# KEDS: Decentralised Network Security for the Smart Home Environment

Justin King-Lacroix(✉) and Andrew Martin

Department of Computer Science, University of Oxford,
Wolfson Building, Parks Road, Oxford OX1 3QD, UK
{Justin.King-Lacroix,Andrew.Martin}@cs.ox.ac.uk

**Abstract.** The increasingly wide deployment of smart grid technologies in the home has resulted in home automation networks becoming multi-stakeholder, with the number of stakeholders increasing over time.

However, the technologies underpinning these networks universally feature a heavily centralised security model, with policy data held on privileged machines that are both security- and availability-critical. On a multi-stakeholder network, no single stakeholder can be trusted with the authority to operate such privileged machines.

This paper presents a novel network architecture for multi-stakeholder networking. It also proposes a set of modifications to ZigBee, an emerging industry standard in the smart grid domain, that would cause it to conform to this architecture. These are used as the basis for an example application: the smart home.

## 1 Introduction

The term smart grid refers to the increasing instrumentation of electricity infrastructure with Internet-connected sensors. These sensors report energy consumption data in real time to utility providers, in order to aid both prediction and management of electricity demand. However, this real-time reporting raises security and privacy concerns [1], especially as the granularity of the reported data approaches the level of individual homes.

Sensors in the home for measuring electricity consumption are referred to as smart meters. They contain embedded microprocessors, and are usually connected to a dedicated backhaul network operating alongside electrical distribution lines. These meters are at the hub of the smart home environment introduced by the widely-cited NIST Framework and Roadmap for Smart Grid Interoperability Standards [2]. Newer home automation solutions are able to integrate into this environment, in order to exchange energy-management data, potentially including per-device energy consumption information, with the electricity provider (via the smart meter). Other utilities, such as gas and water, are also beginning to deploy smart meters for real-time monitoring. These meters must somehow report information to their respective operators; the consumer's Internet connection and the electricity provider's backhaul are the two primary means of achieving this.


J. Cuellar (Ed.): SmartGridSec 2014, LNCS 8448, pp. 63–78, 2014.
DOI: 10.1007/978-3-319-10329-7_5

Smart home networks are thus multi-stakeholder networks of a novel kind. The stakeholders involved cannot completely trust each other, and yet the devices they control must exchange high-level services in order to fulfil their operational goals. Additionally, each stakeholder controls only one or a handful of devices, and so the network cannot be decomposed into stakeholder-specific subnetworks.

### 1.1 Contributions Made in this Paper

The contributions of this paper are:

- To highlight the challenge presented by multi-stakeholder networking.
- To describe a key-exchange protocol, KEDS, for low-power embedded networks.
- To describe a network architecture for multi-stakeholder networks, with no central control points, based on KEDS.
- To combine the preceding two contributions into a set of changes to ZigBee, with a view towards its application in a smart home setting.

### 1.2 Structure of this Paper

The remainder of this paper is structured as follows: In Sect. 2, we outline existing approaches to network security and multi-stakeholder networking, and examine why these are inappropriate for the smart grid case. We then outline the security model and relevant features of ZigBee, an emerging industry standard for smart grid home networks.

In Sect. 3, we highlight the key security and performance requirements of a multi-stakeholder network, and propose a network architecture and security model for network-layer protocols which respects those requirements. We then outline a series of modifications to ZigBee in order to implement that architecture. Section 4 evaluates the architecture and protocol presented, discussing its implications for the security and performance of the network, as well as its potential operational overhead.

In Sect. 5, we return to the smart grid, remarking upon the feasibility of the complete removal of trusted third-parties in this context. Finally, Sect. 6 concludes the paper, and outlines our next steps.

Throughout this paper, terminology from the well-known OSI model for communication systems will be used.

## 2 Background

### 2.1 Existing Approach to Network Security

Network security has generally assumed a strict separation between insiders – people and machines within the network perimeter – and outsiders – those external to it. Network security technologies deployed in the home – in particular,

those underlying Wi-Fi [3] and ZigBee [4] – are built on the assumption that outsiders should be entirely denied network access, while insiders are admitted and treated identically.

Furthermore, such access-control decisions are based on information in central, privileged directories of security principals and authentication metadata. The machines hosting these directories – the Access Point in the Wi-Fi case, or the Trust Center for ZigBee – exercise total control over the network; their owner is assumed to be its owner.

