Hardware Security Device Facilitated Trusted Residential Energy Services

John Zic, Julian Jang, Dongxi Liu, Chen Wang, Martin de Groot
CSIRO ICT Centre
Sydney, Australia

***Abstract*—We report on our experiences in developing a hardware based security solution for a demonstration prototype of a novel, smart-grid enabled energy services delivery model. This model is based on establishing and maintaining trusted, secure dynamic collaborations with agreed, policy driven usage and billing metrics between energy service providers, energy supply companies, and the consumers' residences. In this paper, we describe how we used a specialized secure hardware device, issued by an energy service company and containing a set of tokens and cryptographic functions can be used to ensure that all transactions between the energy service company and the consumer are trusted, secure and private to the participants.**

I. Introduction

The delivery of smart-grid enabled residential scale energy services relies on the interplay and interaction between three types of entities: Energy Services Companies (or *ESCOs*), energy providers, and (of course) residential customers. Each ESCO offers aggregation services to an Energy Provider and management services to Residential Consumers, and until recently, ESCOs have only managed large sites and not individual residences. This is because of two reasons. First, since current ESCOs run and configure their business' using tailored, manual methods, it is difficult for them to scale up to the required residential (or SME) numbers and still maintain profitability. Second, invididual residential consumers do not have adequate knowledge or willingness to effectively manage their energy usage on a long term, sustainable basis. To have ESCOs operate at the residential scale requires (from the ESCO and Energy Provider point of view) the wide spread deployment of low cost, ubiquitously accessible networking infrastructure upon which a rich set of services can be delivered. It also requires that consumers are able and interested to use these services. Ensuring that the consumers have clear financial and "green" incentives to agree to participate are both important drivers for the uptake of these services. However, consumer adoption of such a system requires that they have confidence in the "good behaviour" of all participants at all times, and this can come only when ease of use comes with strong assurances of trust, security and privacy are addressed from the outset.

For the consumer, "good behavior" involves assurances that their private information, including meter readings (that typically contain information about amounts, times, addresses of the house, etc), will be kept private and not used by anyone outside of the consumer/ESCO/provider collaboration. The consumer is also concerned about ensuring that any commands coming from the ESCO to their residence are not only uninterceptable, but also are issued by their ESCO and not from some other entity (e.g. their neighbor). From the point of the ESCO, "good behaviour" from the residence is that the information read from the meters is indeed authenticated, as well as correct (i.e. untampered) and only being able to be read by the ESCO. Similarly, the ESCO wishes that any commands sent to a residential meter arrive in the correct sequence, to the correct meter, and without interference or alteration.

Privacy, security and trust are such major concerns that at least one government has stopped the roll-out of smart meters [1]. Addressing trust and security in such system requires careful consideration, and the integration of a variety of technologies with social and regulatory measures.

In this paper, we consider two technological methods, one "a priori" transactional solution, and one "a posteriori" transactional solution. We concentrate on the former, and discuss in the further work section how the latter can be adapted to the system through the use of a dedicated accountability service to ensure trust, security and privacy are maintained. In particular, we present how we integrated a secure, hardware device into the system so as to start addressing these needs.

The paper first presents some background for the Residential Scale Energy Services project in Section II, then moves onto describing the security solution that was implemented for the CeBIT Australia 2010 demonstration in Section III.

We then conclude the paper with proposed future work in Section IV that includes an accountability service into the Residential Scale Energy Services system as well, and is used to assure the participants in the collaboration that any exceptions from the contract between the participants will be able to be detected and dealt with, once again with the intention of raising consumers' trust in the system.

II. Project background

The CSIRO Residential Scale Energy Services project is examining how to most effectively manage energy use (and minimize costs) for individual, residential households. This involves the introduction of a set of dedicated, and differentiated, Energy Service Companies between the consumer's residence and the energy supply company. It is expected that these ESCOs will offer a richer set of management and billing functionality than would be possible for the energy provider, and allow the consumer to delegate their residential energy management to these energy service companies with specific en-

COLLABORATECOM 2010, October 9-12, Chicago, USA

DOI 10.4108/icst.collaboratecom.2010.41

ergy consumption preferences and access policies. Currently, most ESCOs offer energy saving performance contracts, and some like Energy Response (http://www.energyresponse.com) and Eneroc (http://enernoc.com) offer demand side response management (i.e. allowing the management of large peak loads by controlled load shedding) as well. From the energy supply side, the ESCO is able to aggregate and optimally control loads according to the aggregated demand and costs. From the consumer's point of view, each ESCO is able to control energy consumption for individual residences through the use of smart-meter connected circuits and to minimize their costs according to a pre-agreed base set of energy policies and profiles offered by the ESCO.

