Securing Constrained Networks with ID-based Cryptography and Short Signatures

Related Work Report for AW2

Tobias Markmann

February 13, 2014

Constrained networks, like wireless sensor networks or the Internet of Things, require special security approaches due to their limitations. Solutions to low-level communication patterns in WSNs have been widely studied before. This work provides an overview on security solutions for high-level communication patterns in constrained networks, i.e. the node-to-node, user-to-node and node-to-user patterns. By reviewing three works, based on ID-based cryptography or short signatures, and analyzing them regarding their applicability for constrained networks, I provide an overview, highlight important problems and draw overall conclusions.

Contents

1 Introduction

Constrained networks have been in use for a long time in form of wireless sensor networks (WSNs) and the applications of the Internet of Things (IoT) is continuously increasing. The devices in these networks are usually constrained in different dimensions, from computational power over their communication capabilities to their available energy resources. Due to these tight limits, finding good security solutions for constrained networks becomes especially challenging.

WSNs have a wide range of applications, with a popular one being monitoring scenarios of any kind. The IoT is a very broad term for appliances and devices connected via the Internet, may it be the interconnection of home appliances or on a bigger scale, smart grids[1].

While the IoT is a relatively new area, security in WSNs has been widely studied for more than a decade. Furthermore, there is a trend of WSNs being integrated into or turning into the IoT, as it shows a high heterogeneity and scalability[2].

Symmetric cryptography alone is insufficient to provide authentication of communication in WSNs and the IoT due to their dynamic and flexible structure. Furthermore, scalability of using symmetric signatures, like message authentication codes (MACs), doesn't scale well to a bigger number of participants if one-to-one authentication is required.

To gain an overview of possible solutions to the authentication problem in constrained networks, three different proposals are studied and analyzed. They all approach the problem by using asymmetric cryptography to deal with the flexibility requirements, but differ in their application scenarios and their use of cryptographic methods, namely identity-based cryptography (IBC) and short signatures.

2 Background

2.1 Constrained Networks

WSNs and the IoT both fall into the category of constrained networks. While WSNs are usually limited in scale and interoperability between different installations isn't the highest priority, the IoT resides within the usual global addressable Internet and even though the devices remain behind gateways global device-to-device communication over the Internet is possible. The devices in these networks can be limited in different aspects:

- *Computational Power*: While WSNs usually have computational power around a couple MHz, due to their battery constraints and lifetime requirements, the devices in the IoT have highly varying performance due to their wide range of applications.

- *Memory (RAM)*: Due to the high limitations in WSNs, memory usage of added security mechanisms is critical. If there is only a couple KB of RAM available, the security mechanisms have to be light.

- *Program Storage (ROM)*: This is the fixed memory available for the operating system and the application. For WSN hardware the program storage ranges from a couple ten KB to a couple hundred KB.

- *Power Supply*: Depending on application scenario, sensor nodes are powered by battery or draw their energy from a wired source. Highly efficient security solutions are important for battery-powered devices to ensure long lifetimes.

- *Radio/Communication*: The power requirements of the communication and the transmission speed influence the choice of an ideal security framework for an application. For highly expensive radio usage, compression or other saving mechanisms have to be adopted to reduce the overall energy footprint.

2.2 Asymmetric Cryptography on Constrained Devices

Asymmetric cryptography provides great key-management scalability and flexibility properties compared to symmetric cryptography which ideally suits networks like the IoT or WSNs. Compared to symmetric cryptography, there is no need to manage 1-to-1 key pairs for all potential communication partners. By avoiding the use of shared keys, which is common in symmetric schemes, it reduces the negative effect of node compromises.

Signature and encryption schemes in asymmetric cryptography can be categorized as either *certificate-based* or *certificate-free*.
In cryptography certificates, like standard X.509 certificates as used in the WWW, are used to securely bind an identity to the public key belonging to that identity. In this case the public key information is explicitly bound to the identity using the certificate document, which is cryptographically signed by a common trusted certificate authority (CA). Certificate-based schemes enable a more flexible way to create private/public key information. The user can securely generate her pivate/public keypair and just have to give the public key to the CA to sign it and the authenticated identity.
Certificate-free schemes however have an implicit binding of public key and identity. This is done by a common trusted authority (TA), similar to a CA. A TA generates all the private keys for all participating parties in the system. Thereby, signatures generated in this scheme, can simply be verified by globally known public parameters and the identity that created the signature. However, this inherits the problem of *key escrow*, since all parties have to place a high trust in the TA, generating and having access to all private keys.