### 2.2 Existing Approaches to Multi-stakeholder Networking

Multi-stakeholder networking has seen implementation in two contexts: Internet Network Access Points, and military systems. In both cases, the focus has been on interconnecting a small number of large networks controlled by mutually-distrusting entities. The technologies in use were developed specifically for this purpose, and do not generalise to other application domains.

**Network Access Points.** Internet Network Access Points, and their equivalents in large data centres, have always had a single purpose: the routing of traffic across the Internet. The Border Gateway Protocol [5] operates at these junction points to interconnect the networks of the various organisations present. Participants in these systems make a strong trust assumption: that it is in each stakeholder's best interests to maximise the efficiency of the routing infrastructure. More recent developments [6] weaken this trust assumption by cryptographic means, introducing a trusted third-party certificate issuer which validates route announcements. However, again, this is a solution specific to the use case: a third-party authority already exists for the assignment of Internet addresses, the Internet Assigned Numbers Authority, IANA.

**Military Systems.** Relevant military research focuses on three areas: the routing of packets across hostile (or potentially-hostile) terrain [7], the interconnection of networks with multiple levels of security [8,9], and the formation of ad-hoc wireless networks in a disaster-relief scenario [10]. In all cases, the basic problem is the same as for Network Access Points: the routing of packets over an internetwork [11]. The issue of higher-level services is rarely considered.

In disaster-relief, some work has been done on information exchange between participating organisations. However, this work addresses mainly the policy concerns surrounding the exchange of information between civilian and military stakeholders [12], with little done on the security architecture of the networks being used.

### 2.3 ZigBee

ZigBee [4] is a network protocol specification designed for low-power wireless mesh networking in the embedded space, and is an emerging industry standard for smart grid home networks. It covers the network layer of the stack,

and above (excluding the application layer), with little division between layers. IEEE802.15.4 [13] provides the link layer and below.

Its security model is based on symmetric cryptography, with key distribution the responsibility of a central Trust Center. In high-security mode, packets are encrypted and integrity-protected with keys of two types: the *network key*, a network-wide secret which all nodes must possess, and *link keys*, which are used for pairwise communication between nodes. At network join time, each node must be provisioned with a Trust Center Link Key, which is used for all communication with the Trust Center, including distribution of further keys for communication with the rest of the network.

Nodes are arranged in a tree structure, with each router a node in the tree, and end-devices at its leaves. The root node is known as the Coordinator; the Coordinator is also usually (though not necessarily) the Trust Center. Joining a ZigBee network is a complex operation: the join protocol has multiple branches, and requires a large number of network round-trips (12, in the worst case). The specification mandates that the Trust Center keep a registry of currently-active devices, kept up-to-date by information messages from routers as nodes join and leave. It may at any time instruct a router to eject a node from the network.

Clearly, the Trust Center is a single point of failure for the entire network. It possesses all keys currently in use, and so is capable of decrypting all traffic and impersonating any node, and additionally has the right to admit nodes to or exclude them from the network. Thus, in a multi-stakeholder context, whichever stakeholder controls the Trust Center controls all communications on the network.

**ZigBee Smart Energy Profile.** The ZigBee Smart Energy Profile [14] (SEP) specification introduces a requirement for each node to possess a key pair for use in elliptic-curve cryptographic (ECC) protocols, serving as its cryptographic identity. Link keys can thus be negotiated pairwise between nodes, without potential for eavesdropping by the Trust Center. However, not all SEP operations are mandated to use link keys for security.

## 3 Modern Multi-stakeholder Networks

Multi-stakeholder networks are characterised by the presence of multiple entities with disparate and *potentially competing* interests. In such an environment, if one such entity is granted administrative control of the infrastructure, necessarily that entity gains the ability to prioritise its interests over those of the others, potentially to their detriment. Such a network therefore should not contain any single points of control, since such a point of control would give administrative control of the network infrastructure to its owner.

The introduction of a trusted third party is a natural solution to this problem. However, this presupposes the existence of an entity whose interests do not compete with the other stakeholders on the network, which is unlikely to be

the case the smart grid context. Moreover, devices controlled by such trusted third-parties present an obvioustarget for attack.

