

Anonymity and Security in Delay Tolerant Networks

Aniket Kate, Greg Zaverucha*, and Urs Hengartner

David R. Cheriton School of Computer Science
University of Waterloo
Waterloo ON, N2L 3G1, Canada
{akate,gzaveruc,uhengart}@cs.uwaterloo.ca

March 26, 2007

Abstract

A delay tolerant network (DTN) is a store and forward network where end-to-end connectivity is not assumed and where opportunistic links between nodes are used to transfer data. An emerging application of DTNs are rural area DTNs, which provide Internet connectivity to rural areas in developing regions using conventional transportation mediums, like buses. Potential applications of these rural area DTNs are e-governance, telemedicine and citizen journalism. Therefore, security and privacy are critical for DTNs. Traditional cryptographic techniques based on PKI-certified public keys assume continuous network access, which makes these techniques inapplicable to DTNs because of their disconnected nature. We present the first anonymous communication solution for DTNs and introduce a new anonymous authentication protocol as part of it. Furthermore, we present a security infrastructure for DTNs to provide efficient secure communication. Our anonymity and security solutions are based on identity-based cryptography. We show that our solutions have better performance than previously proposed security infrastructures for DTNs.

1 Introduction

Today's wired and wireless networks have enabled a wide range of devices to be interconnected over vast distances. In spite of their success, parts of the world are still out of reach, due to a lack of end-to-end connectivity. In most of the developing regions, reliable end-to-end network connections are not available nor will be in the near future due to problems like erratic power supply and high infrastructure costs.

Delay tolerant networks (DTNs) [13] are a potential low-cost solution to the problem of connecting devices in areas where end-to-end network connectivity cannot be assumed. DTNs use intermediate nodes to take custody of the transferred data and to forward this data as the opportunity arises. Due to the disconnected nature of DTNs, traditional PKI-based security and privacy solutions are not applicable to these networks.

This paper presents a comprehensive solution for anonymous *and* secure communication in DTNs. To address the disconnected nature of DTNs, our solution exploits identity-based cryptography (IBC) [27]. In particular, our contributions are as follows:

*Research supported by an NSERC PGS-D postgraduate scholarship

1. We introduce a new, IBC-based, anonymous authentication protocol and use this protocol to build the first system for providing anonymous communication in DTNs.
2. We present an IBC-based security infrastructure for DTNs that is more efficient than an existing security infrastructure for DTNs [25].

We provide an overview of DTNs and IBC in Section 2. In Section 3, we discuss related work in the area of DTN security and anonymous communication. In Section 4, we present our architecture for secure DTN communication. Section 5 describes our new anonymous authentication protocol and our anonymity architecture, which, as it turns out, can be integrated into our security architecture with no changes in the setup. This anonymity architecture forms the basis for anonymous and secure communication in DTNs, which we discuss in Section 6. Section 7 investigates system and network-related issues, like performance, routing and billing. In Appendix A, we briefly introduce bilinear pairings and the bilinear Diffie-Hellman problem. We discuss the security and anonymity provided by our anonymous authentication protocol in Appendix B.

2 Background

In this section, we give a survey of DTNs and the special case of rural area DTNs. We also review an IBC scheme, namely the Sakai-Ohgishi-Kasahara key agreement protocol [24] in a Boneh-Franklin identity-based encryption setup [5]. Finally, we give an overview of hierarchical identity-based cryptography (HIBC).

2.1 Delay Tolerant Networks (DTNs)

DTNs deal with communication in extreme and performance-challenged environments, where continuous end-to-end connectivity cannot be assumed. In a DTN, nodes use opportunistic connectivity over intermittent links for communication. Such opportunistic links are generally provided by mobile routers. They offer connectivity by acting as “data mules” to carry data to and from servers with continuous network connectivity (i.e., Internet access). There are many applications for DTNs. In developing regions, especially rural areas, they can be used to provide network access for education, health care or government services [8]. They can also augment low bandwidth Internet connections to transfer large files at low cost, while using the Internet connection for the control messages [31]. DTNs are also applicable in vehicular ad-hoc networks (VANETs) [22], undersea communication [28] and interplanetary networks [9]. For a detailed discussion of DTNs, we refer to Farrell and Cahill’s recent book [14].

2.1.1 Rural Area DTNs.

Though DTNs arise in many situations and may take many forms, our terminology in this paper is slanted towards the particular example of rural area DTNs. The use of this concrete example aids exposition and provides motivation, but does not reduce the applicability of our work to other types of DTNs.

Seth et al. [26] provide a detailed discussion of rural area DTNs. Figure 1 illustrates a typical rural area DTN. We now give a brief overview.

- The approach is applicable to villages and rural areas with no Internet connectivity due to geographic or economic constraints.

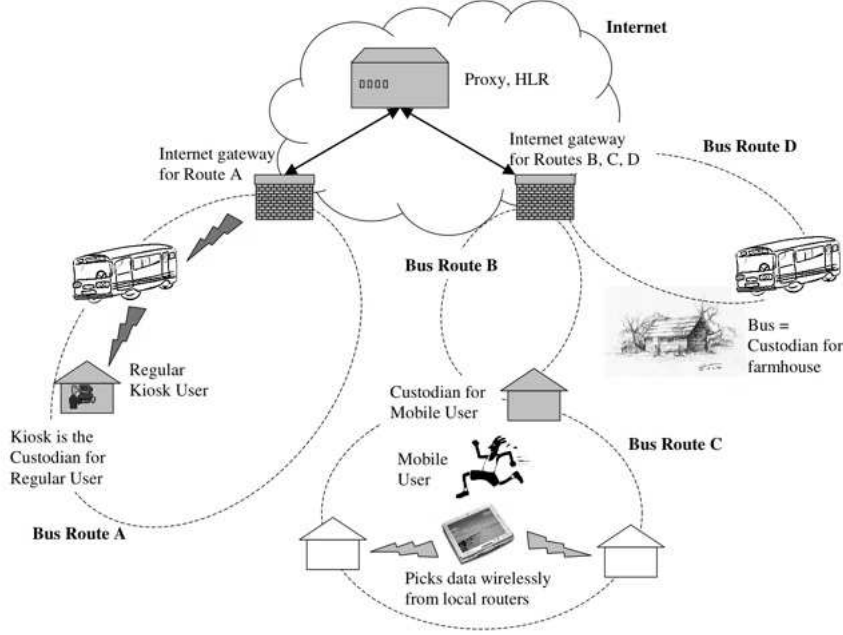


Figure 1: A Typical Rural Area DTN [26].

- There is an Internet connection available in a nearby town and a transport medium from the rural area to the town in the form of a vehicle, such as a bus or a car.
- The terminal with Internet connectivity is called the *gateway*. A transport medium that carries data from the end users in a village to a gateway is called a *mobile router*.
- There is also a special static router called a *kiosk*, which serves as a computing facility for DTN users. The kiosk also provides a persistent data transfer facility, so users do not have to wait for a mobile router to show up.
- There are two types of end users, *mobile users*, who use their own personal devices to connect directly to routers (typically a kiosk), and *kiosk users*, who use a shared terminal at a kiosk. Our anonymous secure communication architecture mainly targets mobile users. However, if a kiosk is trusted, our architecture provides equivalent security and anonymity to kiosk users.

Achieving security and privacy in such disconnected networks is a demanding task, but it is necessary in hostile environments with malicious attackers or even just passive listeners. In rural area DTNs, security and privacy are necessary to effectively implement concepts like e-governance, citizen journalism [1], distance education and telemedicine. In a hostile environment, secure and anonymous DTN communication can provide an efficient way to let informers transfer information while hiding their identity from an enemy. Therefore, the utility of a DTN is greatly expanded when the DTN provides end-to-end security and privacy. The limitations of DTNs require the design of new security and privacy protocols for DTNs, which forms the basis for this work.