Key escrow, also known as key recovery, is the concept where a third party has the ability to recover others keys at any time. In the 1990s the topic raised to higher attention in the cryptography and security communities as some parties pushed it as a way to provide lawful interception by embedding key escrow in all encryption protocols. However, it comes with a large amount of risks and liabilities which makes it unattractive for large international multiparty interaction [3].

Authentication in constrained networks can be done centralized or decentralized. The centralized approach has the advantage that a powerful node or base station can perform heavier operations than the energy constrained nodes. However, it has the disadvantage of being a single point of failure and if it is the sole method of authentication within the network, bogus nodes can start sending messages to other nodes which need to forward, yet unau-

thenticated, messages along a path to the base station, which can only then decide whether the message is authentic or not. The required forwarding of yet unclear authenticated packets leads to higher communication and opens the door for energy depletion style Denial of Service (DoS) attacks[4].

2.3 Identity-based Cryptography

IBC is a form of asymmetric cryptography, where the public key can be deduced from an identity, technically a string which identifies a member in the cryptography system, e.g. an IP or e-mail address. Identity information is commonly already part of most communication protocol messages and therefore it is unnecessary to add public keys or certificates to the messages. IBC has first been proposed by Shamir [5], and Boneh and Franklin [6] finally provided an implementation of identity-based encryption (IBE), which has been an unsolved problem before. The absence of additional public keys cuts down communication size, since contrary to classic public key protocols like transport layer security (TLS), public keys in form of certificates don't have to be transferred.

The main difference in the workflow, compared to classic asymmetric digital signatures like RSA or Digital Signature Algorithm (DSA), is the key generation, which isn't done by the user but the TA and securely transferred to the user and the verification, which uses the identity belonging to a message instead of their public key.

Due to the fact that IBC is a certificate-free scheme and private keys are generated by a commonly trusted party, the TA, it is subject to *key escrow*. This means the key-generating server (KGS) is able to decrypt any messages within the system and produce valid signatures for any user in the system.

Most of the proposed IBC schemes make use of bilinear pairings, also known as pairing-based cryptography (PBC), which are described in more detail in section 2.4.

2.4 Pairing-based Cryptography

Pairing-based cryptography is a research area that deals with building cryptographic tools using a bilinear mapping between two groups. Mathematically speaking it can be defined by two additive groups $\mathbb{G}_1$ and $\mathbb{G}_2$, a multiplicative group $\mathbb{G}_M$ and the final bilinear pairing described as $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_2 \longrightarrow \mathbb{G}_M$. All groups are cyclic and of the same prime order q. For this pairing the properties below must hold:

1. **Bilinearity:** $\hat{e}(a \cdot P, b \cdot Q) = \hat{e}(P, Q)^{a \cdot b}$, $\forall P \in \mathbb{G}_1, Q \in \mathbb{G}_2$ and $a, b \in \mathbb{Z}_q$.

2. **Computability:** The paring of two values via $\hat{e}$ has to be efficiently computable for allowing PBC to be a sensible option for security mechanisms. Efficient algorithms for the computation of pairings exist and are described in [7].

3. **Non-degeneracy:** $\exists P \in \mathbb{G}_1, Q \in \mathbb{G}_2 : \hat{e}(P, Q) \neq 1$.

Based on these groups various problems, believed to be computationally hard, have been described. Among them the Bilinear Diffie-Hellman Problem (BDHP) and the Decision Bilinear Diffie-Hellman Problem (DBDH) which are similar to the classic Diffie-Hellman problem but based on the bilinear paring construction. With these constructs PBC allowed to build solutions for several, previously unsolved,

cryptographic problems.

There have been various different applications in cryptography, where bilinear pairings of this kind have shown to be useful. One example are short signatures, where the BLS signature[8] is the first realization that consistently provides signatures with a security level comparable to common 320 bit Elliptic Curve DSA (ECDSA) signatures but only with a size of 160 bit. This is especially useful where one has external space constraints on the overall size of communication or where human verification is required. The BLS signature scheme is based on the Weil-Pairing.

Another success story of bilinear pairings is the implementation of IBE, where Boneh and Franklin first showed a way to build an ID-based encryption scheme using the Weil-Pairing[6], which has been an unsolved problem before since the initial description if IBC by Shamir in the 1980s.

