AFRL-RI-RS-TR-2012-104

EXPLOITING PAIRING-BASED ZERO KNOWLEDGE PROOF (ZKP) FOR TACTICAL NETWORK AUTHENTICATION

ILLINOIS INSTITUTE OF TECHNOLOGY

MARCH 2012

FINALTECHNICAL REPORT

AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE

■ **AIR FORCE MATERIEL COMMAND** ■ **UNITED STATES AIR FORCE** ■ **ROME, NY 13441**

NOTICE AND SIGNATURE PAGE

Using Government drawings, specifications, or other data included in this document for any purpose other than Government procurement does not in any way obligate the U.S. Government. The fact that the Government formulated or supplied the drawings, specifications, or other data does not license the holder or any other person or corporation; or convey any rights or permission to manufacture, use, or sell any patented invention that may relate to them.

This report was cleared for public release by the 88th ABW, Wright-Patterson AFB Public Affairs Office and is available to the general public, including foreign nationals. Copies may be obtained from the Defense Technical Information Center (DTIC) (http://www.dtic.mil).

AFRL-RI-RS-TR-2012-104 HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION IN ACCORDANCE WITH ASSIGNED DISTRIBUTION STATEMENT.

FOR THE DIRECTOR:

/s/
THOMAS SCATKO
Work Unit Manager

/s/
PAUL ANTONIK, Technical Advisor
Computing and Communications Division
Information Directorate

This report is published in the interest of scientific and technical information exchange, and its publication does not constitute the Government's approval or disapproval of its ideas or findings.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
MAR 2012	Final Technical Report	AUG 2011 – SEP 2011

4. TITLE AND SUBTITLE

EXPLOITING PAIRING-BASED ZERO KNOWLEDGE PROOF (ZKP) FOR TACTICAL NETWORK AUTHENTICATION

5a. CONTRACT NUMBER
FA8750-11-1-0109

5b. GRANT NUMBER
N/A

5c. PROGRAM ELEMENT NUMBER
62788F

6. AUTHOR(S)

Kui Ren

5d. PROJECT NUMBER
G2TA

5e. TASK NUMBER
GR

5f. WORK UNIT NUMBER
NT

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Illinois Institute of Technology
Department of Electrical and Computer Engineering
3300 S. Federal Street
Chicago, Il 60616

8. PERFORMING ORGANIZATION REPORT NUMBER
N/A

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Air Force Research Laboratory/RITE
525 Brooks Road
Rome NY 13441-4505

10. SPONSOR/MONITOR'S ACRONYM(S)
AFRL/RI

11. SPONSORING/MONITORING AGENCY REPORT NUMBER
AFRL-RI-RS-TR-2012-104

12. DISTRIBUTION AVAILABILITY STATEMENT
Approved for Public Release; Distribution Unlimited. PA# 88 ABW-2012-1276
Date Cleared: 12 MAR 2012

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Zero- knowledge proofs (ZKPs) are protocols that enable a prover to convince a verifier of the truth of a statement without leaking any other information. The main properties of ZKP include completeness, soundness and zero-knowledge. These features are correlat3ed with each other, and together with the lighter computational requirements, makes zero-knowledge protocols very attractive in authentication service in airborne networks. Although useful, the non-interactive zero-knowledge proofs based on standard cryptographic assumptions used to be inefficient and not useful in practice. The use of pairing-based ZKP on elliptic curves can potentially enhance the security strength of the system. Besides these advantages, pairing-based ZKPs can also integrate smoothly with other pairing-based cryptographic schemes (e.g., identity-based encryption, pairing-based signatures, key agreement, and proxy re-encryption) making the combined schemes quite efficient.

15. SUBJECT TERMS

zero-knowledge proofs, elliptic curve group pairing, network authentication

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			THOMAS SCATKO
U	U	U	UU	22	19b. TELEPHONE NUMBER (*Include area code*) N/A

TABLE OF CONTENTS

LIST OF TABLES

1. SUMMARY

This Final Technical Report summarizes the major research activities and findings for the performance period of the research Grant: August 01, 2011 to September 30, 2011. The work undertaken involved the investigation of applying Elliptic Curve group pairing-based Zero-Knowledge Protocols ZKP to the problem of authentication in tactical airborne networks.