The most robust solution, therefore, is to distribute security responsibilities over all stakeholders. Since, in the smart home environment, each stakeholder only controls one (or a small number) of nodes, security responsibilities must therefore be distributed over all nodes in the network. In general, this can be done by ensuring all traffic is encrypted with keys known only to its sender and receiver.

On the wider Internet, this is done by means of public-key cryptography. However, prior to the advent of ECC, public-key operations consumed too much CPU power to be usable on the resource-constrained embedded systems that dominate the smart grid. ECC is now a mature and widely-deployed technique, and has been implemented on very low-power devices [15], permitting high-security communications even under strict resource constraints.

**Structure of this Section.** The remainder of this section will describe our proposal. We begin with its position in the software stack, with some mention of interfaces upwards and downwards. We then describe the network-wide key management structure, along with its consequences.

Following this description, we introduce two novel mechanisms to support the key management structure we present: key-exchange with data stapling, and cryptographic delegation. We then apply our proposal to ZigBee, outlining a series of modifications that we propose to make.

### 3.1 Proposed Architecture

**Network Stack Model and Interfaces.** Our proposal covers the network layer of the software stack. IEEE802.15.4 will provide the data link and physical layers, given its wide deployment in the smart grid domain.

We expect implementations to adhere to a mostly-open trust model: all code on a given node should trust the network (and below) layers with unencrypted data. The reason for this is simple: encryption of packets will be done by the data link layer. A minor exception is that the network layer need not expose encryption keys to higher layers.

**Key Management.** Our key management architecture is straightforward:

1. Each node must possess an ECC key pair, which forms its identity.
2. A pair of nodes wishing to communicate must first establish a shared secret (for use in encryption and integrity-protection) using those ECC keys.
3. There are no network-wide shared secrets.

We mandate that all key management be done at the network layer; higher layers should delegate this task downwards where possible. As a result, all communication between the same pair of nodes will use the same key to communicate.

Finally, we require that all network traffic use encryption and integrity protection, using the aforementioned pairwise keys; it is the responsibility of the network layer to arrange for this to occur, with the actual cryptographic work done by the data link layer.

There is an important subtlety related to item 2 above: a new key must be established for *each pair of communicating devices*, whether those devices are neighbours or not. The KEDS protocol below is designed both for use as a fast network join protocol, and for bootstrapping secure channels between nodes several routing hops apart. If key establishment is restricted only to neighbouring routing hops, communications will be vulnerable to attack by intervening routing nodes, and end-to-end security is lost.

**Broadcast/Multicast Traffic.** In this rigid pairwise keying model, broadcast and multicast messages present a challenge. The naïve message broadcast protocol in this environment has router nodes re-encrypt a message once for each neighbouring node to which it is retransmitted. In order to protect against modification by intervening routers, messages are required to be digitally signed by their originators.

This protocol is only suitable for infrequent broadcasts, due both to the processor and radio overhead it imposes on routers, and the large number of public-key transactions required by the rest of the network. For more frequent broadcasts, and any multicasts, an alternative mechanism is necessary. The TESLA [16] protocol is ideally-suited to this use, with initialisation data distributed using the naïve protocol for broadcasts, or unicast transmissions in the multicast case.

### 3.2 Key-Exchange with Data-Stapling (KEDS)

Communication between nodes must always begin with a key-agreement phase. Diffie-Hellman (DH) is the oldest and most popular protocol for this purpose. We have selected one of its ECC-based descendents, ECMQV (as described in the ZigBee SEP specification [14]) for our purposes here, due to the low computational requirements of ECC algorithms.

DH-based protocols consist of four messages (see Fig. 1). We propose a data stapling extension to the protocol: in each key-agreement message, we introduce a field for additional data, which is encrypted and integrity-protected using the resultant key.

The first message is a special case: since neither side yet possesses enough information to derive the resulting key, data cannot be encrypted. However, integrity-verification data can be generated and included in the second message, alongside its stapled data. The SD1DV (for 'Stapled Data 1 Delayed Verification') field is included for this purpose.

