TinyTate: Computing the Tate Pairing in Resource-Constrained Sensor Nodes

Leonardo B. Oliveira*
UNICAMP, Brazil
leob@ic.unicamp.br

Diego F. Aranha
UNICAMP, Brazil
dfaranha@gmail.com

Eduardo Morais
UNICAMP, Brazil
eduardo.morais@ic.unicamp.br

Felipe Daguano
UNICAMP, Brazil
daguano@gmail.com

Julio López
UNICAMP, Brazil
jlopez@ic.unicamp.br

Ricardo Dahab
UNICAMP, Brazil
rdahab@ic.unicamp.br

Abstract

After a few years of intense research, Wireless Sensor Networks (WSNs) still demand new secure and cryptographic schemes. On the other hand, the advent of cryptography from pairings has enabled a wide range of novel cryptosystems. In this work we present TinyTate, the first known implementation of pairings for sensor nodes based on the 8-bit/7.3828-MHz ATmega128L microcontroller (e.g., MICA2 and MICAz motes). We then conclude that cryptography from pairings is indeed viable in resource-constrained nodes.

keywords: *sensor networks; security; cryptography; bilinear pairings; Tate pairing implementation*

1 Introduction

Wireless sensor networks (WSNs) [11, 1] are ad hoc networks comprised of small sensor nodes with limited resources and one or more base stations (BSs), which are much more powerful nodes that connect the sensor nodes to the rest of the world. WSNs are used for monitoring purposes, and can be used in different application areas, ranging from battlefield reconnaissance to environmental protection.

Like any wireless ad hoc network, WSNs are vulnerable to attacks [23, 47]. Besides the well-known vulnerabilities due to wireless communication and ad hocness, WSNs face additional problems, including (i) sensor nodes being small, cheap devices that are unlikely to be made tamper-resistant or tamper-proof; and (ii) their being left unattended once deployed in unprotected, or even hostile areas – which makes them easily accessible to malicious parties. It is therefore crucial to add security to WSNs, specially those embedded in mission-critical applications.

On the other hand, cryptography from pairings (or pairings, for short), is an emerging cryptographic primitive that allows a wide range of applications. Pairings have been attracting the interest of international cryptography community because it enables the design of original cryptographic schemes and makes well-known cryptographic protocols more efficient. The two most important pairings are the Tate and the Weil pairings.

It used to be thought that Public Key Cryptography (PKC) was impractical in resource-constrained nodes and security primitives were achieved through symmetric cryptosystems (e.g., RC5 [41] and SkipJack [22]). However, works (e.g., [14, 32]) based on Elliptic Curve Cryptography (ECC) [36, 25], a more efficient PKC technique, have shown that PKC is in fact feasible in WSNs.

As a result, recent works (e.g. [48, 37, 7, 33, 38]) have used PKC schemes based on pairings for bootstrapping security in WSNs, i.e., for setting up symmetric secret keys between communicating nodes. These works contribute in the sense that they show how WSNs can take advantage of pairings, but actually they do not demonstrate that pairing computation is viable in resource-constrained nodes. Besides, the fact that ECC is feasible in WSNs does not automatically imply that pairings are feasible as well. This is because pairings require parameters much larger than conventional ECC and their computation is thus more challenging.

In this work we present TinyTate, the first known implementation of pairings for sensor nodes based on the 8-bit/7.3828-MHz ATmega128L microcontroller, which is present in the MICA [16] family of sensor nodes (e.g., MICA2 and MICAz motes). Our main contribution is to show that cryptography from pairings is indeed viable in resource-constrained nodes.

The remainder of this work is organized as follows. In

*Supported by The State of São Paulo Research Foundation (FAPESP) under grant 2005/00557-9

Section 2, we discuss the related work. In Section 3, we introduce some pairing concepts. TinyTate and its performance results are presented in Sections 4 and 5, respectively. Finally, we conclude in Section 6.

2 Related Work

WSNs are a subclass of MANETs, and much work (e.g., [49, 18]) has been proposed for securing MANETs in general. These studies are not applicable to WSNs because they assume laptop- or palmtop-level resources, which are orders of magnitude larger than those available in WSNs.

The number of studies specifically targeted to secure WSNs has grown significantly. Below, we provide a sample of studies based on cryptographic methods, and then focus on those targeted to PKC and pairings.