2. INTRODUCTION

The Airborne Network (AN) uses a heterogeneous set of physical links to interconnect space, terrestrial and highly mobile airborne platforms. Acting as a wireless backbone, the airborne network delivers information to wireless units in an on-demand manner, and such information is periodically updated by satellites, ground base stations and wireless tactical units. From the network perspective, given the dynamically changing topology and the bandwidth-limited channel conditions, it will be critical to develop effective link access protocols and routing protocols which can accommodate the unique characteristics of the Airborne Network.

The research along this direction in mobile ad hoc networks received much attention, and many efficient and viable solutions have been proposed [1–5]. From the service perspective, as a large amount of sensitive data is stored in the wireless backbone, data security naturally becomes a big concern. For instance, data reported by military satellites and reconnaissance aircraft are closely related to privacy issues and should be accessible only to authorized units/users. Moreover, in hostile environments, attackers may successfully breach the network defenses and cause damage to the ANs to a certain extent, e.g., the attackers may disrupt the central authority (CA), on which the entities in ANs rely to authenticate each other for data access functionality, and/or the attackers may deplete the energy reserves of the nodes and render them dysfunctional.

These considerations motivate us to develop more robust and efficient authentication and access control mechanisms in airborne networks.

3. METHODS, ASSUMPTIONS and PROCEDURES

In symmetric key cryptography (SKC) based approaches, authentication is realized by using the same key for authentication, and data encryption/decryption. However, it is necessary to know all network members a priori and pre-configure a secret key for every pair of wireless nodes, which leads to scalability problems.

3.1 SKC & PKC Based Approaches for Authentication in Airborne Networks

Compared to SKC-based solutions, public key cryptography (PKC) based approaches use digital signatures for authentication, and can provide better data access security by encrypting data with a recipient's public key, where the compromise of the storage units will not jeopardize the data security. However, the common approach of using a single certificate authority (CA) for certifying every public key is impractical given a large number of users. Actually, when using the public key infrastructure (PKI) for authentication, it is required that all entities obtain a PKI certificate from a CA, in order to access the system functions (e.g., data access).

In the authentication process, the verifier node first uses the CA's public key to verify the prover node's certificate, and then uses the prover node's public key to verify its identity. This solution has several drawbacks:

(1) The amount of calculation is very large. Due to the lack of computing power or energy after attacks, the network nodes may not be able to afford such overhead.

(2) It creates a single point of failure. The CA could become the target of attacks, and if it fails the entire authentication system will stop working.

(3) It requires a three-way exchange of information among the verifier node, prover node and the CA. If the CA is not available after attack, the nodes that do not have the public key of the CA cannot perform authentication.

For these reasons, there is a critical need for more lightweight, efficient and robust authentication mechanisms that can accommodate the unique characteristics of the airborne network and have a dynamic capability to survive or recover from malicious attacks.

3.2 Use of ZKP based on the Factoring Problem Over Zn

In cryptography, Zero-Knowledge Protocols (ZKP) allow identification, key exchange, and other basic cryptographic operations to be implemented without leaking any user-identity information during the conversation and with smaller computational requirements than using comparable public key protocols. Specifically, a ZKP has the following properties:

(1) The verifier cannot learn anything from the protocol. That is, no knowledge is transferred.

(2) The prover cannot cheat the verifier. That is, without knowing the verifier's cryptographically derived credentials i.e., private (key) information, the prover can only succeed with a great amount of good luck.

(3) The verifier cannot cheat the prover. That is, the only thing the verifier can do is to convince himself that the prover knows the private (key) information.

These features are correlated with each other, and together with the lighter computational requirements, make zero-knowledge protocols very attractive in authentication services in airborne networks. In this section, we use the Guillou-Quisquater (GQ) protocol [6] to demonstrate the application of ZKP in authentication services.

The key idea of anonymous communication is to provide privacy among a group of participants, each of which is allocated a share of a secret. In a typical (k, n)-threshold secret sharing scheme, any combination of less than k secret shares reveals no information regarding the secret. On the other hand, no more than k out of n shares are required to fully recover the secret. Obviously, this property can be used to enhance the system dependability. In the proposed scheme, we use the verifiable secret sharing (VSS) [7] scheme to detect the invalid or corrupted shares due to attacks.

Bootstrapping Phase:

1. A trusted authority T selects two random primes p and q and forms a modulus $n = p \bullet q$.
2. T defines a public exponent $v > 4$ with $\gcd(v, (p-1)(q-1)) = 1$ so that T can compute $s = v - 1 \bmod 4(p-1)(q-1)$.
3. T publishes parameters n and v.