2.2 Identity-Based Cryptography (IBC)

Our security and anonymity solutions for DTNs are based on IBC. In particular, our solutions exploit the Sakai-Ohgishi-Kasahara key agreement scheme [24] and hierarchical identity-based encryption and signature schemes. We discuss these protocols in this section.

2.2.1 Sakai-Ohgishi-Kasahara (SOK) Key Agreement Scheme

The SOK key agreement scheme is based on the Boneh-Franklin identity-based encryption scheme (BF-IBE) [5]. In BF-IBE, a trusted authority, called the private key generator (PKG), generates a prime p and two groups \mathbb{G} (written additively) and \mathbb{G}_T (written multiplicatively) of order p such that an efficiently computable bilinear pairing $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ is known. We give a brief introduction to bilinear pairings in Appendix A. The PKG generates a random element $s \in \mathbb{Z}_p^*$, known as the PKG's master secret. (In BF-IBE, the PKG also generates public parameters P and $sP \in \mathbb{G}$, but these are not required for the SOK key agreement scheme.) After the system setup, the PKG computes private keys for its users based on their well-known identities (i.e., their public keys). A user with identity ID_i receives the private key $d_i = sH(\text{ID}_i) \in \mathbb{G}$, where $H : \{0, 1\}^* \rightarrow \mathbb{G}^*$ is a full-domain cryptographic hash function.

Sakai et al. [24] observe that, in a BF-IBE setup, two users belonging to the same PKG can non-interactively compute a shared key given the identity of the other participant and their own private key. For example, two users with identity/private key pairs (ID_U, d_U) and (ID_V, d_V) can independently compute the shared key

$$K_{UV} = e(Q_U, d_V) = e(d_U, Q_V) = e(Q_U, Q_V)^s,$$

where $Q_U = H(\text{ID}_U)$ and $Q_V = H(\text{ID}_V)$. Dupont and Enge [12] prove that this key agreement is secure in the random oracle model under the bilinear Diffie-Hellman assumption in $\langle \mathbb{G}, \mathbb{G}_T, e \rangle$ (see Appendix A).

Note that in all practical applications, the shared key K_{UV} should be *derived* from $e(Q_U, Q_V)^s$ (by hashing for example) instead of using the actual element in \mathbb{G}_T , but to aid exposition throughout this paper, we use $K_{UV} = e(Q_U, Q_V)^s$.

The SOK key agreement scheme provides mutual authentication using explicit key confirmation [29, Sec. 9.2] between two PKG users. It also offers a non-interactive, implicit key authentication mechanism. Here, a sender U transfers a message that is symmetrically encrypted with key K_{UV} to a receiver V . In this setup, sender U is assured that no one other than receiver V can compute key K_{UV} and, on successful decryption of the ciphertext, V is assured that the message was sent by U . Therefore, this non-interactive protocol simultaneously achieves message confidentiality and source authentication, though U is not assured that V actually received the ciphertext, in the absence of key confirmation.

In Section 4, we use these protocols for mutual authentication in DTNs. In Section 5, we modify the SOK key agreement scheme to define a new anonymous key agreement scheme and extend the above authentication schemes for anonymous users.

2.2.2 Hierarchical Identity-Based Cryptography (HIBC)

As the SOK key agreement scheme does not provide mechanisms for secure message transfer when the two participants do not belong to the same PKG, we need a separate mechanism for this case. Namely, we exploit identity-based encryption (IBE) for message confidentiality and identity-based signatures (IBS) for source authentication. For example, we could use BF-IBE [5] for encryption

and corresponding Cha-Cheon IBS [10] for signing. However, these schemes require that both participants have knowledge of the public parameters of the other participant's PKG. Consequently, these schemes lack scalability without a PKG hierarchy.

To provide a scalable architecture for IBE and IBS, we use hierarchical identity-based cryptography (HIBC). Here, users belong to *domain PKGs*, which are the leaf PKGs in a hierarchy tree of PKGs. To communicate with any user belonging to such a hierarchy, knowing that user's identity and the public key of the root PKG is necessary and sufficient, compared to knowing the public key of each PKG in non-hierarchical IBC. In HIBC, a user identity includes the identity of every PKG in the user's ancestry, which can be compared to naming in the Internet DNS hierarchy.

Numerous hierarchical IBE (HIBE) and hierarchical IBS (HIBS) schemes have been proposed in the literature [4, 6, 15, 16, 32, 33]. We can use any combination of a HIBE and a HIBS scheme for secure data transfer, provided that 1) the HIBE scheme requires only knowledge of the receiver identity and the public key of the root PKG for encryption, 2) the HIBS scheme requires only knowledge of the signer identity and the public key of the root PKG for verification and 3) all the operations are possible with the same set of public and private parameters. For example, the combination of Boneh et al.'s HIBE scheme [4] and Yuen and Wei's HIBS scheme [33] satisfies these requirements.

Note that the obvious approach of extending the SOK key agreement scheme, as used in the case where both participants belong to the same domain PKG (see Section 2.2.1), to the HIBC case, where the two participants belong to different domain PKGs, does not seem to be possible. Namely, the structure of the private keys in the HIBC schemes and the inclusion of randomness in these keys make extending the SOK key agreement scheme hard. Therefore, we use the combination of a HIBE and a HIBS scheme instead. We also note that, as a PKG can become a single point of failure, it can be made distributed with a secret sharing scheme.

3 Related Work

Seth and Keshav [25] address the challenges for secure communication in DTNs. They observe that the traditional PKI-based approach is not suitable for disconnected networks. In a PKI, a user authenticates another user's public key using a certificate signed by a certificate authority (CA). In disconnected DTNs, without online access to an arbitrary receiver's public key or certificate, sending an encrypted message on the fly is not possible. Further, PKIs implement key revocation based on frequently updated online certificate revocation lists (CRLs) posted by CAs. In the absence of instant online access to CRLs, a receiver cannot authenticate a sender's public key or certificate in a DTN. To overcome these problems, Seth and Keshav suggest the use of IBC, where the public key of each entity is replaced by its identity and associated public formatting policies. They use the Gentry-Silverberg HIBE and HIBS scheme [15] to achieve end-to-end security. Compared to the existing scheme, our solution has the following two advantages: First, our mutual authentication scheme is more efficient and can optionally be made non-interactive. Second, our secure data transfer mechanism for the *local* case (i.e., between users of same DTN) is more efficient. Furthermore, our contributions allow us to go further and to achieve anonymous communication in DTNs using pseudonyms.

In their recent book on DTNs, Farrell and Cahill investigate issues with the use of IBC for security in DTNs [14, Chap. 8]. They claim that IBC does not solve the key management problem in DTNs. Namely, they believe that it is difficult to identify a particular Bob out of the many in the world sharing this name. Unique identification is a traditional problem in communication of any form. In IBC-based DTNs, it can be solved by combining identities with geographic identifiers (e.g., place

name, state or country), like in the traditional postal system or telephone network. Further, they believe that IBC-based DTN security is not scalable as they assume that a user must know the public parameters for all the PKG servers. Using HIBC, it is possible to reduce the number of public parameters that users have to know significantly and any rare public parameter updates for root PKGs can be conveniently communicated to users along with their periodically rotated private key.

Though there are many existing solutions for anonymity in traditional networks, we found them unsuitable for use in DTNs. Namely, there are two main differences between DTNs and the Internet that affect anonymity systems. Due to the disconnected nature and “take what you can get” routing strategy of DTNs, a sender does not have the freedom to choose a traffic route and confirmation/feedback is difficult to obtain. With opportunistic connections and variable delays, source-routing is not always possible. Further, in DTNs like rural area networks, only a few special routers connect to DTN users. This limits the number of possible routes that traffic can take from one region to another. Onion routing approaches (such as Tor [11]) require knowledge of the network topology and are therefore immediately excluded. MIX-nets [19] are similar to onion routing networks, but have one or more MIXes which relay traffic in a “mixed” order. To be effective, MIX nodes must hold messages in order to build up the anonymity set. With opportunistic links, this additional delay magnifies the overall delay of communication. Also, the time required by MIX nodes depends on how many users are using the system at a given time. To overcome these limitations, we provide anonymity to DTN users with a pseudonym-based approach.