A considerable number of works (e.g., [3, 41, 10, 50, 5, 51, 31, 29, 42, 8, 17, 21, 19, 4, 30, 9, 39, 40]) have focused on efficient key management of symmetric cryptosystems. Perrig *et al.* [41] proposed SPINS, a suite of efficient symmetric key based security building blocks. Eschenauer *et al.* [10] looked at random key predistribution schemes, and originated a large number of follow-on studies [17]. And Zhu *et al.* [50] proposed LEAP, a rather efficient scheme based on local distribution of secret keys among neighboring nodes.

The studies specifically targeted to PKC have tried either to adequate conventional algorithms (e.g. RSA) to sensor nodes, or to employ more efficient techniques (e.g. ECC). Watro *et al.* [46] proposed TinyPK. To perform key distribution, TinyPK assigns RSA efficient public operations to nodes and RSA expensive private operations to better suited external parties. Gura *et al.* [14] reported results for ECC and RSA on the ATmega128 and demonstrated that the first outperforms the latter. Their ECC implementation uses prime fields. Malan *et al.* [32] implemented ECC using binary fields and polynomial basis and presented results for the Diffie-Hellman protocol based on the ECDLP. Liu, Kampanakis, and Ning developed TinyECC [28], the ECC library which we make use in TinyTate. TinyECC provides elliptic curve arithmetic over prime fields and uses inline assembly code to speed up critical operations in the ATmega128 processor.

There are also works that have focused on cryptography from pairings for WSNs, in particular (e.g [48, 37, 7, 38, 33]). Zhang *et al.* [48], for example, have made use of Identity-Based Cryptography (IBC) based on pairings for key distribution in WSNs. They hoped that pairings would be soon feasible in resource-constrained nodes and were not concerned with implementation issues. The work of Doyle *et al.* [7] also focused on IBC and presented some simulation results on pairings. The work, however, has considered a class of nodes more powerful than those found in resource-constrained nodes. Oliveira *et al.* [38] have discussed the synergy between WSNs and IBC. Nevertheless, the work only describes estimates for the Tate pairing computation over sensor nodes and no actual implementation is in fact presented. Finally, McKuster *et al.* [33] have focused on a hardware that both implements primitives for computing the Tate pairing and meets the strict energy constraints of sensor nodes.

3 Pairings: concepts

Bilinear pairings were first used in the context of cryptanalysis [34], but their pioneering use in cryptosystems is due the works of Sakai [43] *et al.* and Joux [20]. In this section we first present some paring concepts and then define the Tate pairing. (For more on these definitions, see for instance Galbraith [12].) In what follows, let $E/\mathbb{F}_q$ be an elliptic curve over a finite field $\mathbb{F}_q$, $E(\mathbb{F}_q)$ be the group of points of this curve, and $\#E(\mathbb{F}_q)$ be the group order.

3.1 Bilinear pairing

Let ℓ be a positive integer. Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be additively-written groups of order ℓ with identity $\mathcal{O}$, and let $\mathbb{G}_T$be a multiplicatively-written group of order ℓ with identity 1.

A *bilinear pairing* is a computable, non-degenerate function

$$e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T.$$

The most important property of pairings in cryptographic constructions is the bilinearity, namely:

$$\forall\, P \in \mathbb{G}_1, \forall\, Q \in \mathbb{G}_2 \text{ and } \forall\, a, b \in \mathbb{Z}^*, \text{ we have}$$

$$e([a]P, [b]Q) = e(P, [b]Q)^a = e([a]P, Q)^b = e(P, Q)^{ab}.$$

In practice, the groups $\mathbb{G}_1$ and $\mathbb{G}_2$ are implemented using a group of points on certain elliptic curves and the group $\mathbb{G}_T$ is implemented using a multiplicative subgroup of a finite field.

3.2 Embedding degree

A subgroup G of $E(\mathbb{F}_q)$ is said to have an *embedding degree* k with respect to ℓ if k is the smallest integer such that $\ell \mid q^k - 1$.