4. A node A has a unique identification Id(A). Everyone can calculate a value $J(A) = f(\mathrm{Id}(A)) \bmod n$ (the redundant identity).

5. T gives to *node A* the secret data where secret(A) = $J(A)^{-s}$, which it can calculate.

This is one time setup process. The secret s is used to generate secret(A) for *node A* authentication.

Secret Distribution:

1. A cyclic group G of prime order p′ with a generator g of G is chosen publicly as a system parameter, where p' satisfies $s < p'$.
2. T generates shares $P(1), \ldots, P(n)$ and distributes to each node one value, where $P(x) = s + a_1 x + \ldots + a_t x^t$ is a random polynomial of degree t and $P(0) = s$ is the original secret.
3. T computes and publishes commitments to the coefficients of P, i.e., $c_0 = g^s$, $c_1 = g^{a1}$, . . . , $c_t = g^{at}$.

Authentication Phase:

Node A proves his identity to node B using t rounds, each of which consists of:

1. A selects a random secret r and sends his identity Id(A) and $x = rv \bmod n$ to B.
2. B selects a random challenge e in {1, 2, ..., v}.
3. A computes and sends the following response to B: $y = r \cdot \mathrm{secret}(A)^{\mathrm{e}} \bmod n$.
4. B receives y, constructs $J(A) = f(\mathrm{Id}(A)) \bmod n$, computes $z = J(A)^{\mathrm{e}}\, y^v$, and accepts this round if $z = x \bmod n$.

System Recovery Phase:

Even if T is not available, the nodes that survive attacks can reconstruct the authentication system. Assume a node C initiates the recovery process.

1. C uses the authentication protocol to verify the validity of its neighbor nodes. C should also authenticate itself to others.

2. If authentication succeeds, each node further verifies the integrity of its own share: node i that holds $u = P(i)$ checks if the following equation holds:

$$\mathrm{g}^{\mathrm{u}} = c_0 c_1^{i^1} \ldots c_{\mathrm{t}}^{i^t} = \prod_{j=0}^{t} \mathrm{g}^{a_j i^j} = \mathrm{g}^{P(i)}. \qquad (1)$$

3. If $k-1$ nodes pass the authentication and integrity verification processes, C generates a temporary public/private key pair (k_{pub} , k_{pri}). C broadcasts the request for secret shares and the self-generated public key k_{pub} to the verified nodes.

4. Each node encrypts its stored secret share using C's temporary public key k_{pub} and sends to C.

5. C decrypts the received shares. After retrieving (i, $P(i)$) of all i in B (B is the set of the $k-1$ share holders), C applies the Lagrange interpolation formula to reconstruct:

$$s = \sum_{i \in B} b_i \, P(i), \text{ where } \; b_i = \prod_{j \in B, j \neq i} \frac{j}{j-i} \; . \qquad (2)$$

Analysis. The bootstrapping and secret distributions are both one-time setup processes. T would no longer be involved in the system after the private keys have been issued. In the authentication process, all legitimate entities obtain a unique pair ($J(A)$,secret(A)). The verifier B offers a hard problem (i.e., factoring extremely large numbers). Only if the prover A knows the solution to the hard mathematical problem can he be able to provide any of the solutions asked for. The proposed scheme has the following properties:

1. Compared to public key based protocols, the proposed protocol does not need to validate the certificate before communication. Furthermore, the authentication protocol can achieve the same results with approximately one to two orders of magnitude less computing power [8]. That implies the authentication can still work even with limited energy after certain energy depletion attacks. Thus, efficiency is achieved.
2. This is a distributed protocol. The centralized authority T is not involved in the system after the boot-strapping process. The nodes can authenticate each other solely based on their ($J(A)$,secret(A)) pairs, without interacting with T . Moreover, the coalition of a certain number of nodes can reconstruct the secrets and collaboratively act as T, so as to configure and generate authentication information for the newly joined nodes. By using VSS the individual nodes can also verify the integrity of their stored secret shares before secret reconstruction. Hence, both the system availability and dependability are enhanced compared to a centralized approach.

Such efficient and robust mechanisms will greatly enhance the data security/privacy of current versions of airborne networks.