4 DTN Security Architecture

Seth and Keshav [25] present an HIBC-based end-to-end security architecture. Our solution is based on this architecture. In this section, we show how incorporating the SOK key agreement scheme presented in Section 2.2.1 results in a more efficient solution. We first present the threat model, followed by the system setup and user registration and finally discuss secure communication.

4.1 Threat Model

In DTNs, we expect rogue routers or unauthorized users to masquerade as valid routers/users. Further, malicious users of the system may try impersonating an honest DTN user or router. Passive adversaries can eavesdrop on the messages sent over DTN links. An active adversary can compromise some DTN routers or users, although this will eventually be detected.

We assume that the PKG and the DTN gateway are fully trusted by DTN users. As mentioned in Section 2.1, we expect users to have their own devices. If this is not the case, a user must fully trust a kiosk, since the kiosk will require access to the user’s private keys.

4.2 System Setup and User Registration

In our solution, we make a distinction between local and long distance communication, much like the telephone network. Each domain (regional) PKG in the IBE hierarchy has a limited “coverage area” corresponding to a DTN region. Each DTN user will have two private keys for her identity. The *long distance key* is a key for signing and decrypting messages from anyone knowing the public key of the root PKG. The *local key* is a key used for authentication with routers in the local DTN region and for secure communication with other users of the local PKG. As discussed in Section 2.2, the separate local and long distance infrastructures are required to obtain an efficient solution with all of the security and anonymity features. We use the term *service provider* for the entity

which is providing DTN service to a region. We now describe the system setup and user registration processes.

4.2.1 Local System Setup

A service provider working as the domain PKG for a DTN region performs a BF-IBE setup to generate system parameters $\mathbb{G}, \mathbb{G}_T, e$ and a master secret s and publishes $\langle \mathbb{G}, \mathbb{G}_T, e \rangle$. In this setup, using appropriate hash functions, a DTN user with a valid private key can perform mutual authentication and secure and authenticated message transfers with other nodes belonging to the same DTN using the SOK key agreement scheme and its extensions, as explained in Section 2.2.1.

4.2.2 Long Distance System Setup

Each domain PKG is a leaf node in a HIBC hierarchy. Users of such a domain PKG can securely communicate with any user in the hierarchy using the public parameters from the root PKG. In case there are multiple hierarchies, users can also communicate with users belonging to a different hierarchy using the public parameters for the root PKG of that hierarchy. This significantly reduces the number of public parameters users need to know, which aids scalability. In a HIBC setting, the domain PKG generates the long distance master key s_{LD} (called private key in some settings) and computes private keys for its users. We assume that all PKGs are well connected through the Internet.

4.2.3 User Registration

Similar to the cellphone network, when a new user signs up for a DTN service with a distribution agent, she is provided with a client software, local and long distance public PKG parameters, a unique subscription number (U_N) and a long term symmetric key (K_{U_N}) from her domain PKG. The domain PKG provides this information via the distribution agent in storage devices (such as USB keys) with tamper-resistant packaging to avoid any third party from tampering with the symmetric key or the parameters. The long term symmetric key is used for delivering a user's private key to the user, once the user has informed the PKG of her identity. The PKG keeps a record of $\langle U_N, K_{U_N} \rangle$ pairs. Our registration protocol modifies Seth and Keshav's user registration [25, Sec. 5.3]. We replace costly HIBS signatures with MACs based on the SOK key agreement scheme. Figure 2 presents our DTN user registration process.

After successful registration, the user U has two private keys: the local private key d_{U_L} and the long distance private key $d_{U_{LD}}$. The domain PKG also appends the user's identity to the corresponding $\langle U_N, K_{U_N} \rangle$ pair.

Mobile users roaming from one DTN region to another should acquire private keys in the new region. If this is not possible, public key based mutual authentication [29, Sec. 9.3] using HIBS can be used. However, this approach is computationally inefficient, as we find in Section 7.1, and anonymity is not possible.

4.3 Secure Communication

Secure communication in a DTN requires mutual authentication between two DTN nodes before initiating a data transfer. In this section, we discuss mutual authentication between two DTN nodes and mechanisms for secure end-to-end data transfer.

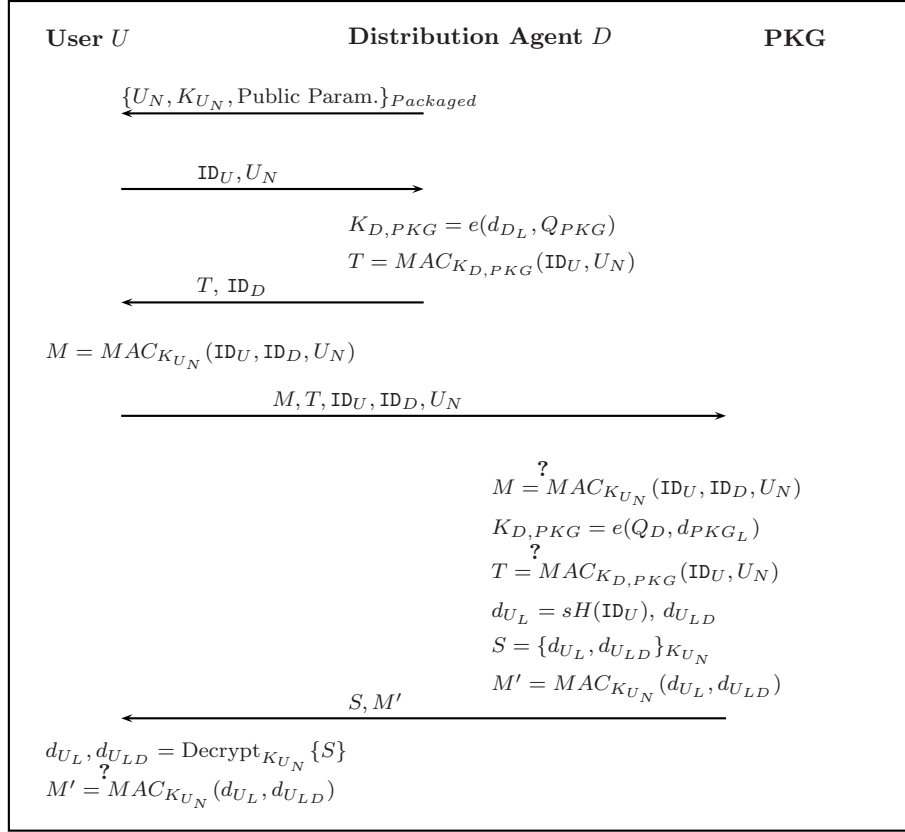


Figure 2: A user U registers with the PKG using a distribution agent D .

4.3.1 Mutual Authentication

In a DTN, when a registered user and a mobile or static router meet over an opportunistic link, they need to authenticate each other before transferring data. Seth and Keshav [25] use a mutual authentication protocol based on a HIBS scheme. Since many opportunistic links in DTN are time-constrained, we instead suggest use of the more efficient SOK key agreement scheme for this mutual authentication. As discussed in Section 2.2.1, this scheme can be performed in two ways. Based on the DTN environment and type of communication link, the interactive three-flow authentication scheme or the single-flow non-interactive authenticated key agreement scheme can be chosen. The non-interactive SOK key agreement scheme is suitable for DTNs where opportunistic links are highly time-constrained and interactive communication and involved online computation is not feasible. On the other hand, the interactive three-flow mutual authentication scheme is more suitable for a DTN where mutual authentication is mandatory to avoid denial of service attacks by malicious entities.