### 3.3 Cryptographic Delegation

There may still remain classes of devices for which the frequency of public-key transactions in KEDS is too high. For these devices, we introduce the following

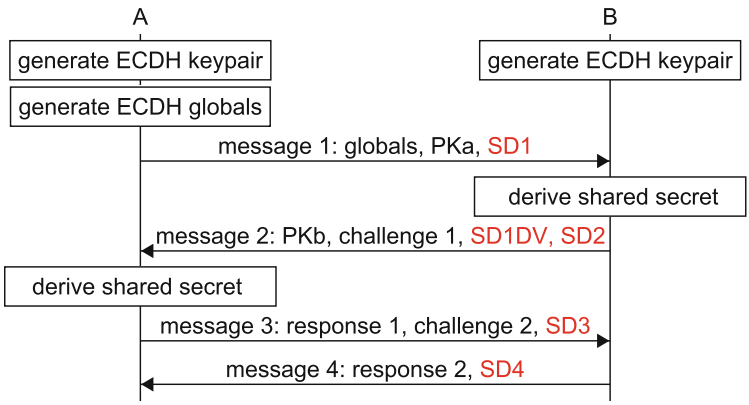


**Fig. 1.** The ECMQV protocol. Fields in red are added by KEDS. Note that SD1 is necessarily unencrypted.

feature: *cryptographic delegation*. A node may issue a digital certificate designating another (presumably computationally stronger) node as its *cryptographic delegate*. This certificate confers upon the delegate the right to conduct key-agreement transactions on its delegator's behalf, and can be sent to a prospective communication partner at the time it issues a key-agreement request (KEDS message 1).

Delegation clearly leaves the delegator vulnerable to eavesdropping and impersonation attacks: a delegate necessarily possesses all keys it negotiates on its delegate's behalf. A node can therefore issue a *revocation certificate* for a delegate it no longer trusts. This certificate can be transmitted immediately to existing peers, which must, upon its reception, immediately begin negotiating new keys.

Cryptographic delegation has been previously applied in grid computing [17], where X.509 proxy certificates allow users to issue a time-limited permission for jobs to execute on their behalf without requiring explicit authorisation for every run. However, the approach to revocation – that of timed expiry of certificates – assumes globally synchronised clocks, which is not a safe assumption for a network of embedded systems.

### 3.4 Modifications to ZigBee

The ZigBee Smart Energy Profile already introduces many of the elements in our protocol, chiefly the use of elliptic-curve cryptography to negotiate pairwise keys between nodes. However, security responsibilities are still largely centralised, since link keys are negotiated at the application layer, and only used for certain operations (with many transactions still using the network key), and the Trust Center additionally still exercises control over admissions to the network.

The KEDS architecture requires security responsibilities to be fully distributed. The following modifications are thus necessary to ZigBee to produce a protocol that conformed to it:

1. All current group keying – particularly the network key – are eliminated. A *multicast* key type is added to support TESLA operations.
2. The ECMQV key-agreement protocol introduced by SEP becomes the sole and mandatory key-exchange mechanism, to be used both with neighbouring nodes (during a network-join) and distant nodes (after the network-join is complete).
3. The Trust Center is entirely removed.
4. Of the various frame security levels supported by ZigBee, all except AES-CCM-128 (which is mandatory in IEEE802.15.4) are disallowed.
5. The ZigBee join protocol is deprecated in its entirety. Instead, the KEDS mechanism would be used, using data stapling to transmit network configuration information.
6. The broadcast and multicast mechanisms from Sect. 3.1 are added.

**Backwards-Compatibility.** As presented, KEDS breaks compatibility with existing ZigBee software. This is deliberate, since the ZigBee security model is incompatible with that of KEDS. However, backwards-compatibility could be implemented in the following way: a KEDS node could act as the ZigBee Trust Center for a network subtree of which it is the root. It would also act as KEDS cryptographic delegate for all devices in that subtree.

ZigBee devices need not join as end-devices; routers are also easily supported. However, only some of the possible branches of the ZigBee join protocol can be allowed: MAC-layer associations would not, only the ZigBee NWK join. Naturally, the KEDS frame security requirements would also necessarily be extended to ZigBee nodes.

## 4 Implications

In this section, we discuss the implications for the security, performance, and energy consumption of a network based on KEDS, as well as examining ease of administration and development. For this purpose, it is worth remarking on a similarity between smart home and wireless sensor networks: both network types consist largely of embedded devices under similar constraints, permitting discussion of one to be applied easily to the other. We will therefore borrow the rich set of terminology available for the evaluation of wireless sensor networks from an overview of the field by Lee *et al.* [18]. The definitions from that paper that we will be reusing are reproduced in Table 1.

