009): More Anonymous Onion routing Through Trust, *Proc. IEEE Computer Security Foundations Symposium*.

KATE, A., ZAVERUCHA, G. and GOLDBERG, I. (2009): Pairing-Based Onion Routing with Improved Forward Secrecy. *ACM Transactions on Information and System Security* 13(4): 1–32.

KATTI, S., KATABI, D. and KATARZYNA, P. (2005) Slicing the Onion: Anonymous Routing without PKI, *Proc. 4th ACM Workshop on Hot Topics in Networks (HotNets)*.

MAUW, S., VERSCHUREN, J.H.S. and VINK, E.P.De (2004): A Formalization of Anonymity and Onion Routing, *Proc. 9th European Symposium on Research in Computer Security (ESORICS)*: 109–124.

MILLER, V.S. (2004): The Weil Pairing and Its Efficient Calculation, *Journal of Cryptology*. 17(4): 235–261.

MOLLER, B. (2003): Provably Secure Public-Key Encryptionfor Length-Preserving Chaumian Mixes, *Proc. CT-RSA*. LNCS 2612: 244–262.

OVERLIER, L. and SYVERSON, P. (2007): Improving Efficiency and Simplicity of Tor Circuit Establishment and Hidden Services. *Proc. PETS*: 134–152.

PENG, K. (2011): Efficiency Optimisation of Tor Using Diffie-Hellman Chain, *Proc. The 10th International Conference on Networks (ICN 2011)*: 41–46.

REED, M.G., SYVERSON, P.F. and GOLDSCHLAG, D.M. (1998): Anonymous Connections and Onion routing. *IEEE Journal on Selected Areas in Communications* 16: 482–494.

SYVERSON, P. (2011): A Peel of Onion. *Proc. Annual Computer Security Applications Conference (ACSAC 2011)*: 123–135.

SYVERSON, P., REED, M. and GOLDSCHLAG, D. (1999): Onion Routing Access Configurations. *Proc. DARPA Information Survivability Conference & Exposition (DISCEX 00)*: 34–40.

THE PAIRING-BASED CRYPTOGRAPHY LIBRARY (2010): http://crypto.stanford.edu/pbc/. Accessed 11-Sep-2010.

VANHILST, M., FERNANDEZ, E.B. and BRAZ, F. (2009): A Multi-Dimensional Classification for Users of Security Patterns, *Journal of Research and Practice in Information Technology* 41(2): 87–97.

Biographical Notes

Junbeom Hur

Junbeom Hur *received the BSc degree in computer science from Korea University in 2001, the MSc and PhD degrees in computer science from KAIST in 2005 and 2009, respectively. He was at the University of Illinois at Urbana-Champaign as a post-doctoral researcher from 2009 to 2011. He is currently an assistant professor in the School of Computer Science and Enginnering at the Chung-Ang University in Korea. His research interests include information security, mobile computing security, network security, and cryptography.*

Dong Kun Noh

Dong Kun Noh *received BSc, MSc, and PhD degrees in EECS from Seoul National University in 2000, 2002, and 2007, respectively. He was at the University of Illinois at Urbana-Champaign as a postdoctoral researcher from 2007 to 2010. He is currently an assistant professor in the School of Electronic Engineering at the Soongsil University in Korea. His research interests include mobile embedded system, mobile communication, QoS control, and information security on wireless ad-hoc and sensor network.*