3.3 Bilinear Diffie-Hellman Problem

Most of the pairing applications rely on the hardness of the following problem for their security [12]: given P, $[a]P$, $[b]P$, and $[c]P$ for some $a, b \in \mathbb{Z}^*$, compute

$$e(P, P)^{abc}.$$

This problem is known as the *Bilinear Diffie-Hellman Problem*. The hardness of the Bilinear Diffie-Hellman Problem depends on the hardness of the Diffie-Hellman problems both on $E(\mathbb{F}_q)$ and in $\mathbb{F}_{q^k}$. So, for most pairing applications the parameters q, ℓ, and k must satisfy the following security requirements:

1. ℓ must be large enough so that solving the Elliptic Curve Discrete Logarithm Problem (ECDLP) in an order-ℓ subgroup of $E(\mathbb{F}_q)$ is infeasible (e.g. using Pollard's rho algorithm);

2. k must be large enough so that solving the Discrete Logarithm Problem (DLP) in $\mathbb{F}_{q^k}$ is infeasible (e.g., using the index-calculus method).

3.4 The Tate pairing

Let $E(\mathbb{F}_q)$ contain a subgroup of prime order ℓ coprime with q and with embedding degree k. (In most applications, ℓ also is a large prime divisor of $\#E(\mathbb{F}_q)$.) The *Tate pairing* is the bilinear pairing

$$\hat{e} : E(\mathbb{F}_{q^k})[\ell] \times E(\mathbb{F}_{q^k})/[\ell]E(\mathbb{F}_{q^k}) \rightarrow \mathbb{F}^*_{q^k}/(\mathbb{F}^*_{q^k})^{\ell}.$$

4 TinyTate – Computing the Tate pairing on resource-constrained nodes

The time consuming part while evaluating applications based on pairings is the pairing computation[1]. In this section we present TinyTate, an implementation of the Tate pairing for resource-constrained nodes.

Below, we discuss the implementations issues and the decision-makings we faced while developing TinyTate. Recall from Section 3 that $E/\mathbb{F}_q$ is an elliptic curve defined over $\mathbb{F}_q$, ℓ is a large prime divisor of $\#E(\mathbb{F}_q)$ coprime to q, and k is the embedding degree.

4.1 Field selection

Given a cryptosystem, the hardness of its underlying problem dictates the size of the security parameters. Namely, the harder the problem, the smaller the parameter size. The parameter size, in turn, dictates the efficiency, i.e., the smaller the parameter size, the faster the computation time. The DLP in prime fields is considered to be harder than the DLP in binary fields. As a result, we use prime fields in TinyTate.

[1] Other operations computed efficiently [13] in ATmega128L might also be needed (e.g. hashing).

Algorithm 1 Tate pairing computation

Input: $P \in E(\mathbb{F}_{q^k})[\ell]$, $Q \in E(\mathbb{F}_{q^k})[\ell]$
Output: $\hat{e}(P, Q)$
1. $T \leftarrow P$
2. $f \leftarrow 1$
3. **For** $i \leftarrow \lfloor lg(\ell) \rfloor - 1$ **downto** 0 **do**
4. Compute tangent l and vertical v lines for $[2]T$
5. $T \leftarrow [2]T$
6. $f \leftarrow f^2 \cdot \frac{l(Q)}{v(Q)}$
7. **If** the ith bit of ℓ is one, **then**:
8. Compute lines l and v for $T + P$
9. $T \leftarrow T + P$
10. $f \leftarrow f \cdot \frac{l(Q)}{v(Q)}$
11. **end for**
12. **Return** $f^{(q^k-1)/\ell}$

Figure 1. Tate pairing computation: Miller's algorithm and final exponentiation.

4.2 Curve selection

Supersingular curves have been shown empirically to be faster [45] than nonsupersingular curves. Authors, however, tend to choose nonsupersingular curves rather than supersingular curves because they feel that the latter have security advantages compared to the formers. Since until now no concrete evidence for that has appeared [45], we use supersingular curves in TinyTate.

4.3 Parameters q and ℓ

The choice of the parameters q and ℓ is a key factor in the efficiency of pairing computation, as curve operations are performed using arithmetic of the underlying field. In prime fields, by choosing q a Mersenne prime (i.e., a number of the form $2^p - 1$) helps in computing modular reduction operations efficiently. However, it has been shown recently that such technique also decreases the hardness of the DLP in $\mathbb{F}_q$ (e.g., [44]) and is potentially unsafe in the context of pairings. For ℓ, on the other hand, it is possible to choose a Solinas prime, which decreases the number of operations in the Miller's algorithm (Fig 1, Steps 7, 8, 9, and 10) and makes the pairing computation faster.

4.4 Embedding degree

TinyTate uses the embedding degree $k = 2$ since it provides a number of benefits while computing pairings [45]. For example, $k = 2$ allows the denominator elimination optimization and makes $\mathbb{F}_{q^k}$ arithmetic easier to implement.