3 High-level Communication Schemes

In contrast to a low-level view of communication patterns, like unicast, broadcast, convergecast and local gossip, this section provides an overview of high-level communication schemes, like communication within a network of similar nodes and with external users. This basically covers scenarios of communication within a self-contained system for data management and routing management purposes, but also communication between members of the system and entities outside of the system. The three different major communication patterns reviewed in this paper are shown in Figure 1.

Asymmetric cryptography shows to fit these different patters very nicely due to its flexibly and dynamic key management properties. However, these advantages come with heavier computational requirements and particular security properties, which need to be considered for each specific application.

By studying high-level interactions and ways to secure these, a broader security overview is presented. Solutions proposed in the three main papers[9]–[11] are reviewed and set in perspective from the general problem to secure constrained networks.

Authentication for networks is the first essential step for providing security. For the provided overview, the following two network participants are of interest:

1. *Nodes* within the network, which communicate with each other in a direct or indirect way and need to authenticate other nodes of the system.

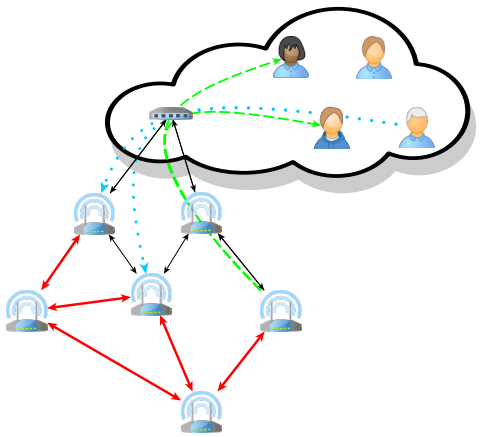

Figure 1: Different high-level communication patterns: a) Node-to-Node (*solid red*), b) User-to-Node (*dashed green*) and c) Node-to-User (*dotted cyan*).

2. *External users* of the network, which need to be authenticated and identified by the nodes to ensure only legitimate users are allowed access and to differentiate levels of access among valid users.

3.1 Node-to-Node

Note-to-node communication, communication pattern a) in Figure 1, includes all schemes where only nodes within the same network are communicating with each other. This covers most self-contained sensor networks but also the IoT use case.

This is a very common communication pattern, not only from a high-level view but also on a lower-level. Most WSN protocols work by communicating monitoring results up to some base station which can happen over multiple hops. The same goes for communication of control messages from a base station back to the sensor nodes. On a higher level, it is also practice for Internet enabled devices in the IoT to have rather direct communication between sensors and actors.

In either case, having end-to-end authentication is an important step to establish secure communication between the participants. For multi-hop communication, which is mostly any communication nowadays, may it be in smaller WSNs or the large Internet, it is beneficial to verify authentication of packets at any intermediate hop to early drop false packets. Dropping false packets, issued for example by attackers, as early as possible is essential to counter DoS attacks from depleting precious energy resources of the network as a whole. Ideally, only the entry point in the network for the attacker is affected from this attack.

3.1.1 Authentication Framework for Wireless Sensor Networks

As part of their proposed authentication framework for WSNs, Yasmin, Ritter, and Wang devise an ID-based Online/Offline signature scheme for direct authenticated broad- and multicast of messages by senders[9].

Authenticated broad- and multicast for WSNs has previously been broadly considered by the scientific community, however the authentication schemes require the redistribution of the message to be broad-/multicast by a base station. The most prominent example in this area is μTESLA[12].

μTESLA is an efficient protocol for authenticated broadcast for wireless sensor nodes. It is based on hash chains and delayed disclosure of authentication keys. By avoiding public-key cryptography (PKC), their protocol has a very low computational overhead but also comes with some downsides, as the authors note. It only allows indirect broadcast messages by requiring all broadcast messages to be signed and finally distributed by the base station. In addition, by building on delayed authentication, messages that can't be immediately verified need to be buffered for some amount of time. This opens attack on the storage capacities of sensor nodes.

The protocol proposed by Yasmin, Ritter, and Wang for authenticated broadcast by sensor nodes uses an identity-based online/offline signature (IBOOS) scheme.
The online/offline signature allows to split the usually computational expensive signature calculation in an offline phase, to be performed on a high-power device, and an online phase which involves little computation. The usual step of signing in the IBC workflow is split into *offline signing*, return-

ing a message independent signature based on the private key and the system parameters, and *online signing*, taking the offline signature and the message and returning the final signature for a message. The verification step still happens completely online.
This is especially essential for low-powered and energy-constrained devices, like the nodes in WSNs.