4. RESULTS AND DISCUSSIONS

4.1 A Proposed Solution Using ZKP based on Pairing-based Cryptography

Recently, it has been shown that the practically efficient constructions of ZKPs that are based on standard intractability assumptions come from pairing based-cryptography [9–12], where a pairing is used between elements of two cryptographic groups to a third group to construct cryptographic systems. If elliptic curve (EC) groups [13, 14] are utilized as the cryptographic groups, solving the discrete log problem over elliptic curves takes exponential time as compared to ZKP protocols where the security of the system usually relies on the hardness of the discrete logarithm problem over sub-exponential solving times [15].

The use of pairing-based ZKP on elliptic curves can potentially enhance the security strength of the system. On the other hand, compared to ZKP protocols based on the factoring problem over Zn, pairing-based ZKPs can also integrate smoothly with other pairing-based cryptographic schemes (e.g., identity-based encryption, pairing-based signatures, key agreement, and proxy re-encryption), making the combined schemes quite efficient. So it is envisioned that pairing-based ZKPs will have a wide application in constructing cryptographic systems with various functionalities.

4.2 Bilinear maps and Pairings

A bilinear map is a map $e : G \times G \rightarrow G_T$, where G is a Gap Diffie-Hellman (GDH) group and G_T is another multiplicative cyclic group of prime order p with the following properties: (i) Computable: there exists an efficiently computable algorithm for computing e; (ii) Bilinear: for all $h1$, $h2 \in$ G and a, $b \in$ Z p , $e(h_1^a, h_2^b) = e(h_1, h_2)^{ab}$; (iii) Non-degenerate: e(g, g)≠1, where g is a generator of G.

The proposed authentication scheme using ZKP based on pairings is as follows:

Bootstrapping Phase:

1. A trusted authority T chooses s as the secret and computes a private key for node i as $h(\mathrm{Id}(i))^s$, where h is a hash function. Assume $h(\mathrm{Id}(i)) = g^{ai}$.

2. T publishes $g^{ai} = \mathrm{h}(\mathrm{Id}(i))$ and g^s .

3. A node A with unique identification Id(A) get its private key $h(\mathrm{Id}(A))^s = g^{aA\,s}$ and public key $(g, g^{aA} = h(\mathrm{Id}(A)), \mathrm{v_A} = e(g, g)^{aA\,s} = e(h(\mathrm{Id}(A)), g^s))$.

Secret Distribution:

1. A cyclic group G of prime order p' with a generator g of G is chosen publicly as a system parameter, where p' satisfies $s < p'$.

2. T generates shares $P(1), \ldots, P(\mathrm{n})$ and distributes each node one value, where $P(x) = s + a_1 x + \ldots + a_t x^t$ is a random polynomial of degree t and $P(0) = s$ is the original secret.

3. T computes and publishes commitments to the coefficients of P, i.e., $c_0 = g^s$, and $c_1 = g^{a1}, \ldots, c_t = g^{at}$.

Authentication Phase:

Node A proves his identity to node B using t rounds, each of which consists of:

1. A selects a random secret $\mathrm{r} \in [0, \mathrm{p}]$ and computes $W_A = \mathrm{e(g, g)^r}$. A sends Id(A) and W_A to B.

2. B selects a random challenge c in $[0, 2^{\mathrm{k}}]$ and sends it to A.

3. A computes and sends the following response to B: $Y_A = \mathrm{g^r} \times (\mathrm{g^{aA\,s}})^{\mathrm{c}}$.

4. B receives Y_A and verify $e(g, \mathrm{Y_A}) = W_A \times v_c$, and accepts this round if it is true.

System Recovery Phase:

Even if T is not available, the nodes that survive attacks can reconstruct the authentication system. Assume a node C initiates the recovery process.

1. C uses the authentication protocol to verify the validity of its neighbor nodes. C should also authenticate itself to others.

2. If authentication succeeds, each node further verifies the integrity of its own share: node i that holds $u = \mathrm{P}(i)$ checks if the following equation holds:

$$\mathrm{g^u} = c_0 c_1^{i^1} \ldots c_{\mathrm{t}}^{i^t} = \prod_{j=0}^{t} \mathrm{g}^{a_j i^j} = \mathrm{g}^{P(i)}. \qquad (3)$$

3. If k − 1 nodes pass the authentication and integrity verification processes, C generates a temporary public/private key pair (k_{pub}, $\mathrm{k_{pri}}$). C broadcasts the request for

secret shares and the self-generated public key k_{pub} to the verified nodes.