4.5 Twists

Let d be a quadratic non-residue in $\mathbb{F}_q$. The *twist* of an elliptic curve $E/\mathbb{F}_q : y^2 = x^3 + ax + b$ is given by $E^t/\mathbb{F}_q : y^2 = x^3 + d^2ax + d^3b$. For $k = 2$, there exists a mapping $\phi : E(\mathbb{F}_{q^2}) \rightarrow E^t(\mathbb{F}_q)$ such that $\phi[(a, 0), (0, d)] \rightarrow (-a, d)$, and arithmetic in $E(\mathbb{F}_{q^2})$ can be thus carried out faster in the group $E^t(\mathbb{F}_q)$. In TinyTate, twists are used to speed-up arithmetic operations.

4.6 Security level

Parameter sizes often pose a tradeoff between security level and efficiency. For most pairing schemes, the security requirements described in Section 3 can be satisfied by choosing $\ell \geq 2^{160}$ and $q^k \geq 2^{1024}$. However, security requirements in WSNs are often relaxed [41] to meet their needs for efficiency. This is possible because of their short lifetimes and because the goal is not to protect each node individually, but the network operation as a whole. Until now, the larger parameters sizes for which the ECDLP and the DLP in prime fields are known to be solved are 2^{109} [26] and 2^{448} [6], respectively. Therefore, it seems that $\ell \geq 2^{128}$ and $q^k \geq 2^{512}$ are able to meet the current security requirements of WSNs[2].

4.7 Coordinate system

The two most common coordinate systems are the *projective* system (x, y, z) and the *affine* (x, y) system . The affine system requires inversions while performing point addition or doubling operations. The inverse operation, in turn, is commonly expensive. The projective system, on the other hand, reduces the need for inverse and thus seems to be more adequate to our implementation.

4.8 ECC library

Like any other cryptographic primitive based on ECC, finite fields and elliptic curve arithmetics are crucial to performance of pairings. The efficiency of these arithmetics, in turn, depends on the library that is being used. We have chosen TinyECC [28] as ECC library for TinyTate since it was developed specifically to our target processor, i.e., the ATmega128L.

4.9 Computation

Pairing computation is challenging because it includes operations in extended fields, i.e., fields much larger than fields used in conventional ECC.

[2]Very recently, it has come to our attention that a new record for DLP in prime fields stands at 530 bits [24]. However, because of the WSNs' idiosyncrasies it does not seem to be a problem.

Algorithm 2 BKLS Algorithm

Input: $P \in E(\mathbb{F}_{q^k})[\ell]$, $Q \in E(\mathbb{F}_{q^k})[\ell]$
Output: $\hat{e}(P, Q)$
1. $m = 1$
2. $A = P$
3. $n = \ell - 1$
4. **For** $i \leftarrow \lfloor lg(\ell) \rfloor - 2$ **downto** 0 **do**
5. $\quad m = m^2 \cdot g(A, A, Q)$
6. $\quad$ **if** $n_i = 1$ **then** $m = m \cdot g(A, P, Q)$
7. **end for**
8. $m = \overline{m}/m$
9. $m = m^{(q+1)/\ell}$
10. **Return** m

Figure 2. Tate pairing computation using the BKLS algorithm.

The algorithm for computing the Tate pairing consists of two stages: (i) application of the Miller's algorithm [35] (Fig. 1) and (ii) a final exponentiation (Fig. 1, Step 12). The last stage is necessary for the algorithm to output a unique value. To compute the two stages, TinyTate uses the BKLS algorithm from Barreto, Kim, Lynn, and Scott [2] (Fig. 2). Concerning this algorithm, we will only mention that the function g (Fig. 2, Steps 5 and 6) is responsible for evaluating $\frac{l(Q)}{v(Q)}$ (Fig. 1, Step 10). For more on the algorithm, please refer to the original paper [2].

5 Results

In this section, we present results on computing TinyTate over MICAz, the new generation of MICA mote node [16]. MICAz is powered with the ATmega128L microcontroller (8-bit/7.3828-MHz processor, 4KB SRAM, 128KB flash memory).