Comparing certificate based broadcast authentication with storage or transfer or certificates and classic identity-based signatures (IBSs), it shows that their evaluated IBOOSs schemes can sign two to ten times more messages with the same energy. The traffic overhead for the authentication is low, compared to the certificate based schemes that transfer the certificates and no storage of public keys of other nodes is required.

However, it is to note that, according the detailed implementation results[13], the computational overhead for verification is about two orders of magnitude more expensive than for the online signature generation counterpart.

3.1.2 Signcryption Scheme for Smart Grid

So, Kwok, Lam, *et al.* suggest to use IBC to secure the communication within a smart grid[11] with their signcryption scheme.

Signcryption is a term coined by Zheng in 1997, describing the concept of public-key methods that simultaneously provide authentication and confidentiality, the properties of digital signatures and encryption[14], and having a smaller cost overhead compared to applying signature and encryption separately.

While the focus of this work is mainly on

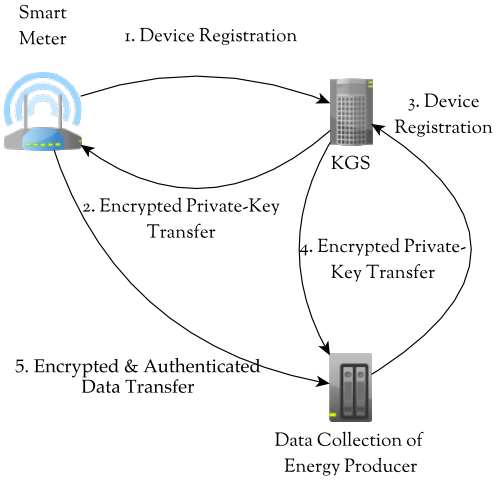

Figure 2: Protocol workflow of signcryption protocol described in [11].

authentication, their work also provides confidentiality due to their smart-grid usage scenario, where private electrical consumption measurements are collected and send over uncontrolled public networks to the energy provider. Due to the large size and dynamic evolution of a advanced metering infrastructure (AMI), they provide a zero-configuration solution which, compared to symmetric and classical asymmetric cryptography approaches, is characterized by very little configuration overhead and great scalability properties.

Their signcryption scheme consists of two phases, 1) the device registration at the KGS, acting as the TA of the system, with one-time keys and 2) the actual transfer of user data in an encrypted and authenticated fashion. The overall protocol flow is shown in Figure 2.

The steps of the signcryption workflow are:

0. **System Setup:** This can be done before the actual deployment. Here,

the KGS generates the system-wide parameters. All devices, that are supposed to interact within this system, have the public system parameters and a pair of one time device-registration keys embedded. This is described in [11, Section 3.C].

1.–4. **Device Registration:** *Device:* At this time, when new devices are introduced in the system, they first need to register at the KGS. This is done using their device-registration keys, which are one-time ID-based public/private keys. The device sends its registration public key, the long time public key derived from the ID, the serial number, and a signature, created using the one-time private key, to the KGS.

KGS: On receive of the device-registration message from a device, the signature is verified. Assuming the message holds a valid signature, the ID-based private key is calculated. Finally, this key is encrypted using the one-time public key of the device, signed by the KGS using its private key and send back to the device.

Device: The device-registration response is verified using the public key of the KGS and decrypted with the one-time private key. From now on the device and directly communicate with other devices in the network in a secure way[11, Section 3.C].

5. **Secure Data Transfer**

   a) **Sending Data:** First, based on the ID of the receiver, his public key is calculated, a random multiple of the systems base point is chosen and a shared secret for the packet is obtained using the Tate-Pairing. Utilizing a standard block cipher, with the shared secret as key, the clear text message is encrypted.

   In addition, a signature is calculated using the Tate-pairing and a standard hash function[1]. Finally, the device sends the compressed multiple of the base point, the encrypted message and the signature to the receiver.

   b) **Receiving Data:** The receiver also calculates the shared secret with the Tate-Pairing, but using its private key and the received multiple of the base point. Afterwards the packet can be decrypted using this secret and the same standard block cipher. Lastly, the signature is verified and the packet discarded, if the verification fails.