4.3.2 Local Data Transfer

For a receiver V belonging to the same DTN as a sender U , the SOK key provides message confidentiality, authentication and integrity. Here, the sender transfers her and the receiver's identities, a symmetric encryption $\{M\}_{K_{UV}}$ and a message authentication code $\text{MAC}_{K_{UV}}(M)$ to a DTN router.

$$C = \text{ID}_U, \text{ID}_V, \{M\}_{K_{UV}}, \text{MAC}_{K_{UV}}(M) \quad (1)$$

where M is a plaintext and $K_{UV} = e(Q_U, Q_V)^s$ with $Q_U = H(ID_U)$ and $Q_V = H(ID_V)$. The encryption $\{M\}_{K_{UV}}$ provides confidentiality and $MAC_{K_{UV}}(M)$ provides authentication and integrity. The receiver identity (ID_V) allows the gateway to route C to the receiver and the sender identity (ID_U) facilitates source authentication and computation of SOK key K_{UV} at the receiver end.

4.3.3 Long Distance Data Transfer

For a receiver V outside the DTN region of a sender U , she sign-then-encrypts the message with the HIBE and HIBS combination for the receiver identity ID_V using her long distance private key $d_{U_{LD}}$ as follows:

$$C = ID_U, ID_V, HIBE_V(M || HIBS_U(M || ID_V)). \quad (2)$$

An et al. [3] prove that this sign-then-encrypt method provides the same security guarantees as the individual encryption and signature schemes.

Both for local and long distance data transfer, the sender mutually authenticates with a router and transfers C . The router forwards this data towards a gateway over opportunistic links. The gateway, depending on the receiver's location, routes C to a kiosk in the same DTN (see section 7.4 for a discussion of routing in a DTN), to the gateway for another DTN using traditional Internet routing or to a receiver on the Internet, again using traditional Internet routing. (We assume that there are domain PKGs that hand out private keys to regular Internet users.) The gateway can learn information about the receiver's location from the receiver's ID (similar to DNS). If C is routed to another DTN gateway, this gateway will then route C to a kiosk in its DTN, which provides C to the receiver V on authentication.

Though two users of the same DTN could also transfer data using long distance data transfer, data transfer based on the non-interactive SOK key agreement scheme is more efficient. We compare the computation costs for our two secure data transfer schemes in Section 7.1.

4.4 Key Revocation

Key revocation for suspended users is an important issue in all public key infrastructures. For IBC, Boneh and Franklin [5] suggest attaching a validity period (say t) to a user identity (say ID_U), in some format defined by a public policy, while encrypting a message for the user. In such a case, a user U needs to obtain the private key for $\{ID_U || t\}$ at each validity period and the PKG can revoke access from her by withholding her private key $d_{U,t}$. For simplicity, throughout the paper, we assume that validity periods are attached to user identities, though we do not mention them explicitly.

5 DTN Anonymity Architecture

In this section, we discuss user anonymity in DTNs. As it turns out, we can build our DTN anonymity architecture based on our DTN security architecture with no changes in the setup. The goal of anonymous communication is to permit the types of communication required in the examples given in Section 2.1, such as citizen journalism. In these applications, the sender and receiver know each other's identity, but observers and network entities should not be able to determine the identity of a sender or receiver. In our solution, anonymity is provided by pseudonyms and by protocols which allow DTN routers to know the pseudonym belonging to a valid user without learning the user's identity. In this section, we start with the threat model and describe our DTN anonymity

architecture. In Section 6, we exploit our architecture for secure and anonymous communication in DTNs.

5.1 Threat Model

As DTN security is a basis for DTN anonymity, the threat model for anonymous communication includes the threat model for secure communication introduced in Section 4.1. Furthermore, DTN routers, including kiosks, must not be able to learn the identities of communicating DTN users. DTN gateways are trusted and can know these identities. In addition, we require that each kiosk services a number of users, which determines a user’s anonymity set. We assume physical anonymity at DTN routers, that is, they do not have any form of identification device, like a camera. Attackers may be able to perform traffic analysis, however this attack vector can only reveal with whom a particular pseudonym has communicated, but not the identity of the owner of the pseudonym. This is a primary difference between our threat model and the one used by systems that anonymize Internet traffic. We discuss this attack further in Section 7.3.

5.2 Anonymous Authentication

In anonymous authentication in a DTN, an anonymous user wants to confirm the identity of a DTN router and a router needs to be sure that the user is a valid user of its PKG. We achieve this by introducing a new unconditionally anonymous key agreement protocol for a BF-IBE setup (such as our local setup described in Section 4.2).

This protocol modifies the SOK key agreement scheme by replacing user identities with their pseudonyms. In this anonymous authentication, a participant can confirm that the other participant is a user of the same PKG, but cannot determine her identity, even after multiple interactions.

Suppose users U and V want to anonymously authenticate each other. User U , with local (identity, private key) pair (ID_U, d_{U_L}) , generates a random number r_U and computes a pseudonym and corresponding private key $(P_U = r_U Q_U = r_U H(ID_U), r_U d_{U_L} = sP_U)$. Similarly, user V generates a random number r_V and computes a (pseudonym, private key) pair $(P_V = r_V Q_V = r_V H(ID_V), r_V d_{V_L} = sP_V)$ for his pair (ID_V, d_{V_L}) . In the two-way anonymous authentication, U and V exchange their pseudonyms P_U and P_V , which enables them (and nobody else expect the PKG) to independently compute the session key

$$K_{UV} = e(sP_U, P_V) = e(P_U, sP_V) = e(P_U, P_V)^s.$$

We discuss security and anonymity for this anonymous key agreement scheme in Appendix B. In general, by replacing participant identities by their pseudonyms in any mutual authentication protocol (say [29, Sec. 9.2]), participants (U and V) can compute a key K_{UV} and perform anonymous mutual authentication.

Anonymous authentication in a DTN generally requires anonymity for only one of the two participants (a DTN user) and the other participant often works as a service provider (a DTN router). Therefore, we replace only the user’s identity with a pseudonym in our anonymous authentication protocol. Our protocol for one-way anonymous authentication in a DTN is illustrated in Figure 3.

5.3 Non-Interactive Anonymous Transfer

In some DTN situations, the opportunistic link between a user and a router may be time-constrained and any three-flow mutual authentication might be infeasible. In such cases, we describe how two DTN nodes can perform non-interactive single flow data transfer with one-way anonymity.

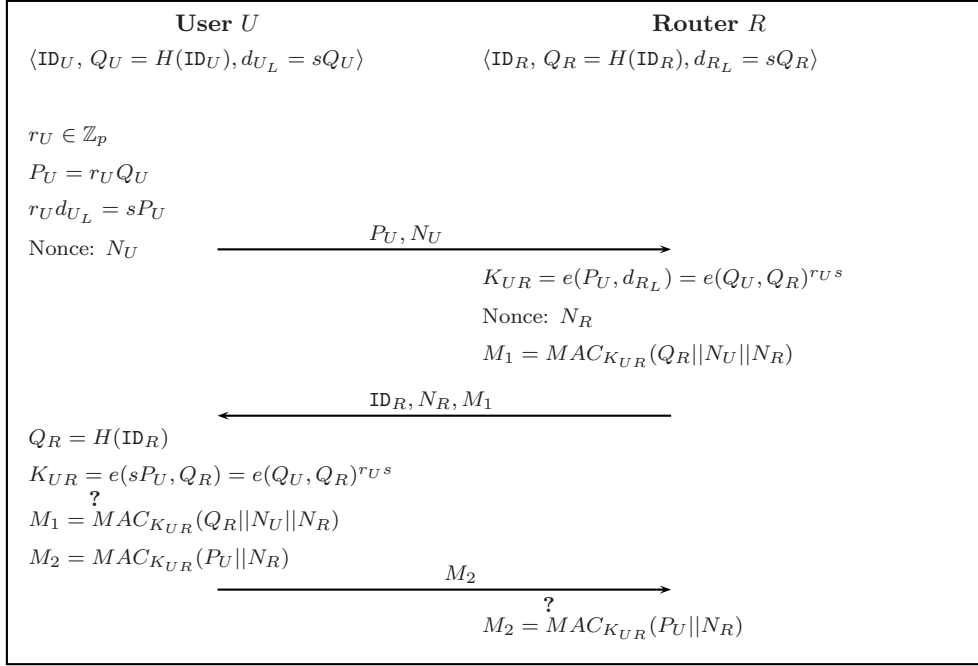


Figure 3: Anonymous authentication between a user U and a router R .