4. Each node encrypts its stored secret share using *C*'s temporary public key k_{pub} and sends to *C*.

5. *C* decrypts the received shares. After retrieving (i, P(i)) of all i in *B* (*B* is the set of the $k - 1$ share holders), *C* applies the Lagrange interpolation formula to reconstruct:

$$s = \sum_{i \in B} b_i P(i), \text{ where } b_i = \prod_{j \in B, j \neq i} \frac{j}{j-i} . \quad (4)$$

Analysis. We now analyze the efficiency of the proposed scheme in terms of computational complexity, focusing on the number of group exponentiations or bilinear map pairing evaluations, which are the most computationally extensive operations. During the authentication process, at the prover side, the number of group exponentiations is approximately two (note that c is a k-bit number which is significantly smaller than p), and the number of evaluations of bilinear mapping is zero; At the verifier side, the number of group exponentiations can be neglected since c is very small, and the number of evaluations of bilinear mapping is only one.

Implementation. We next evaluate the pairing-based authentication scheme in Linux. Our experiment is conducted using C on a system with an Intel Core 2 processor running at 2.4 GHz, 768 MB RAM, and a 7200 RPM Western Digital 250 GB Serial ATA drive with an 8 MB buffer. The authentication algorithms are implemented using the Pairing-Based Cryptography (PBC) library version 0.4.21 [16] and the crypto library of OpenSSL version 0.9.8h. All the curve groups we work on have a 160-bit group order. All results are the averages of 100 trials. Table 1 lists all the pairing parameters for different elliptic curve types.

Table 1: Description of all the pairing parameters for different elliptic curve types.

Elliptic curve type	Base field size (bits)	Embedding degree k	Dlog security
a	512	2	1024
d_n	n	6	$6n$
e	1024	1	1024
f	160	12	1920
g_n	n	10	$10n$
a_1	1024	2	2048

As suggested in [16], a type a curve can achieve fastest pairing and is good for cryptosystems where group element size is not critical; a type d_n curve is good for

cryptosystems when group elements must be as short as possible; a type e curve can be easily found and requires only modular arithmetic to implement; a type f curve is useful for insuring against future finite field discrete log algorithm improvements; a type g_n curve runs slower than embedding degree 12 pairings; and a type a1 curve is useful when some cryptosystems need the curve order to be a specific number. Table 2 shows the experimental results using different elliptic curve types.

Table 2: Time complexity of operations during the authentication phase.

Operations / Elliptic curve type	$W_A = e(g,g)^r$	$Y_A = g^r \times (g^{a_A s})^c$	$e(g, Y_A) = W_A \times v_A^c$
a	13.7ms	12.6ms	8.3ms
d_n	15.6ms	3.6ms	18.0ms
e	43.3ms	31.2ms	28.5ms
f	75.1ms	3.7ms	89.9ms
g_n	44.9ms	3.4ms	54.8ms
a_1	296.6ms	199.9ms	211.7ms

Note: we set n = 159 for type d_n curves and n = 149 for type g_n curves.

As we can see, the operations performed in the authentication phase using pairing-based cryptography are efficient. And the use of a and d_n curve types for pairing-based operations can achieve a better average time efficiency than the others. In practice, however, it is more preferable to choose d_n type curves since the base field size of d_n (n = 159) is smaller than that of type a curve, i.e., the corresponding size of group elements in type d159 curves is smaller.

4.3 Integration of Pairing-based ZKP and Attribute-based Encryption (ABE)

Recently, the notion of ABE, which was proposed by Sahai and Waters [17], has attracted much attention in the research community for designing flexible and scalable access control systems. For the first time, ABE enables public key based one-to-many encryption. Therefore, it is envisioned as a highly promising public key primitive for realizing scalable and fine-grained access control systems, where differential yet flexible access rights can be assigned to individual users.

To address complex and general access policy, two kinds of ABE have been proposed: Key-Policy ABE (KP-ABE) [18, 19] and Ciphertext-Policy ABE (CP-ABE) [20–22]. In KP-ABE, access policy is assigned in attribute private keys, whereas, in CP-ABE, the access policy is specified in the ciphertext.