More specifically, the results consider: (i) the Tate pairing on elliptic curves defined over fields with a large prime characteristic; (ii) the embedding degree $k = 2$, q a 256-bit prime, and ℓ a 128-bit Solinas prime; and (iii) group field arithmetic using projective coordinates. To be concrete, we use the curve $E/\mathbb{F}_q : y^2 = x^3 + x$ with the parameters:

$q = 377816068895982358567455764726583947214816257$
$071533302983957476142038207746163;$
$k = 2;$
$\ell = 170141188531071632644604909702696927233;$
$h = 222060320700642449943812747791145685108;$

where h stands for the cofactor of the curve order $\#E(\mathbb{F}_q)$[3].

Results in Table 1 were measured on a MICAz node running TinyOS [27]. The average execution time to compute a pairing is 30.21s. The costs concerning RAM and ROM (flash) memory are 1,831 and 18,384 bytes, respectively.

Tate Pairing		
Time (seconds)	**RAM (bytes)**	**ROM (bytes)**
30.21	1,831	18,384

Table 1. Costs to evaluate the Tate Pairing on MICAz.

Like any other network, WSNs use PKC schemes to bootstrap security and then employ symmetric schemes to communicate. That means that in most applications sensor nodes will evaluate pairings only once and, therefore, the costs above are not a heavy burden to the whole system.

6 Conclusion

After some years of intense research, WSNs still demand new secure and cryptographic schemes. On the other hand, the advent of pairings has enabled a wide range of novel cryptosystems. In this work we present TinyTate, the first known implementation of pairings for sensor networks based on the 8-bit/7.3828-MHz ATmega128L microcontroller. By doing that, we show that pairings are indeed viable in resource-constrained nodes.

For future work, we will consider other pairings (e.g. Ate pairing [15]) and finite fields (e.g., binary fields).

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. A survey on sensor networks. *IEEE Communications Magazine*, 40(8):102–114, August 2002.

[2] P. S. L. M. Barreto, H. Y. Kim, B. Lynn, and M. Scott. Efficient algorithms for pairing-based cryptosystems. In *the 22nd Annual Int'l Cryptology Conference on Advances in Cryptology CRYPTO '02*, pages 354–368, 2002.

[3] D. W. Carman, P. S. Kruus, and B. J. Matt. Constraints and approaches for distributed sensor network security. Technical report, NAI Labs, The Security Research Division, Network Associates, Inc., 2000.

[4] S. A. Çamtepe and B. Yener. Combinatorial design of key distribution mechanisms for wireless sensor networks. In *9th European Symposium on Research Computer Security (ESORICS'04)*, pages 293–308, Sophia Antipolis, France, September 2004. Lecture Notes in Computer Science.

[5] H. Chan, A. Perrig, and D. Song. Random key predistribution schemes for sensor networks. In *IEEE Symposium on Security and Privacy (S&P'03)*, pages 197–213, may 2003.

[6] A. Dorofeev, D. Dygin, and D. Matyukhin. Nabble forums – number theory. `http://www.nabble.com/Discrete-logarithm-in-GF(p)-----135-digits-t2870677.html`.

[7] B. Doyle, S. Bell, A. F. Smeaton, K. McCusker, and N. O'Connor. Security considerations and key negotiation techniques for power constrained sensor networks. *The Computer Journal (Oxford University Press)*, 49(4):443–453, 2006.

[8] W. Du, J. Deng, Y. S. Han, S. Chen, and P. Varshney. A key management scheme for wireless sensor networks using deployment knowledge. In *Conference of the IEEE Communications Society (INFOCOM'04)*, 2004.

[9] W. Du, J. Deng, Y. S. Han, P. K. Varshney, J. Katz, and A. Khalili. A pairwise key pre-distribution scheme for wireless sensor networks. *ACM Transactions on Information and System Security*, 8(2):228–58, 2005. Also in CCS'03.

[10] L. Eschenauer and V. D. Gligor. A key management scheme for distributed sensor networks. In *9th ACM conf. on Computer and communications security (CCS'02)*, pages 41–47, 2002.

[11] D. Estrin, R. Govindan, J. S. Heidemann, and S. Kumar. Next century challenges: Scalable coordination in sensor networks. In *Mobile Computing and Networking (MobiCom'99)*, pages 263–270, Seattle, WA USA, 1999.

[12] S. Galbraith. Pairings. In I. Blake, G. Seroussi, and N. Smart, editors, *Advances in Elliptic Curve Cryptography*, London Mathematical Society Lecture Notes, chapter IX, pages 183–213. Cambridge University Press, 2005.