The exact calculation and procedure can be found in [11, Section 3.D-E].

The authors evaluated their signcryption scheme on a rather high powered system, Pentium IV @ 1.6 GHz. While this doesn't fall in the category of constrained devices I am trying to study, their measurements are still interesting for our overall analysis, due to the little amount of configuration and key management required at runtime. They also show substantial optimizations for their protocol, which allow to reduce the number of required pairing calculations by using a key caching scheme.

[1]The standard block cipher used in the actual protocol is AES. The hash function is MD5.

3.2 User-to-Node

External users of a constrained network may want to inspect specific nodes' monitored data, reconfigure the devices or send data to the device for other reasons. Authentication of these external users is critical for otherwise unmonitored devices. If no security precautions are taken, any external users, including attackers, can simply inspect the monitored data which may include confidential data or upload malicious code on the devices. This high-level communication pattern is depicted as b) in Figure 1.

The authentication framework[9], described by Yasmin, Ritter, and Wang, includes a scheme which allows limited devices in a network to authenticate external users. In particular, their scheme delivers three required tasks for user access: user authentication, access control and session key establishment.

The general work flow in this scheme is as follows:

1. The user registers at the base station and on success retrieves private key and system parameters in a secure fashion.

2. The user sends a signed data request to the desired sensor node.

3. The sensor node verifies the signature to check if the user is a) belonging to the system as a whole and b) if the identified user is allowed to access the requested information.

4. Lastly, the sensor node and user establish a session key which further allows encrypted communication in addition to the authentication.

Since the described framework already bases on IBC the use of an ID-based key establishment protocol would be an ideal fit for the part of session key establishment. The authors specifically suggest a one-pass ID-based key establishment protocol[15], which doesn't require any round trips as compared to the classic Diffie-Hellman key exchange. This is favorable, since constrained networks commonly have bigger round-trip times compared to the public Internet.

According to Yasmin, Ritter, and Wang, the problem of user revocation is twofold: a) revoking users with expired access time and b) malicious users. The first case can easily be handled by using an expire time in addition to the identity of a user to compute his ID-based private key. The case of malicious users on the other hand is handled by revocation lists. However, to keep these lists short, which is important for storage limited devices like wireless sensor nodes, the authors suggest to keep the default allowed access period for users rather short.

The authors compare their proposed solutions to two existing ones, RRUSAN, based on ECDSA signatures and DP^{2}AC, based on classic RSA signatures. Compared to the ECDSA method their scheme requires less computation to sign messages and verify signatures and while classic RSA signatures need very little computation for verification it comes with rather large keys and signatures as part of the message which has a negative influence on the storage limited devices.

3.3 Node-to-User

In the Node-to-User communication scheme, pattern c) in Figure 1, constrained

network nodes are communicating with outside users. This scenario is analyzed in detail by Oliveira, Kansal, Gouvêa, *et al.* in [10]. They target an application scenario, where deployed sensor nodes have sensing capabilities which are of interest to different, not directly related, users. By avoiding intermediary gateways, they allow direct communication of low-power sensor nodes to multiple users over the public Internet. Various signature mechanisms are compared within a setup of a Secure-TinyWebService / IP stack on two different sensor node hardware platforms.

Their design analysis includes consideration for both symmetric and asymmetric signatures. However, symmetric approaches are discarded due to key distribution and synchronization problems with the existing protocols, e.g. μTESLA, which aren't well suited for a scenario of a changing set of users sharing some sensor nodes over the public Internet.

In their analysis of available asymmetric methods to provide digital signatures, they decided against certificate-free schemes, like IBC. The TA in an IBC system always knows everybody's private key and can impersonate any user. In single self-contained sensor networks it might be acceptable that there is one entity which escrows everybody's private key. However, in a public shared network with different and changing parties, key escrow is unacceptable.

Certificate-free schemes, i.e. IBC schemes but without having the TA escrowing users' private keys, have been proposed before[16], but are also disqualified due to their very high computational complexity by the authors.

Which leads to the conclusion that certificate-based schemes are further analyzed, where Oliveira, Kansal, Gouvêa, *et al.* include not only classic DSA and its elliptic-curve version ECDSA but also more recent short-signature schemes, BLS[8] and ZSS[17].