### 4.1 Security

Much of this paper has been devoted to highlighting security issues, since these are a driving force in the design of KEDS. Much of the security impact of what we have proposed has therefore already been covered.

**Table 1.** Definitions from Lee et al. [18]

| Term | Definition |
|---|---|
| Confidentiality | Nodes should not reveal any data to unintended recipients |
| Integrity | Data should not be changed between transmissions due to the environment or malicious activities |
| Data freshness | Old data should not be used as new (i.e., prevent replay attacks) |
| Authentication | Data used in decision-making processes must originate from the correct source |
| Robustness | When some nodes are compromised, the entire network should not also become compromised. The quantitative value with which this requirement should be satisfied depends on the application |
| Self-organization | Nodes should be independent and flexible enough to be self-organizing (autonomous) and self-healing (failure tolerant) |
| Availability | The network should not fail frequently |
| Time synchronization | Collaborative node applications need time synchronization. Time synchronization protocols should not be manipulated to produce inaccurate time |
| Secure localization | Nodes should be able to accurately and securely acquire location information |

Distributing security responsibilities requires security policy decisions to be made and enforced on each device, since there is by design no longer a central decision or enforcement point on the network. The removal of this single point of failure is clearly an improvement in robustness, but also in self-organisation.

Pairwise keying is beneficial from the standpoints of confidentiality, integrity, and authenticity: no node is capable of altering or forging messages, and all messages are confidential to the nodes exchanging them.

The lack of global secrets (or, indeed, global policy) or central control nodes to compromise creates an equivalence between insider and outsider attacks, and makes both difficult.

Network-layer key management has a subtle privacy advantage over the application-layer management favoured in ZigBee: packets need no longer indicate which key they are using (since the source and destination node addresses uniquely determine this). As a result, an observer cannot determine the application to which the packet belongs, eliminating a class of traffic-analysis attack.

### 4.2 Performance

Evaluation of the performance of a network of embedded devices centers on the consumption of various resources in a limited environment. Most important are CPU time, memory, and energy; it is these three which we consider here.

Pairwise keying has a substantial disadvantage compared to group keying: its memory requirement scales linearly with the size of the network. Put another way, each node must have sufficient key storage to hold one key (plus associated metadata) for each other node with which it will communicate. This memory must also be powered, creating an associated energy overhead.

Each public-key transaction also incurs an energy cost. Since one such transaction must be performed for each pair of communicating devices, the network-wide energy cost of key agreement scales as the square of the size of the network, in the worst case.

Encryption and integrity-protection of every packet also costs resources: both the CPU time of actually performing the cryptographic operations, and the energy required to power it during those operations. Integrity-protection additionally reduces the available application data per packet, potentially requiring more packets to be transmitted, at a cost of yet more energy.

Broadcasts have a particularly high cost. In the naïve protocol, each broadcast packet transmission requires re-encryption by every intervening router, for every peer to which they are to be retransmitted; this costs both processor time, for the large number of cryptographic operations, and energy, for both that processor time and the large number of packet retransmissions. TESLA operations are slightly different: once the protocol has been bootstrapped, each TESLA message requires two packet broadcasts (the first being the message itself, and the second its TESLA key). However, unlike in the naïve case, these broadcasts need not be re-encrypted; the energy cost is almost entirely due to radio transmission. Additionally, the total number of radio transmissions is reduced compared to the naïve protocol, since routers need only retransmit each packet once.

ZigBee already incurs some of these costs: the Smart Energy Profile requires all packets to be encrypted and integrity-protected, and introduces ECC protocols (including ECMQV) to ZigBee networks. TESLA is being considered for use in vehicular networks [19], although was originally designed for wireless sensor networks.

Despite requiring public-key cryptography, use of KEDS has the potential to reduce energy consumption. The ZigBee network join protocol requires 12 round-trips in the worst case. By contrast, a network join using KEDS requires three round-trips, and a security association with another node once joined only two. Additionally, data-stapling can reduce the total number of data packets that need to be transmitted. Finally, the cryptographic delegation mechanism can permit particularly low-powered devices to offload most of their KEDS processing to a more powerful neighbour.