1. Assume a DTN user U with identity ID_U wishes to anonymously transfer data to a router R with identity ID_R .
2. The user chooses a random integer $r_U \in \mathbb{Z}_p^*$ and generates the corresponding pseudonym $P_U = r_U Q_U$ and private key $r_U d_{U_L} = sP_U$. She obtains $Q_R = H(\text{ID}_R)$ and calculates the session key $K_{UR} = e(sP_U, Q_R) = e(Q_U, Q_R)^{r_U s}$.
3. She then computes the symmetric encryption $\{M\}_{K_{UR}}$ and the $\text{MAC}_{K_{UR}}(M)$ of a message M and sends the tuple $\{M\}_{K_{UR}}, \text{MAC}_{K_{UR}}(M), P_U$ to the router R .
4. The router, using P_U and its private key d_{R_L} , computes the session key $K_{UR} = e(P_U, d_{R_L}) = e(Q_U, Q_R)^{s r_U}$. It then decrypts $\{M\}_{K_{UR}}$ to learn the message M and authenticates the message source verifying the MAC.

We discuss the performance advantages of this non-interactive protocol in Section 7.1.

In this protocol, the anonymous user U is assured that only router R can decrypt the message, while, on successful MAC verification, R is assured that a valid DTN user has sent the message M . Since explicit authentication does not occur in this protocol, an invalid malicious router can pretend to be R and drop messages. We discuss flooding-based routing strategies to address this attack in Section 7.4.

5.4 Receiver Anonymity From Ciphertext

In the majority of IBE schemes, given a ciphertext and a list of probable receiver identities, it is possible to determine the identities of the receiver. An adversary, such as a curious router, can use this property to determine the identity of the receiver. Therefore, we need receiver anonymity from a ciphertext. Out of the numerous IBE schemes defined, only BF-IBE [5], 2-HIBE [16] and Anonymous HIBE [6] have ciphertexts that provide receiver anonymity. Unfortunately, BF-IBE is

not hierarchical and thus not scalable to multiple DTN regions, 2-HIBE is only conditional collusion resistant and Anonymous HIBE is quite complex and does not have an associated HIBS scheme for source authentication. Therefore, we need an additional mechanism to protect receiver anonymity from a ciphertext.

The idea is to encrypt the ciphertext a second time with a symmetric encryption algorithm to hide the receiver's identity. The symmetric key is exchanged non-interactively between anonymous users and the gateway. The sender encrypts the message for the gateway, the gateway decrypts it and forwards it to the gateway in the receiver DTN. The transfer between this gateway and the receiver is encrypted in the same way.

We achieve this using the protocol for non-interactive anonymous transfer described above. In this anonymous data transfer, a sender U computes a pseudonym P_U , corresponding private key sP_U and a session key $K_{UG} = e(sP_U, Q_G)$ to communicate with a gateway G . She then encrypts the ciphertext destined for the receiver a second time, using K_{UG} , and performs a non-interactive anonymous message transfer with gateway G through intermediate routers. The gateway G computes $K_{UG} = e(P_U, d_{G_L})$, decrypts and authenticates the ciphertext and forwards it to the recipient (an Internet user or a DTN user) or another gateway. The receiver gateway uses the same mechanism to transfer the ciphertext to the receiver kiosk, using a default recipient pseudonym. We elaborate on the concept of default pseudonyms in Section 6.2.

5.5 Key Revocation

The key revocation problem is nontrivial in the anonymous setting as, given a pseudonym, the validity period of a user identity cannot be verified. In anonymous authentication, a verifying router can only assure that the local private key used by a user was provided by the PKG for some identity and some validity period, but cannot determine if this combination is currently valid.

The only feasible solution for this problem involves periodically changing the domain PKG's local master secret s , as this invalidates all older local private keys. Since the PKG's long distance (HIBC) public key is used for encryption and signature, a change in the local (BF-IBE) master secret of the PKG does not affect long distance communication.

Thus, in our anonymous communication system, the local master key s of a domain PKG changes periodically and users are accordingly provided with new local private keys. Since users' long distance private keys must be updated for their identities and validity periods, using the same validity period for both local and long distance keys updates can further simplify key management and revocation.

6 Secure and Anonymous Communication

In this section, we discuss secure and anonymous DTN communication using the anonymity architecture described in Sections 5. Anonymous communication in DTNs requires no changes to the setup needed for secure communication. Our DTN users can securely communicate with anybody inside or outside their DTN without revealing their identity to routers and observers. We divide anonymous DTN communication into three different categories.

- Sending messages anonymously,
- Receiving messages anonymously, and
- Anonymous Message Fetching (Sending + Receiving).

We next discuss each of these anonymous data transfers.

6.1 Anonymous Message Sending

In this anonymous data transfer, a DTN user wants to anonymously send a message to a receiver, for example an encrypted e-mail to an Internet user. The sender U achieves this as follows:

1. With her identity ID_U , she generates a pseudonym $P_U = r_U H(ID_U)$ and a corresponding private key $r_U d_{U_L} = s P_U$. She uses this pair to anonymously authenticate with a router R .
2. She computes C for a receiver V , as in (1) or (2) depending on the receiver's DTN region. Note that user U uses her real identity in this step.
3. She then encrypts the message C for the gateway G using the non-interactive anonymous message transfer mechanism with her session pseudonym P_U . The encrypted C is sent to the authenticated router R , which routes it to the gateway G along with the sender pseudonym P_U received during the authentication step. This hides the sender and receiver identities and C from DTN routers and adversaries.
4. The gateway G uses the pseudonym P_U and its private key d_{G_L} to generate $K_{UG} = e(P_U, d_{G_L}) = e(Q_U, Q_G)^{r_U s}$ and decrypts the received message to obtain ID_U , ID_V and C . The gateway then routes C back into its DTN network, in case the receiver is in the same DTN, or into the Internet, based on the receiver's location, as described in Section 4.3.

6.2 Anonymous Message Receiving

In this anonymous data transfer, a gateway wants to anonymously transfer a message to a DTN receiver because of a sender's request for anonymity or some policy specified by a receiver. None of the routers in the DTN should learn the identity of the receiver. The sender uses the receiver's identity and sends the message to the sender gateway using anonymous message sending, as discussed above. The sender and receiver gateways communicate over the Internet, without revealing sender and receiver identities, which is possible using existing security mechanisms (like SSL). Using the same mechanism, an Internet-based sender also directly communicates with the receiver gateway.

As the receiver's gateway cannot route a ciphertext in the DTN without a receiver identity, a user concerned about her privacy has to provide the gateway with a default pseudonym. When a message arrives addressed to a user with identity ID_V , the gateway automatically re-addresses it to her default pseudonym (say $P_{V_{Default}}$). Note that these default pseudonyms are random elements of \mathbb{G} obtained by taking random multiples of identity hashes and the collision probability is negligible. Anonymous message receiving works as follows:

1. The gateway G , on accepting a ciphertext C and the receiver identity ID_V , queries its database of $\{\text{User Identity, Default Pseudonym, Kiosk}\}$ tuples to obtain the default pseudonym $P_{V_{Default}}$ and the corresponding kiosk.
2. G then computes the key $K_{VG} = e(P_{V_{Default}}, d_{G_L})$ and encrypts C and ID_V again with key K_{VG} and routes the message along with the default pseudonym $P_{V_{Default}}$ to the receiver's kiosk.
3. The receiver V performs anonymous mutual authentication with the kiosk using her default pseudonym $P_{V_{Default}}$. On successful anonymous authentication, the kiosk is assured that the receiver has the private key for the pseudonym $P_{V_{Default}}$ and transfers the message over the authenticated channel. The kiosk is assured the message has been delivered to the true recipient and no longer needs to store it.