For their usage scenario, a changing set of external users accessing an optimized web service on sensor nodes, they conclude that Schnorr signatures[18] or, if broader compatibility is needed, ECDSA is used. A Schnorr signature is digital signature scheme similar to ECDSA, however in comparison it is more efficient since it doesn't require expensive modular inverse computations.

Their measurements[10, p. 392] are visualized in Figure 3, which shows, that the additional energy required for the short-signatures has little influence on the communication cost. The additional cost for shorter signatures don't result in much energy savings on the radio communication side. However, on platforms where radio usage is more energy heavy, operation lifetime improvements could be gained from using short signatures like BLS and ZSS.

In addition, their practical evaluation on two different hardware platforms for sensor nodes, MSP[2] and AVR[3], shows that while on a high level, the computation complexity of various signature schemes are similar, the specific implementations on different hardware show relevant differences.

4 Conclusions and Outlook

Constrained networks show an interesting and challenging application for cryptogra-

[2] MSP430 16-bit @ 16 MHz, 116 KB ROM, 8 KB RAM

[3] ATmega128 8-bit @ 7 MHz, 128 KB ROM, 4 KB RAM

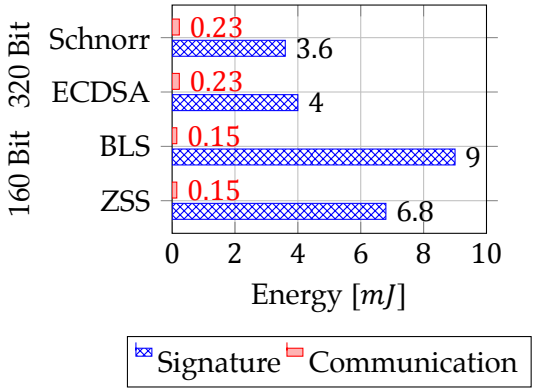

Figure 3: Energy comparison between signature computation and communication. Based on data from [10].

phy to find suitable security mechanisms. After studying three papers addressing security of different communication patters in constrained networks, IBC shows to be one popular solution. Especially for node-to-node communication and interaction with external users, it allows distributed authentication without further interaction with a central entity. This reduces the configuration to a minimum and simplifies security protocols and their validation.

In addition to classic asymmetric signatures like RSA or ECDSA, basic ID-based solutions don't need to transfer or store certificates for public keys on their device, which is ideal for storage constraint sensor nodes. Furthermore, IBSs fit the use case of authenticated multi-/broadcast very good, due to keeping the message size low and for their support for easy distributed authentication.

However, when it comes to a multi-party setup, where it is impossible to find a commonly trusted third party, classic identity-based solution come with the inherent *key escrow* property, which is unsuitable for this scenario. One option is to use normal asymmetric signatures or short signatures in these scenarios, however this loses the configuration and key management advantages of IBC based solutions.

There has been active research on creating IBC systems which don't have the key escrow problem with their key generation center, which started early after the initial working proposal for IBE with the description of certificateless public-key cryptography (CL-PKC)[16]. While the consideration of a certificateless scheme depends on the exact usage scenario, a good overview and comparison of public-key infrastructure (PKI), IBC and certificateless schemes can be found in [19].

After broader review of existing work in the area of constrained networks and providing security based on IBC, some practical evaluation is due.

First, it is planned to compare the proposed IBSs under fair conditions and verify the results of the papers, initially on desktop hardware and afterwards on a more constrained device. Secondly, possible integrations of IBC into existing protocols and applications will be investigated.

References

[1] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1645–1660, 2013.

[2] A. Gluhak, S. Krco, M. Nati, D. Pfisterer, N. Mitton, and T. Razafindralambo, "A Survey on Facilities for Experimental Internet of Things Research," *Communications Magazine, IEEE*, vol. 49, no. 11, pp. 58–67, Nov. 2011.

[3] H. Abelson, R. N. Anderson, S. M. Bellovin, J. Benaloh, M. Blaze, W. Diffie, J. Gilmore, P. G. Neumann, R. L. Rivest, J. I. Schiller, and B. Schneier, "The Risks of Key Recovery, Key Escrow, and Trusted Third-Party Encryption," 1997.

[4] D. R. Raymond and S. Midkiff, "Denial-of-Service in Wireless Sensor Networks: Attacks and Defenses," *Pervasive Computing, IEEE*, vol. 7, no. 1, pp. 74–81, Jan. 2008.