### 4.3 Operational Overhead

Development for and administration of distributed networks is generally considered more complex and difficult than their centralised counterparts. This is, however, not always the case. In particular, the cryptographically-strong node identities afforded by public-key cryptography permit both developers and

administrators to reason about the identities of those nodes with a high degree of confidence: unlike MAC addresses, private keys cannot be forged or spoofed (although they can be stolen). Additionally, nodes can be deployed without preloading of symmetric keys, since those keys can be safely sent over the network.

The combination of KEDS and a single security mode vastly simplifies development and deployment. Mismatches between supported cipher suites can no longer occur, and the KEDS network join protocol is vastly simpler than that of ZigBee. Application programmers are no longer required to manage cryptographic keys, since this responsibility is delegated to the network layer; they need only implement their application's access-control policy.

The removal of the Trust Center eliminates the central registry of devices it would otherwise maintain. This may actually be a benefit: the currency of such a registry must be enforced at network join points, which in the case of ZigBee are at every router. This requires all routers to be trusted; the elimination of the registry thus also eliminates this requirement.

Finally, the cryptographic delegation feature has a more subtle advantage: on a multi-stakeholder network, it indicates strongly that the delegating node trusts the target node, and thus that the stakeholders involved trust each other in a similar way. This can be used as a form of vouching [20].

## 5 A Note on the Smart Grid

Much of the work in this paper has been devoted to removing the need for trusted third-parties in order to bootstrap security. However, on the smart grid, contact with devices of unknown provenance or type are expected to be commonplace. There must be some way, therefore, for two nodes to be able to assert their hardware capabilities to each other in a trustworthy manner. As a result, we expect that there will be some kind of certification of device characteristics, either by national or supranational regulatory authorities, industry bodies, or agreements between stakeholders.

Note that while these entities are trusted third-parties, they are of a different kind to the ones hitherto discussed. The protections that we introduce to the network environment defend against device-impersonation and man-in-the-middle attacks launched from active participants on the network. The certification required in the smart grid case, and the credentials digitally expressing that certification, do not come from such active participants, but from external entities, and the range of attacks they can launch is different: their certificates can make arbitrary assertions about the *capabilities* and *properties* of a device, but no more (and in particular, they cannot impersonate a device, nor compromise the secure channels to which it is party).

### 5.1 An Illustrative Example

To make these ideas concrete, let us consider an example home automation network. In this example, we will make the following simplifying assumptions:

- Each device belongs to a single stakeholder. This allows us to neglect issues of operating system security, which would otherwise be relevant towards isolating colocated stakeholders from each other.
- The home is free-standing (that is, not a flat or apartment). This eliminates the building administrator as a potential stakeholder, as well as possible interactions between flats in the same building.
- Its owner is its sole resident. This eliminates other residents as potential competing stakeholders.

It is important to note that the cases eliminated from this example are *not* outside the scope of our protocol; they are simply excluded from this example for the purpose of clarity.

Our example network will consist of the following devices: (see Fig. 2)

**An Internet router/gateway device (IGD):** a mains-powered device which connects the automation network to the Internet.

**Smartphone:** a battery-powered device with a powerful CPU and a rich user interface.

**Electricity meter:** a mains-powered device with some processing power, but little or no user interface. It reports power usage to the electricity provider, via a dedicated connection to the grid's backhaul.

**Water meter:** a battery-powered device with a low-power CPU and little or no user interface. It reports water usage to the water provider over the Internet, via the IGD.

**Gas meter:** a battery-powered device with a low-power CPU and little or no user interface. It reports gas usage to the gas provider over the Internet, via the IGD.

**Hot water tank:** a mains-powered device which attempts to heat water when electricity prices are lowest. The electricity provider can toggle this device for demand-shedding purposes.

**Washing machine:** a mains-powered device which reports maintenance data to its manufacturer, attempts to heat water when prices are low, without delaying the washing too long. The electricity provider can toggle this device for demand-shedding purposes, but only during certain phases of cycle, and only for so long.

**Heating system:** a mains-powered device whose main energy source is burning gas. It runs on a schedule, though can be toggled by the electricity provider for demand-shedding purposes, provided the house stays close to the set temperature.