In KP-ABE, data are associated with attributes, and for each a public key component is defined. The encrypter associates the set of attributes to the message by encrypting it with the corresponding public key components. Each user is assigned an access structure, which is usually defined as an access tree over data attributes, i.e., interior nodes of the access tree are threshold gates and leaf nodes are associated with attributes. The access tree is constructed using iterative secret sharing techniques between different layers of nodes, and the user secret key consisting of a set of secret key components is defined to reflect the access structure. Thus, only if the attributes associated with a ciphertext satisfy the user key's access structure can the user recover the root secret and further decrypt the data. This access tree can support very expressive types of control.

Given that airborne distributed data storage and access systems are moving fast toward network centric architectures [23], we can integrate pairing-based ZKP and attribute-based encryption (ABE) [18] to design a unified authentication and data access control system in a single public key cryptography context (i.e., using the same elliptic curve groups). Specifically, to achieve fine-grained data access control, we propose to explore key-policy attribute-based encryption (KP-ABE) [18] that cryptographically binds files to their policies and seamlessly integrates the access structure with data encryption such that only users possessing the requisite set of attributes are able to access the data.

Key-Policy ABE (KP-ABE): A KP-ABE scheme is composed of four algorithms which can be defined as follows:

Setup: This algorithm takes as input a security parameter κ and the attribute universe $U = \{1, 2, \ldots, N\}$ of cardinality N. It defines a bilinear group G_1 of prime order p with a generator g, a bilinear map $e : G_1 \times G_1 \rightarrow G_2$ which has the properties of bilinearity, computability, and non-degeneracy. It returns the public key PK as well as a system master key MK as follows:

$$\begin{aligned} \text{PK} &= (Y, T_1, T_2, \ldots, T_N) \\ \text{MK} &= (y, t_1, t_2, \ldots, t_N), \end{aligned} \tag{5}$$

where $T_i \in G_1$ and $t_i \in Z_p$ are for attribute i, $1 \leq i \leq N$, and $Y \in G_2$ is another public key component. We have $T_i = g^{ti}$ and $Y = e(g, g)^y$, $y \in Z_p$. While PK is publicly known to all the parties in the system, MK is kept as a secret by the authority party T.

Encryption: This algorithm takes a message *M*, the public key PK, and a set of attributes *I* as input. It outputs the ciphertext *E* with the following format:

$$E = (I, \tilde{E}, \{E_i\}_{i \in I}), \tag{6}$$

where $\tilde{E} = \mathrm{MY}^s$, $\mathrm{E_i} = T_i^s$, and s is randomly chosen from Z_p .

Key Generation: This algorithm takes as input an access tree *T*, the master key MK, and the public key PK. It outputs a user secret key SK as follows. First, it defines a random polynomial $\mathrm{p_i}$ (x) for each node *i* of *T* in the top-down manner starting from the root node *r*. For each non-root node *j*, $p_j(0) = p_{\mathrm{parent(j)}}(\mathrm{idx}(j))$ where parent(*j*) represents *j*'s parent and idx(*j*) is *j*'s unique index given by its parent. For the root node *r*, $p_r(0) = y$. Then it outputs SK as follows:

$$\mathrm{SK} = \{sk_i\}\, i \in \mathrm{L}, \tag{7}$$

where *L* denotes the set of attributes attached to the leaf nodes of *T* and $\mathrm{sk}_i = \mathrm{g}^{\frac{p_i(0)}{t_i}}$.

Decryption: This algorithm takes as input the ciphertext *E* encrypted under the attribute set *I*, the user's secret key SK for access tree *T*, and the public key PK. It first computes $e(E_i, sk_i) = e(g, g)^{pi(0)\mathrm{s}}$ for leaf nodes. Then, it aggregates these pairing results in a bottom-up manner using the polynomial interpolation technique. Finally, it may recover the blind factor $Y^s = e(g, g)^{ys}$ and output the message *M* if and only if *I* satisfies *T* .

It can be see that ABE is also based on pairing-based cryptography. Therefore, we can seamlessly integrate pairing-based ZKP and ABE to design a unified authentication and data access control system in a single public key cryptography context (i.e., using the same elliptic curve groups). We propose to use the ABE-based fine-grained distributed data access control scheme in [24]. The basic idea of the data access control scheme is follows: each network node is preloaded with a set of attributes as well as the public key PK. Note that the set of attributes can be any attribute that can be utilized to describe data accessibility in a very expressive manner, e.g., the type of the network node, the type of data, location, time etc. Each user is assigned an access structure and the corresponding secret key SK. The lifetime of the airborne network is divided into m stages, numbering as 1, 2, · · ·, *m*. The stage number is reset to 1 when it increases to m + 1. Each period is further divided into n phases, numbering 1, 2, · · · , *n*, where we set $n < k$, $k = \max_{\forall i \in N} |I_i|$.