[5] A. Shamir, "Identity-Based Cryptosystems and Signature Schemes," English, in *Advances in Cryptology*, ser. Lecture Notes in Computer Science, vol. 196, Springer Berlin Heidelberg, 1985, pp. 47–53.

[6] D. Boneh and M. Franklin, "Identity-Based Encryption from the Weil Pairing," English, in *Advances in Cryptology — CRYPTO 2001*, ser. Lecture Notes in Computer Science, vol. 2139, Springer Berlin Heidelberg, 2001, pp. 213–229.

[7] P. S. Barreto, H. Y. Kim, B. Lynn, and M. Scott, "Efficient Algorithms for Pairing-Based Cryptosystems," in *Advances in Cryptology — CRYPTO 2002*, ser. Lecture Notes in Computer Science, vol. 2442, Springer Berlin Heidelberg, 2002, pp. 354–369.

[8] D. Boneh, B. Lynn, and H. Shacham, "Short Signatures from the Weil Pairing," in *Advances in Cryptology — ASIACRYPT 2001*, ser. Lecture Notes in Computer Science, vol. 2248, Springer Berlin Heidelberg, 2001, pp. 514–532.

[9] R. Yasmin, E. Ritter, and G. Wang, "An Authentication Framework for Wireless Sensor Networks using Identity-Based Signatures," in *Computer and Information Technology (CIT), 2010 IEEE 10th International Conference on*, 2010, pp. 882–889.

[10] L. B. Oliveira, A. Kansal, C. P. Gouvêa, D. F. Aranha, J. López, B. Priyantha, M. Goraczko, and F. Zhao, "Secure-TWS: Authenticating Node to Multi-user Communication in Shared Sensor Networks," *The Computer Journal*, vol. 55, no. 4, pp. 384–396, 2012.

[11] H.-H. So, S. Kwok, E. Lam, and K.-S. Lui, "Zero-Configuration Identity-Based Signcryption Scheme for Smart Grid," in *Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on*, 2010, pp. 321–326.

[12] A. Perrig, R. Szewczyk, J. D. Tygar, V. Wen, and D. E. Culler, "SPINS: Security Protocols for Sensor Networks," *Wireless Networks*, vol. 8, no. 5, pp. 521–534, Sep. 2002.

[13] R. Yasmin, E. Ritter, and G. Wang, "An Authentication Framework for Wireless Sensor Networks Using Identity-Based Signatures: Implementation and Evaluation," *IEICE Transactions on Information and Systems*, vol. E95.D, no. 1, pp. 126–133, 2012.

[14] Y. Zheng, "Digital Signcryption or How to Achieve Cost(Signature & Encryption) $\ll$ Cost(Signature) + Cost(Encryption)," in *Advances in Cryptology — CRYPTO '97*, ser. Lecture Notes in Computer Science, vol. 1294, Springer Berlin Heidelberg, 1997, pp. 165–179.

[15] M. C. Gorantla, C. Boyd, and J. M. González Nieto, "ID-based One-pass Authenticated Key Establishment," in *Proceedings of the Sixth Australasian Conference on Information Security - Volume 81*, ser. AISC '08, Wollongong, NSW, Australia: Australian Computer Society, Inc., 2008, pp. 39–46.

[16] S. S. Al-Riyami and K. G. Paterson, "Certificateless Public Key Cryptography," in *Advances in Cryptology - ASIACRYPT 2003*, ser. Lecture Notes in Computer Science, vol. 2894, Springer Berlin Heidelberg, 2003, pp. 452–473.

[17] F. Zhang, R. Safavi-Naini, and W. Susilo, "An Efficient Signature Scheme from Bilinear Pairings and Its Applications," in *Public Key Cryptography – PKC 2004*, ser. Lecture Notes in Computer Science, vol. 2947, Springer Berlin Heidelberg, 2004, pp. 277–290.

[18] C.-P. Schnorr, "Efficient Signature Generation by Smart Cards," *Journal of Cryptology*, vol. 4, no. 3, pp. 161–174, 1991.

[19] A. Ahmad, A. Biri, H. Afifi, and D. Zeghlache, "TIBC: Trade-off between Identity-Based and Certificateless Cryptography for future Internet," in *Personal, Indoor and Mobile Radio Communications, 2009 IEEE 20th International Symposium on*, Sep. 2009, pp. 2866–2870.