**Lights and light switches throughout the house:** all mains-powered, but with low-power CPUs and no user interface.

Table 2 shows the stakeholder considered to 'own' each device, as well as notes on other stakeholders with an interest in its function. Where a stakeholder can control a device it does not own, it is assumed to do so via one that it does – so, for example, commands from the electricity provider to toggle the hot water tank or washing machine should originate from the electricity meter.

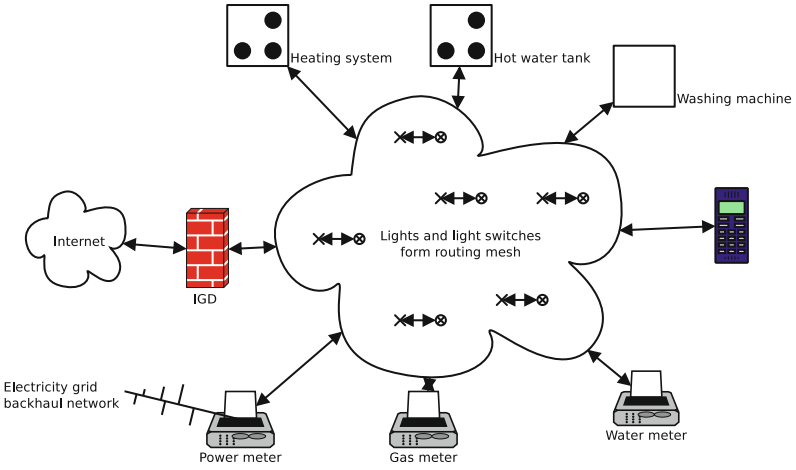


**Fig. 2.** Our example home network.

The homeowner is a partial exception: she is the only stakeholder with a human presence in the home. She can thus operate devices via their physical control panels as well as via her smartphone. She is also capable of bringing devices temporarily into close physical proximity to one another (for example, tapping her smartphone on an NFC pad on the hot water tank), which may be necessary for some authentication/KEDS sequences.

Security associations between devices are relatively clear: lights and light switches must be paired with each other; the hot water tank, heating system, and washing machine must be paired with the electricity meter; and the water meter, gas meter, and washing machine must be paired with the IGD. Some or all of these devices may also be paired with the smartphone, depending on the status information they expose to the user.

Each such pairing requires different information to be supplied, during the KEDS exchange, to authenticate the channel. Pairings involving the electricity meter might require it to present some certificate asserting its presence at and control over the relevant address or electrical connection point (if there are several electrical providers acting at the same address). That between the heating system and the electricity provider might require some similar certificate from the systems manufacturer, asserting that it can actually respond to load-shedding commands and indicating the maximum load that can be shed. The connection between the heating system and smartphone might not need any certificates at all, instead relying on some assertion of physical proximity (such as an NFC pad) to pair. The smartphone could then even issue a certificate vouching for the physical presence of that heating system in the home during some other KEDS transaction (such as the aforementioned between heating and power meter).

**Table 2.** 'Stakeholder map' of our example network.

| Device | Owner | Notes |
|---|---|---|
| IGD | Homeowner | |
| Smartphone | Homeowner | |
| Electricity meter | Electrical provider | |
| Water meter | Water provider | |
| Gas meter | Gas provider | |
| Hot water tank | Homeowner | |
| Washing machine | Manufacturer | Operated by homeowner, partial control by electricity provider |
| Home heating system | Homeowner | Partial control by electricity provider |
| Lights | Manufacturers | Operated by homeowner |
| Light switches | Manufacturers | Operated by homeowner |

Security associations also allow for find-grained access control: the IGD could be configured to only allow the washing machine, water meter, and gas meter access to the Internet, refusing any KEDS association requests from other devices. Equally importantly, the confidentiality and integrity requirements on all communications mean that irrespective of the path through the mesh that any given message takes, its contents remain both secret and unalterable, either by a malicious or faulty device.

## 6 Conclusions and Future Work

We have presented a novel network architecture for multi-stakeholder networks. This architecture distributes security responsibilities among the nodes that make up the network, rendering insider attacks as difficult as attacks by outsiders by eliminating trusted third-parties on the network. We have also discussed how we expect this architecture to be implemented by modifying ZigBee, an emerging industry standard in smart grid networks. This discussion included the implications of such a network from the perspectives of security and performance, with some additional discussion on administrative and development complexity. This discussion ended with some comments specific to the smart grid, including an example application to the smart home; one comment is that trusted third-parties may still be necessary, but only in order to lend weight to devices' assertions of their hardware capabilities.