[13] P. Ganesan, R. Venugopalan, P. Peddabachagari, A. Dean, F. Mueller, and M. Sichitiu. Analyzing and modeling encryption overhead for sensor network nodes. In *2nd ACM international conference on Wireless sensor networks and applications*, pages 151–159. ACM Press, 2003.

[14] N. Gura, A. Patel, A. Wander, H. Eberle, and S. C. Shantz. Comparing elliptic curve cryptography and rsa on 8-bit cpus. In *Workshop on Cryptographic Hardware and Embedded Systems (CHES'04)*, pages 119–132, 2004.

[15] F. Hess, N. Smart, and F. Vercauteren. The eta pairing revisited. *IEEE Transactions on Information Theory*, 52(10):4595–4602, October 2006.

[16] J. L. Hill and D. E. Culler. Mica: A wireless platform for deeply embedded networks. *IEEE Micro*, 22(6):12–24, 2002.

[17] D. Huang, M. Mehta, D. Medhi, and L. Harn. Location-aware key management scheme for wireless sensor networks. In *2nd ACM workshop on Security of ad hoc and sensor networks (SASN'04)*, pages 29–42. ACM Press, 2004.

[18] J.-P. Hubaux, L. Buttyán, and S. Capkun. The quest for security in mobile ad hoc networks. In *2nd ACM international symposium on Mobile ad hoc networking & computing*, pages 146–155. ACM Press, 2001.

[19] J. Hwang and Y. Kim. Revisiting random key predistribution schemes for wireless sensor networks. In *2nd ACM workshop on Security of ad hoc and sensor networks*, pages 43–52. ACM Press, 2004.

[3] Note that in this particular case there is a twist with the same equation of the original curve.

[20] A. Joux. A one round protocol for tripartite diffie-hellman. *J. Cryptology*, 17(4):263–276, 2004. Proceedings of ANTS-IV, 2000.

[21] R. Kannan, L. Ray, and A. Durresi. Security-performance tradeoffs of inheritance based key predistribution for wireless sensor networks. In *1st European Workshop on Security in Wireless and Ad-Hoc Sensor Networks (ESAS´ 04)*, Heidelberg, Germany, August 2004.

[22] C. Karlof, N. Sastry, and D. Wagner. Tinysec: A link layer security architecture for wireless sensor networks. In *2nd ACM SensSys*, pages 162–175, Nov 2004.

[23] C. Karlof and D. Wagner. Secure routing in wireless sensor networks: Attacks and countermeasures. *Elsevier's AdHoc Networks Journal, Special Issue on Sensor Network Applications and Protocols*, 1(2–3):293–315, 2003. Also apeared in 1st IEEE International Workshop on Sensor Network Protocols and Applications.

[24] T. Kleinjung. Discrete logarithms in gf (p) ¿ 160 digits. `http://listserv.nodak.edu/cgi-bin/wa.exe?A2=ind0702&L=nmbrthry&T=0&P=194,`.

[25] N. Koblitz. Elliptic curve cryptosystems. *Mathematics of computation*, 48:203–209, 1987.

[26] R. Lercier. Home page: Computations - discrete logarithms. `http://medicis.polytechnique.fr/~lercier/?lng=en`.

[27] P. Levis, S. Madden, J. Polastre, R. Szewczyk, K. Whitehouse, A. Woo, D. Gay, J. Hill, M. Welsh, E. Brewer, and D. Culler. TinyOS: An operating system for wireless sensor networks. In W. Weber, J. Rabaey, and E. Aarts, editors, *Ambient Intelligence*. Springer-Verlag, New York, NY, 2004.

[28] A. Liu, P. Kampanakis, and P. Ning, 2006.

[29] D. Liu and P. Ning. Location-based pairwise key establishments for static sensor networks. In *1st ACM workshop on Security of ad hoc and sensor networks (SASN´03)*, pages 72–82. ACM Press, 2003.

[30] D. Liu, P. Ning, and R. Li. Establishing pairwise keys in distributed sensor networks. *ACM Transactions on Information and System Security (TISSEC)*, 8(1):41–77, 2005. Also in CCS'03.

[31] D. Liu and P.Ning. Efficient distribution of key chain commitments for broadcast authentication in distributed sensor networks. In *10th Annual Network and Distributed Systems Security Symposium (NDSS'03)*, pages 263–276, 2003.

[32] D. J. Malan, M. Welsh, and M. D. Smith. A public-key infrastructure for key distribution in tinyos based on elliptic curve cryptography. In *1st IEEE International Conference on Sensor and Ad Hoc Communications and Networks (SECON'04)*, Santa Clara, California, October 2004.