Network nodes encrypt and store data on the phase basis. For each network node, the data are encrypted by a symmetric encryption such as AES, and the data encryption keys during one stage form a one-way key chain, one data encryption key for each phase. The master key of each stage is always generated during the preceding stage, and encrypted under the preloaded attributes. Upon request for data, the network node responds with the encrypted master key as well as the ciphertext of the data. If the user is an intended receiver, he is able to decrypt the master key and derive the data encryption key, and then obtain the data.

Analysis: In the distributed data storage and access system, each network node is assigned a set of attributes and each user is assigned an access structure. The proposed scheme has several advantages. First, data are encrypted under the attributes such that only the users whose access structures "accept" these attributes can decrypt. As the access structure is very expressive so that the data access capability of each user can be controlled. Second, it is efficient in terms of key storage, computation, and communication overhead. Finally, it is resistant against user collusion, i.e., the cooperation of colluding nodes will not lead to the disclosure of additional data.

5. CONCLUSIONS

Zero-knowledge Protocols (ZKPs) enable a prover to convince a verifier of the truth of a statement without leaking any other information. The main properties of ZKPs include completeness, soundness and zero-knowledge. These features are correlated with each other, and together with the lighter computational requirements, make zero-knowledge protocols very attractive in authentication services in airborne networks. Although useful, the non-interactive zero-knowledge proofs based on standard cryptographic assumptions used to be inefficient and not useful in practice.

Recently, it has been shown that the only practically efficient constructions of non-interactive zero-knowledge proofs that are based on standard intractability assumptions come from pairing based-cryptography, where a pairing is used between elements of two cryptographic groups to a third group to construct cryptographic systems. In addition, if elliptic curve (EC) groups are utilized as the cryptographic groups, compared to ZKPs where the security of the system usually relies on the hardness of the discrete logarithm problem over Zn with sub-exponential solving time, solving the discrete log problem over elliptic curves takes exponential time. Therefore, the use of pairing-based ZKP on elliptic curves can potentially enhance the security strength of the system. Besides these advantages, pairing-based ZKPs can also integrate smoothly with other pairing-based cryptographic schemes (e.g., identity-based encryption, pairing-based signatures, key agreement, and proxy re-encryption) making the combined schemes quite efficient.

So it is envisioned that pairing-based ZKPs will have a wide application in constructing cryptographic systems with various functionalities. Under AFRL sponsorship in this research grant we:

1) Conducted a survey on the state of the art of pairing-based ZKPs with different elliptic curve groups, including both interactive and non-interactive ZKPs.
2) Investigated and compared the performance of the pairing-based ZKP protocol with other ZKPs based on different hard problems through both theoretical analysis and extensive simulations. Performance was evaluated in terms of computational efficiency, communication efficiency, and security strength.
3) Examined the integration of pairing-based ZKP and attribute-based encryption (ABE) for the design of a unified authentication and data access control system in a single public key cryptography context.

4) Explored key-policy attribute-based encryption (KP-ABE) that cryptographically binds files to their policies and seamlessly integrates the access structure with data encryption.

The results of our investigation indicate that encryption schemes employing Elliptic Pairing-based Zero Knowledge Proofs have application in Tactical Airborne Networks and offer potential for enhancing authentication protocols.

REFERENCES

[1] A. Goel, K. G. Ramakrishnan, D. Kataria, and D. Logothetis, "Efficient computation of delay-sensitive routes from one source to all destinations," in Proc. of INFOCOM'01, 2001.

[2] H. Frey and I. Stojmenovic, "On delivery guarantees of face and combined greedy-face routing in ad hoc and sensor networks," in Proc. of MobiCom'06, 2006.

[3] S. Jain and S. Das, "Exploiting path diversity in the link layer in wireless ad hoc networks," Ad Hoc Networks, 2008.

[4] B. Awerbuch, D. Holmer, and H. Rubens, "The medium time metric: high throughput route selection in multi-rate ad hoc wireless networks," Mobile Networks and Applications, 2006.

[5] D. Zheng, W. Ge, and J. Zhang, "Distributed opportunistic scheduling for ad-hoc communications: An optimal stopping approach," in Proc. of MobiHoc'07, 2007.