Our next step will be to implement the proposed protocol, and perform real-world power, resource-consumption, and throughput measurements, to support the predictions made in this paper.

## References


1. Paverd, A.: Trustworthy remote entities in the smart grid. In: Proceedings of the ACM Symposium On Applied Computing (SAC) Student Research Competition, pp. 9–10 (2013)
2. National Institute of Standards and Technology (NIST). NIST special publication 1108R2: NIST framework and roadmap for smart grid interoperability standards, release 2.0. Technical report (2012)
3. IEEE: Standard for Local and metropolitan area networks, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. IEEE Std 802.11-2012
4. Alliance, Z.: ZigBee Specification (2008)
5. Gregori, E., Improta, A., Lenzini, L., Rossi, L., Sani, L.: BGP and inter-AS economic relationships. In: Domingo-Pascual, J., Manzoni, P., Palazzo, S., Pont, A., Scoglio, C. (eds.) NETWORKING 2011, Part II. LNCS, vol. 6641, pp. 54–67. Springer, Heidelberg (2011)
6. Butler, K., Farley, T., McDaniel, P., Rexford, J.: A survey of BGP security issues and solutions. Proc. IEEE **98**(1), 100–122 (2010)
7. Gohari, A.A., Pakbaz, R., Melliar-Smith, P.M., Moser, L.E., Rodoplu, V.: RMR: reliability map routing for tactical mobile ad hoc networks. IEEE J. Sel. Areas Commun. **29**(10), 1935–1947 (2011)
8. Gibson, T.: An architecture for flexible multi-security domain networks. In: Proceedings of the Network and Distributed Systems Security Symposium, San Diego, February 2001
9. Schumacher, H.J.J., Ghosh, S., Lee, T.S.: Top secret traffic and the public ATM network infrastructure. Inf. Syst. Secur. **7**(4), 27–45 (1999)
10. Mason, A.R.: Exploring of wireless technology to provide information sharing among military, United Nations and civilian organizations during complex humanitarian emergencies and peacekeeping operations. Master's thesis, Naval Postgraduate School, March 2003
11. Hughes, B., Sharpe, T.: NATO Tacoms. In: MILCOM, IEEE, pp. 1–7 (2006)
12. Wentz, L.: An ICT primer: Information and communication technologies for civil-military coordination in disaster relief and stabilization and reconstruction. Technical report, National Defense University Center for Technology and National Security Policy, Washington, DC, USA (2006)
13. IEEE: Standard for Local and metropolitan area networks, Part 15.4: Low-Rate Wireless Personal Area Networks. IEEE Std 802.15.4-2011
14. Alliance, Z.: ZigBee Smart Energy Profile Specification (2011)
15. Gupta, V., Millard, M., Fung, S., Gura, N., Eberle, H.: Sizzle: a standards-based end-to-end security architecture for the embedded Internet. In: IEEE International Conference on Pervasive Computing and Communications, pp. 247–256 (2005)
16. Perrig, A., Song, D., Canetti, R., Tygar, J.D., Briscoe, B.: Timed Efficient Stream Loss-Tolerant Authentication (TESLA): Multicast Source Authentication Transform Introduction. RFC 4082 (Informational), June 2005
17. Welch, V., Foster, I., Kesselman, C., Mulmo, O., Pearlman, L., Gawor, J., Meder, S., Siebenlist, F.: X.509 proxy certificates for dynamic delegation. In: Proceedings of the 3rd Annual PKI R&D Workshop (2004)

18. Lee, J., Leung, V., Wong, K., Chan, H.: Key management issues in wireless sensor networks: current proposals and future developments. IEEE Wirel. Commun. Mag. **14**(5), 76–84 (2007)
19. Hartenstein, H., Laberteaux, K.: A tutorial survey on vehicular ad hoc networks. IEEE Commun. Mag. **46**(6), 164–171 (2008)
20. Li, F., Mittal, P., Caesar, M., Borisov, N.: SybilControl. In: Proceedings of the 7th ACM Workshop on Scalable Trusted Computing, pp. 67–78, October 2012