4. The receiver then computes the SOK session key $K_{VG} = e(sP_{V_{Default}}, Q_G)$ and decrypts the message to obtain C .
5. The receiver then decrypts C and verifies the message source using the attached MAC or HIBS signature.

Users should change their default pseudonym periodically by sending the new default pseudonym to the gateway in an encrypted message. Whenever the receiver updates her default pseudonym at the gateway, all the routing tables in the region may have to be modified. In Section 7.4, we suggest a routing based solution to this problem using the concept of flooding.

Although default pseudonyms do not divulge the real identity of a user, the default pseudonym can be used to track a message to its receiver kiosk in the receiver’s DTN and can leak information about the communication pattern of some DTN user. To avoid this attack, the gateway can allow the user to hide among all the users of the receiver kiosk by encrypting the message-pseudonym pair using the SOK key between the gateway and the receiver’s kiosk. In this case, on receiving the data, the kiosk decrypts it to obtain a message-pseudonym pair.

6.3 Anonymous Message Fetching (Sending + Receiving)

Anonymous message fetching is a combination of anonymous message sending and receiving. For example, a user U requests a file (say a news article) from the Internet. She generates a session pseudonym P_U and sends an anonymous request to the gateway using anonymous message sending. When a gateway receives a request, it retrieves the file, and returns it using the session pseudonym P_U in anonymous message receiving, as defined above. In this case, the default pseudonym is not required.

7 System and Network Issues

In this section, we discuss some of the critical system and network related issues for secure and anonymous communication in DTNs. We investigate performance, billing, routing and traffic analysis.

7.1 Performance

We compare the performance of our (anonymous) mutual authentication protocols with Seth and Keshav’s (non-anonymous) mutual authentication protocol [25]. We evaluate secure and authenticated end-to-end data transfer in both local and long distance settings and also describe the computational overheads incurred when providing anonymity.

For the computational comparisons, we consider the three most costly operations, namely, computing bilinear pairing, exponentiating in \mathbb{G} and exponentiating \mathbb{G}_T . Other, cheaper operations, like AES symmetric encryption, group operations in \mathbb{G} and \mathbb{G}_T and hashing are ignored for simplicity and clarity of the presentation. All computation timings were gathered on a 3.0 GHz Pentium D PC using the PBC pairing-based cryptography library [21]. We used the “type A” curve from PBC for our experiments. Details about this elliptic curve are available on the PBC website. These timings are applicable to rural area DTNs, where projects like “One laptop per child” [2] distribute devices with comparable computation power (now or in the near future) to individual users.

Table 1: Mutual Authentication between DTN nodes (Computation at each node)

| Operation | Time (ms) | Mutual authentication | | |
|----------------------------------|--------------|-----------------------|------------|-----------|
| | | S. & K. [25] | This paper | |
| | | | Secure | Anonymous |
| Pairing | 2.9 | $h + 1$ ^a | 1 | 1 |
| Exponentiation in \mathbb{G} | 1.5 | 2 | 0 | 0 |
| Exponentiation in \mathbb{G}_T | 0.2 | 0 | 0 | 0 |
| Total time (ms) | | $2.9h + 5.9$ | 2.9 | 2.9 |

^a h is the height of the hierarchy tree.

Table 2: Comparison Between long distance and local data transfer

| Operation | Time (ms) | Local | | Long Distance ^a | |
|----------------------------|--------------|--------|----------|----------------------------|----------|
| | | Sender | Receiver | Sender | Receiver |
| Pairing | 2.9 | 1 | 1 | 0 | 8 |
| Exponen. in \mathbb{G} | 1.5 | 0 | 0 | $h + 8$ ^b | 0 |
| Exponen. in \mathbb{G}_T | 0.2 | 0 | 0 | 1 | 1 |
| Total time (ms) | | 2.9 | 2.9 | $12.2 + 1.5h$ | 23.4 |

^a We consider the efficient HIBE scheme suggested by Boneh et al. [4] and the corresponding HIBS scheme by Yuen and Wei [33] for secure and authenticated long distance data transfer.

^b h is the height of the hierarchy tree.

7.1.1 Mutual Authentication

Mutual authentication between two DTN nodes is the most critical operation with respect to performance, since opportunistic communication links between DTN nodes are time-constrained. In Table 1, we compare the time required for mutual authentication in Seth and Keshav (“S. & K.”)’s protocol [25] with mutual authentication in our architecture.

We observe that our mutual authentication mechanism performs significantly better. Our mutual authentication protocol is approximately 4 times faster, assuming a modest two level hierarchy ($h = 2$). As our users can compute a pseudonym and the corresponding private key offline, anonymous and non-anonymous authentication protocols have equivalent computational time.

The non-interactive key agreement protocols (non-anonymous and anonymous) with key authentication require the same computations as mutual authentication. However, as they involve only one communication flow and all computation can be done off-line, they offer superior performance.

7.1.2 Long Distance and Local Secure Data Transfer

In Section 4, we suggest the use of separate local and long distance mechanisms for secure end-to-end data transfer. Though for simplicity, long distance communication based on HIBC can be used for both forms of communication, we advocate using the non-interactive SOK key agreement scheme for local communication, as we achieve significant savings in computational costs using the latter scheme. In Table 2, we find that local data transfer using the SOK key agreement scheme is approximately 8 times faster than data transfer using HIBE and HIBS. We also note that Seth and Keshav’s data transfer solution [25] based on Gentry-Silverberg HIBE [15] is far less efficient than

Boneh et al.'s HIBE scheme [4] and it does not provide source authentication.

7.1.3 Overhead for Anonymity

Our anonymous communication solution, which is built on our security architecture, does not incur any significant increase in computation and no increase in the communication cost. The only extra computation required for anonymity is in the form of a symmetric encryption of ciphertext C and a receiver identity using the SOK shared key K_{UG} (for the gateway identity ID_G and a user's pseudonym P_U). The user can always pre-compute K_{UG} . This certainly shows that our anonymous communication solution is efficient in terms of computational costs, which makes the practicality of our system apparent.

7.2 Billing

Certain DTNs (such as rural area DTNs) may require a mechanism to bill users for network access. Billing is easily possible with non-anonymous users. The router asks for a signed confirmation for every message transferred and later transfers these non-repudiable confirmations to the service provider for billing. When users can be anonymous, billing becomes more challenging.

In this section, we describe how payments for DTN network access can be made with electronic cash. This allows users to pay for usage anonymously, balancing privacy against business needs. This model is similar to calling cards in the phone industry. Since electronic cash systems with the required features exist in the literature, we will not give the details here. Instead we refer readers to Law et al. [20] and Brands [7].

7.2.1 Sender Billing

In a DTN, an e-cash payment system cannot be online, that is, the e-cash cannot be validated at the time of payment. In particular, when a mobile router receives a payment from a sender, it will not have access to a network connection that would allow it to validate a coin presented by the sender. Using an offline payment system, the router can accept the e-cash and validate it later. As it turns out, the DTN payment system is not completely offline; once the message reaches the Internet with the "postage" attached, the gateway can at that point verify that the coin is valid. If it is invalid, the data can be dropped or returned to the user.

Recall that the gateway is controlled by the service provider. Our DTN payment system works as follows:

- Users buy electronic cash from the service provider. The electronic cash scheme requires that they give the service provider a share of their real identity to detect double spending.
- They use the e-cash when sending data. The mobile router verifies (as best it can) that the e-cash is valid and that the amount is sufficient.
- The gateway re-checks the e-cash before forwarding the data. In particular, the gateway can identify users if they attempt to double spend the same coin. If the gateway detects misbehaviour by a user, it will drop the user's data.