[33] K. McCusker, N. O'Connor, and D. Diamond. Low-energy finite field arithmetic primitives for implementing security in wireless sensor networks. In *2006 International Conference on Communications, Circuits And Systems*, volume III - Computer, Optical and Broadband; Communications; Computational Intelligence, pages 1537–1541, June 2006.

[34] A. Menezes, T. Okamoto, and S. Vanstone. Reducing elliptic curve logarithms to logarithms in a finite field. *IEEE Transactions on Information Theory*, 39(5):1639–1646, 1993.

[35] V. Miller. Short program for functions on curves, 1986. unpublished manuscript.

[36] V. Miller. Uses of elliptic curves in cryptography, advances in cryptology. In *Crypto'85, Lecture Notes in Computer Science*, volume 218, pages 417–426. Springer-Verlag, 1986.

[37] L. B. Oliveira and R. Dahab. Pairing-based cryptography for sensor networks. In *5th IEEE International Symposium on Network Computing and Applications*, Cambridge,MA,USA, July 2006. fast abstract.

[38] L. B. Oliveira, R. Dahab, J. Lopez, F. Daguano, and A. A. F. Loureiro. Identity-base encryption for sensor networks. In *3rd IEEE PerCom Workshop on Pervasive Wireless Networking (PerSeNS'07). In proceedings of IEEE PerCom 2007*, White Plains, NY, March 2007.

[39] L. B. Oliveira, H. C. Wong, M. Bern, R. Dahab, and A. A. F. Loureiro. SecLEACH – a random key distribution solution for securing clustered sensor networks. In *5th IEEE International Symposium on Network Computing and Applications*, Cambridge,MA, July 2006. p. 145-154.

[40] L. B. Oliveira, H. C. Wong, R. Dahab, and A. A. F. Loureiro. On the design of secure protocols for hierarchical sensor networks. *International Journal of Networks and Security*, 2(3/4):216–227, 2007. Special Issue on Cryptography in Networks.

[41] A. Perrig, R. Szewczyk, V. Wen, D. Culler, and J. D. Tygar. SPINS: Security protocols for sensor networks. *Wireless Networks*, 8(5):521–534, Sept. 2002. Also in MobiCom'01.

[42] R. D. Pietro, L. V. Mancini, and A. Mei. Random key-assignment for secure wireless sensor networks. In *1st ACM workshop on Security of ad hoc and sensor networks (SASN'03)*, pages 62–71, 2003.

[43] R. Sakai, K. Ohgishi, and M. Kasahara. Cryptosystems based on pairing. In *Symposium on Cryptography and Information Security (SCIS2000)*, pages 26–28, Jan 2000.

[44] O. Schirokauer. The number field sieve for integers of low weight. Cryptology ePrint Archive, Report 2006/107, 2006. `http://eprint.iacr.org/`.

[45] M. Scott. Computing the tate pairing. In *Topics in Cryptology - CT-RSA*, volume 3376 of *Lecture Notes in Computer Science*, pages 293–304. Springer, 2005.

[46] R. J. Watro, D. Kong, S. fen Cuti, C. Gardiner, C. Lynn, and P. Kruus. Tinypk: securing sensor networks with public key technology. In *2nd ACM Workshop on Security of ad hoc and Sensor Networks (SASN'04)*, pages 59–64, 2004.

[47] A. D. Wood and J. A. Stankovic. Denial of service in sensor networks. *IEEE Computer*, 35(10):54–62, Oct. 2002.

[48] Y. Zhang, W. Liu, W. Lou, and Y. Fang. Securing sensor networks with location-based keys. In *IEEE Wireless Communications and Networking Conference (WCNC'05)*, 2005.

[49] L. Zhou and Z. J. Haas. Securing ad hoc networks. *IEEE Network*, 13(6):24–30, 1999.

[50] S. Zhu, S. Setia, and S. Jajodia. LEAP: efficient security mechanisms for large-scale distributed sensor networks. In *10th ACM conference on Computer and communication security (CCS'03)*, pages 62–72. ACM Press, 2003.

[51] S. Zhu, S. Xu, S. Setia, and S. Jajodia. Establishing pairwise keys for secure communication in ad hoc networks: A probabilistic approach. In *11th IEEE Inter'l Conference on Network Protocols (ICNP'03)*, pages 326–335, Atlanta, Nov 2003.