[6] L. C. Guillou and J.-J. Quisquater, "A practical zero-knowledge protocol fitted to security microprocessor minimizing both transmission and memory," in Proc. of EUROCRYPT'88, New York, NY, USA, 1988.

[7] P. Feldman, "A practical scheme for non-interactive verifiable secret sharing," in Proc. of IEEE Symposium on Foundations of Computer Science (FOCS'87), vol. 0. Los Alamitos, CA, USA: IEEE Computer Society, 1987, pp. 427–438.

[8] H. A. Aronsson, "Zero knowledge protocols and small systems," http://www.tml.tkk.fi/Opinnot/Tik110.501/1995/zeroknowledge.html, 1995.

[9] J. Groth and A. Sahai, "Efficient non-interactive proof systems for bilinear groups," in Proc. of EUROCRYPT'08, 2008, pp. 415–432.

[10] X. Boyen and B. Waters, "Full-domain subgroup hiding and constant-size group signatures," in Proc. of PKC'07, 2007, pp. 1–15.

[11] J. Groth, "Short pairing-based non-interactive zero-knowledge arguments," in Proc. of ASIACRYPT'10, 2010, pp. 321–340.

[12] J. Groth, "Pairing-based non-interactive zero-knowledge proofs," in Proc. of Pairing'10, 2010, p. 206.

[13] V. S. Miller, "Use of elliptic curves in cryptography," in Proc. of CRYPTO'85, 1986, pp. 417–426.

[14] B. Lynn, "On the implementation of pairing-based cryptosystems," Ph.D. dissertation, Stanford University, 2007.

[15] S. Almuhammadi, N. T. Sui, and D. McLeod, "Better privacy and security in e-commerce: Using elliptic curve-based zero-knowledge proofs," in Proc. of IEEE ECE'04. Los Alamitos, CA, USA:IEEE Computer Society, 2004, pp. 299–302.

[16] B. Lynn, “The pairing-based cryptography library,” 2011.

[17] A. Sahai and B. Waters, “Fuzzy identity-based encryption,” in Proc. of Eurocrypt’05, vol. 3494. Springer, 2005, pp. 457–473.

[18] V. Goyal, O. Pandey, A. Sahai, and B. Waters, “Attribute-based encryption for fine-grained access control of encrypted data,” in proc. of CCS’06, Alexandria, VA, October 2006.

[19] R. Ostrovsky, A. Sahai, and B. Waters, “Attribute-based encryption with non-monotonic access structures,” in Proc. of the 14th ACM Conference on Computer and Communications Security (CCS’07), Alexandria, VA, October 2007.

[20] J. Bethencourt, A. Sahai, and B. Waters, “Ciphertext-policy attribute-based encryption,” in Proc. of IEEE Symposium on Security and Privacy, Oakland, CA, May 2007.

[21] L. Cheung and C. Newport, “Provably secure ciphertext policy ABE,” in Proc. of the 14th ACM conference on Computer and Communications Security (CCS’07), Alexandria, VA, USA, 2007, pp. 456–465.

[22] V. Goyal, A. Jain, O. Pandey, and A. Sahai, “Bounded ciphertext policy attribute based encryption,” in Proc. of the 35th International Colloquium on Automata, Languages and Programming (ICALP’08), Reykjavik, Iceland, July 2008.

[23] K. Benjamin and P. Eric, “Metadata modeling for airborne data acquisition systems,” in Proc. of ITC’07, Las Vegas, Nevada, 2007.

[24] S. Yu, K. Ren, and W. Lou, “FDAC: Toward fine-grained distributed data access control in wireless sensor networks,” in Proc. of INFOCOM’09, 2009, pp. 963–971.

LIST of SYMBOLS, ABBREVIATIONS, and ACRONYMS

ABE	Attribute-Based Encryption
AES	Advanced Encryption Standard
AN	Airborne Networks
CA	Central Authority
CP	Ciphertext-Policy
EC	Elliptic Curves
GQ	Guillou-Qusquater
GHz	Giga Hertz
KP	Key-Policy
MB	Mega Byte
MK	Master Key
PBC	Pairing-Based Cryptology
PKI	Public Key Infrastructure
RAM	Random Access Memory
RPM	Revolutions-Per-Minute
SK	Secret Key
SKC	Symmetric Key Cryptology
VSS	Verifiable Secret Sharing
ZKP	Zero-Knowledge Protocol