7.2.2 Receiver Billing

In a pay-by-usage system, users must pay when receiving data, as well as when sending data. A similar situation exists in cellphone networks, where phone owners must pay for incoming calls.

To help their users decide whether to accept a call and incur charges, cellphone providers send information about the caller (typically their name and number). DTN providers can do something similar when the sender is not anonymous. Here, the recipient may view the sender's identity or application specific information, such as email headers. When the sender is anonymous, there is no information other than the size of the data that the service provider can offer.

7.3 Traffic Analysis and Mixing

With the threat models of Sections 4 and 5, it may be possible for attackers to compromise some of the DTN routers. Suppose a user is communicating anonymously using a pseudonym. Should the attacker compromise all of the routers in the path, he will be able to link traffic to a particular pseudonym. In this worst-case scenario, the anonymity of the user rests solely on the attacker's inability to link a pseudonym to the user identity.

An attacker might try and link the traffic between a pseudonym and recipients on the Internet. In this way he could build a profile of communications involving the particular pseudonym. The user has the ability to generate pseudonyms easily, she should change them frequently. Users should also periodically change their default pseudonym (via an encrypted message to the gateway), as discussed in section 6.2.

Should the attacker not have complete control of the path, linking traffic to pseudonyms will be difficult. Due to the way traffic moves through a DTN, there is a certain amount of "natural" mixing that occurs. Since the link to the next hop is opportunistic, routers buffer the messages and send them out in groups to the next router(s). Similar behaviour can be found at the gateway, which receives groups of messages which it sends out to recipients on the Internet in mixed order.

A few other factors frustrate traffic analysis. Links between routers/gateway are encrypted, ephemeral and geographically located. The attacker must physically be in the right place at the right time in order to observe traffic between routers.

7.4 Routing in DTN

There have been a number of routing protocols suggested for various DTN types [17,18,23,26]. Our anonymous and secure communication architecture is generic in nature and should work with any routing strategy, as it only assumes that a message sent by a user reaches the gateway and a message forwarded by the gateway with some routing information reaches the corresponding receiver. Here, we briefly discuss location management and routing at the gateway and the advantages of flooding for anonymity in DTNs.

7.4.1 Location Management and Routing

We assume that the gateway knows a way to transfer a message to a user in its domain. Thus, our gateway maintains information about a user's identity, a default pseudonym and a corresponding kiosk. It keeps tuples of the form {User Identity, Default Pseudonym, Kiosk} for all of its users and utilizes them to route a message to a user based on their identity or a default pseudonym (as required). In anonymous message fetching, where session pseudonyms are used, reverse path forwarding [26, Sec. 8] can be used.

7.4.2 Flooding in DTN

In many DTNs, due to the unreliability of a single message reaching its destination, flooding messages to certain parts of a DTN can improve reliability. For DTN users, using non-interactive

message transfer with key authentication, flooding also improves the chance that a message actually reaches a gateway. In the opposite direction, the gateway can flood a message in part of the DTN, so as to improve the probability the recipient actually receives it. Though such flooding involves a lot more communication cost and storage space, it certainly aids for anonymity. It makes the adversaries job to locate a receiver a far more difficult as pseudonyms can not be linked to a specific kiosk of a DTN sub-region.

8 Conclusion

Security and anonymity are critical in many DTN implementations. Due to the unique disconnected nature of DTNs, traditional security solutions based on public key infrastructure are not suitable for these emerging networks. For the same reason, existing anonymity solutions are not directly applicable, either. In this paper, we presented an anonymous and secure communication architecture for DTNs using identity-based cryptography (IBC). We defined a new pseudonym-based anonymous authentication scheme in IBC and utilized it for DTN anonymity. We addressed receiver anonymity from ciphertext and key revocation and proposed feasible solutions. We also discussed system and network issues, like performance, traffic analysis and routing. Further, we investigated billing of anonymous users and explained how this can be accomplished with an electronic cash system. Our practical solution to DTN security and anonymity will be advantageous in many DTN situations.

In terms of future work, we note that a solution for the open problem of key agreement in hierarchical IBC will also enable a simpler secure and anonymous architecture for DTNs. In more general, with the continued research interest in IBC, more efficient solutions for DTN security and anonymity might become possible in the future.

Acknowledgments

We are grateful to Aaditeshwar Seth for motivating us towards anonymity in DTNs and for precious comments on the preliminary version of the paper. We also thank Ian Goldberg for suggesting electronic cash as a solution for DTN billing.

References

- [1] Citizen Journalism. Wikipedia, http://en.wikipedia.org/wiki/Citizen_journalism. Accessed March 2007.
- [2] One Laptop per Child (OLPC) Foundation. <http://www.laptopfoundation.org>. Accessed March 2007.
- [3] J. H. An, Y. Dodis, and T. Rabin. On the security of joint signature and encryption. In *EUROCRYPT '02*, pages 83–107, 2002.
- [4] D. Boneh, X. Boyen, and E. Goh. Hierarchical identity based encryption with constant size ciphertext. In *Advances in Cryptology—EUROCRYPT 2005*, volume 3494 of *Lecture Notes in Computer Science*, pages 440–456. Berlin: Springer-Verlag, 2005.
- [5] D. Boneh and M. K. Franklin. Identity-based encryption from the Weil pairing. In *CRYPTO '01: Proceedings of the 21st Annual International Cryptology Conference on Advances in Cryptology*, pages 213–229, London, UK, 2001. Springer-Verlag.

- [6] X. Boyen and B. Waters. Anonymous hierarchical identity-based encryption (without random oracles). In *Advances in Cryptology CRYPTO '06*, pages 290–307, 2006.
- [7] S. Brands. Untraceable off-line cash in wallet with observers. In *CRYPTO '93: Proceedings of the 13th annual international cryptology conference on Advances in cryptology*, pages 302–318, New York, NY, USA, 1994. Springer-Verlag New York, Inc.
- [8] E. Brewer, M. Demmer, B. Du, M. Ho, M. Kam, S. Nedevski, J. Pal, R. Patra, S. Surana, and K. Fall. The case for technology in developing regions. *IEEE Computer*, 38(6):25–38, 2005.
- [9] S. Burleigh, A. Hooke, L. Torgerson, K. Fall, V. Cerf, B. Durst, K. Scott, and H. Weiss. Delay-tolerant networking: an approach to interplanetary Internet. *IEEE Communications Magazine*, 41(6):128–136, 2003.
- [10] J. Cha and J. Cheon. An identity-based signature from gap Diffie-Hellman groups. In *Public Key Cryptography*, pages 18–30, 2003.
- [11] R. Dingledine, N. Mathewson, and P. Syverson. Tor: The second-generation onion router. In *Proceedings of the 13th USENIX Security Symposium*, August 2004.
- [12] R. Dupont and A. Enge. Provably secure non-interactive key distribution based on pairings. *Discrete Applied Mathematics*, 154(2):270–276, 2006.
- [13] K. Fall. A delay-tolerant network architecture for challenged internets. In *SIGCOMM '03: Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 27–34. ACM Press, 2003.
- [14] S. Farrell and V. Cahill. *Delay- and Disruption- Tolerant Networking*. Artech house, Boston, London, 2006.
- [15] C. Gentry and A. Silverberg. Hierarchical ID-based cryptography. In *ASIACRYPT '02*, pages 548–566. Springer-Verlag, 2002.
- [16] J. Horwitz and B. Lynn. Toward hierarchical identity-based encryption. In *EUROCRYPT '02*, pages 466–481, London, UK, 2002. Springer-Verlag.
- [17] S. Jain, K. Fall, and R. Patra. Routing in a delay tolerant network. In *SIGCOMM '04*, pages 145–158, 2004.
- [18] E. Jones, L. Li, and P. Ward. Practical routing in delay-tolerant networks. In *SIGCOMM 2005 Workshop on Delay Tolerant Networking (WDTN-05)*, 2005.
- [19] D. Kesdogan, J. Egner, and R. Büschkes. Stop-and-go MIXes: Providing probabilistic anonymity in an open system. In *Proceedings of Information Hiding Workshop (IH 1998)*. Springer-Verlag, LNCS 1525, 1998.
- [20] J. Solinas L. Law, S. Sabett. How to make a mint: the cryptography of anonymous electronic cash. *Unpublished NSA Technical Report*, June 1996. <http://jya.com/nsamint.htm>.
- [21] B. Lynn. PBC Library – The Pairing-Based Cryptography Library. <http://crypto.stanford.edu/pbc/>. Accessed February 2007.

- [22] J. Ott and D. Kutscher. Drive-thru Internet: IEEE 802.11b for “automobile” users. In *INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies*, 2004.
- [23] A. Pentland, R. Fletcher, and A. Hasson. Daknet: rethinking connectivity in developing nations. *IEEE Computer*, 37(1):78–83, 2004.
- [24] R. Sakai, K. Ohgishi, and M. Kasahara. Cryptosystems based on pairing. In *SCIS 2000: Symposium on Cryptography and Information Security*, 2000.
- [25] A. Seth and S. Keshav. Practical security for disconnected nodes. In *1st IEEE ICNP Workshop on Secure Network Protocols, 2005. (NPSec)*, pages 31–36, 2005.
- [26] A. Seth, D. Kroeker, M. Zaharia, S. Guo, and S. Keshav. Low-cost communication for rural Internet kiosks using mechanical backhaul. In *Proc. MOBICOM 2006*, September 2006.
- [27] A. Shamir. Identity-based cryptosystems and signature schemes. In *Proceedings of CRYPTO 84 on Advances in cryptology*, pages 47–53, New York, NY, USA, 1985. Springer-Verlag New York, Inc.
- [28] T. Small and Z. Haas. The shared wireless infostation model: a new ad hoc networking paradigm (or where there is a whale, there is a way). In *MobiHoc '03*, pages 233–244, 2003.
- [29] D. R. Stinson. *Cryptography: Theory and Practice*. CRC Press, Inc., 3rd edition, 2005.
- [30] E. Verheul. Evidence that XTR is more secure than supersingular elliptic curve cryptosystems. In *Advances in Cryptology—Eurocrypt 2001, Lecture Notes in Computer Science 2045*, pages 195–210, 2001.
- [31] R. Y. Wang, S. Sobti, N. Garg, E. Ziskind, J. Lai, and A. Krishnamurthy. Turning the postal system into a generic digital communication mechanism. *SIGCOMM Comput. Commun. Rev.*, 34(4):159–166, 2004.
- [32] B. Waters. Efficient identity-based encryption without random oracles. In *EUROCRYPT '05*, pages 114–127, 2005.
- [33] T. Yuen and V. Wei. Constant-size hierarchical identity-based signature/signcryption without random oracles. *Cryptology ePrint archive: <http://eprint.iacr.org/2005/412>*, 2005.

A Bilinear Pairings

Consider two additive cyclic groups \mathbb{G} and $\hat{\mathbb{G}}$ and a multiplicative cyclic group \mathbb{G}_T , all of the same prime order p . A bilinear map e is a map $e : \mathbb{G} \times \hat{\mathbb{G}} \rightarrow \mathbb{G}_T$ with following properties.

1. **Bilinearity:** For all $P \in \mathbb{G}$, $Q \in \hat{\mathbb{G}}$ and $a, b \in \mathbb{Z}_p$, $e(aP, bQ) = e(P, Q)^{ab}$.
2. **Non-degeneracy:** The map does not send all pairs in $\mathbb{G} \times \hat{\mathbb{G}}$ to the unity in \mathbb{G}_T .
3. **Computable:** There is an efficient algorithm to compute $e(P, Q)$ for any $P \in \mathbb{G}$ and $Q \in \hat{\mathbb{G}}$.

Most of the pairing-based cryptographic protocols use a special form of bilinear map called a *symmetric pairing* which has $\mathbb{G} = \hat{\mathbb{G}}$. For such pairings $e(P, Q) = e(Q, P)$ for any $P, Q \in \mathbb{G}$. The modified Weil pairing over elliptic curve groups [30] is an example of a symmetric bilinear pairing.

A.1 The Bilinear Diffie-Hellman Assumption

The *bilinear Diffie-Hellman* (BDH) problem is to compute $e(P, P)^{abc} \in \mathbb{G}_T$ given a generator P of \mathbb{G} and elements aP, bP, cP for $a, b, c \in \mathbb{Z}_p^*$. An equivalent formulation of the problem, due to the bilinearity of the map, is to compute $e(A, B)^c$ given a generator P of \mathbb{G} , and elements A, B and cP . If there is no efficient algorithm to solve the BDH problem for $\langle \mathbb{G}, \mathbb{G}_T, e \rangle$, they are considered to satisfy the *BDH assumption*.

B Security and Anonymity of Our Key Agreement

In this section, we discuss the security and anonymity of our key agreement schemes in the random oracle model. We make following claims:

1. **Unconditional Anonymity:** It is impossible for the other participant, the PKG or any third party to learn the identity of an anonymous participant in a protocol run.
2. **No Impersonation:** It is infeasible for a malicious user of the PKG to impersonate another (non-anonymous) user in a protocol run. It is also infeasible for an adversary, who is not a user of the PKG, to impersonate as a valid anonymous user.
3. **Session Key Secrecy:** It is infeasible for anyone other than the two participants or the PKG to determine a session key generated during a protocol run.

Next, we discuss each of our claims in detail.

B.1 Unconditional Anonymity

For an anonymous user U with identity ID_U , the pseudonym $P_U = r_U Q_U \in \mathbb{G}$ is the only parameter exchanged during the protocol that is derived from her identity. Because \mathbb{G} is a cyclic group of prime order, multiplying by the random r_U perfectly blinds the underlying identity.

B.2 No Impersonation

We made two claims about impersonation, one related to an anonymous participant and other related to a non-anonymous participant.

Non-anonymous Participant Suppose an adversarial user with ID_{Adv} wishes to impersonate a non-anonymous participant (say, user V with ID_V) while communicating with an anonymous user U with pseudonym P_U . The adversary would need to compute $K_{UV} = e(P_U, Q_V)^s$ given P_U, Q_V, Q_{Adv} and sQ_{Adv} . But this is just the BDH problem, so under the BDH assumption on $\langle \mathbb{G}, \mathbb{G}_T, e \rangle$, impersonation of other users is infeasible.

Anonymous Participant The mutual authentication protocol requires anonymous participants have a private key from the PKG. We would like to be sure that a non-user of the PKG cannot compute a valid private key, ensuring only valid users of the PKG will be authenticated. Assuming that the master key s is secret and the discrete log problem over \mathbb{G} is hard, it is not feasible for an adversary to generate/obtain a pair $(P_{Adv}, d_{Adv} = sP_{Adv})$ of pseudonym and corresponding private key. Without a P_{Adv}, sP_{Adv} pair, an adversary can not compute a session key to impersonate a valid user in any secure mutual authentication scheme.

B.3 Session Key Secrecy

Dupont and Enge [12] prove the security of the key agreement scheme of Sakai et al. in the random oracle model. According to this proof, an attacker cannot compute the shared key if the BDH assumption holds on $\langle \mathbb{G}, \mathbb{G}_T, e \rangle$, and H is modelled by a random oracle. Our protocol simply modifies that of Sakai et al. to use $P_U = H'(\text{ID}_U)$ instead of $Q_U = H(\text{ID}_U)$, where $H'(x) = r_U \cdot H(x)$ for a random value r_U , so Dupont and Enge's proof of security applies to our protocol as well.