However, we make the following two observations. First, electricity providers have a conflict of interest when it comes to providing energy cost saving services and managing demand. They derive profit from selling more electricity, not by reducing a consumers electricity bill. However, this profit is offset if there are large scale system failures due to under-provisioning of their base load infrastructure and so a risk-benefit analysis needs to be considered. We hypothesize that as loads become higher and energy becomes scarer and difficult to manage, the risks of not adopting the ESCO model outweigh the benefits. Second, we make hypothesis that ESCOs of the kind we are proposing in this paper must offer a better service than suppliers. It is outside the scope of this paper for the economic rationale for the ESCO's economic viability. Both of these hypotheses will be verified in subsequent work and associated trials to be carried out in Australia in the near future.

The ESCO and consumer policies are part of a contractual agreement that is entered into between the energy supplier, the ESCO and the consumer. It is important to note that these arrangements are not static, and any participant is free to vary the terms of the contract at any time. Should one or more of the parties find that these are no longer suitable, and a resolution cannot be found, that particular collaboration may be terminated.

As will be seen, the introduction of a specialized hardware device carrying policies, policy enforcement mechanisms, security tokens and associated cryptographic functionality, allows the consumer to terminate one collaboration, then start up a new one by simply swapping over to another ESCO's token. We refer to this device as an *ESKey*. The device is issued by the ESCO, and under its control at all times. Further, it is sufficiently cheap to be disposed of or revoked easily should the need arise (e.g. consumer not wishing to deal with ESCO, or the ESKey fails, or misbehaves and violates the agreed contract behaviour due to deliberate tampering).

A prototype has been developed jointly by CSIRO and Saturn South Pty Ltd [2] and demonstrated for CeBIT Australia 2010. This prototype utilizes Web Services and readily accessible home automation equipment integrated over an infrastructure architecture shown in Figure 1, and we believe it to be a particularly cost effective way of introduction smart grid technologies whose underlying principles and algorithms

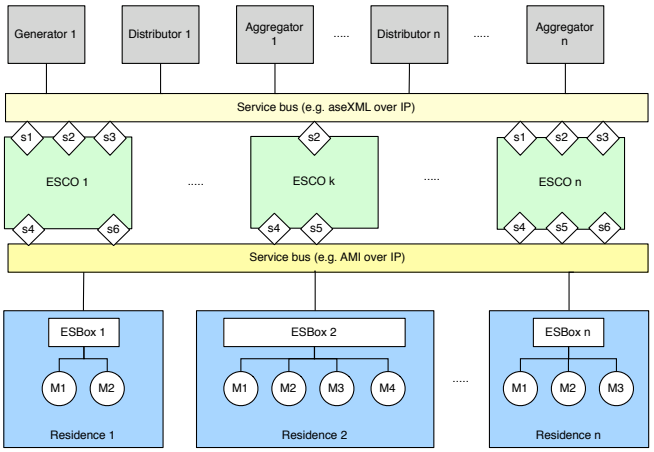

Fig. 1. Market Architecture

have been presented in [3], [4]. One of the unique and economically compelling features of the demonstrated system is that it may be readily deployed into customer residences without requiring any alterations or extensions to the internal wiring of the house. Figure 2 shows the architecture of the system for one residence.

Figures 1 and 2 adopt the following notations:

- *Blue Boxes*: Residential premises.
  - *Rectangles within, ESBox label*: Residential ESBox, as described below. Essentially a houshold energy service gateway.
  - *Circles within, M labels*: Residential mini-meters. These are direct replacements for the residential circuit breakers and provide power readings and control information to (and from) the ESCO via the ESBox according to the agreed upon Service Level Agreement between the residence and the ESCO.
  - *Squares within, D labels*: Devices attached per home circuit (e.g. toaster, electric kettle, microwave oven on circuit 1; washing machine, clothes dryer circuit 2; lighting in zone 1 on circuit 3; lighting in zone 2 on circuit 4; etc).
- *Green boxes*: Energy Service Company (ESCO).
  - *Square Diamond, s labels*: Service bus interfaces. Each ESCO will provide different services, both to the consumer (residential side) and the energy providers. For example, ESCO 1 offers services *s4* and *s6* to a consumer, whereas ESCO n offers these as well as another service *s5*.
- *Grey boxes*: Energy Providers and large scale distributor/aggregator companies.
- *Yellow rectangles*: Service bus, controlled and installed via an ESCO but run over a broadband Internet connection between the ESCO and the residence.These service buses run standardised message exchange protocols, e.g. Advanced Metering Infrastructure over IP, or aseXML over IP.

Our system assumes that each customer's residence is fitted

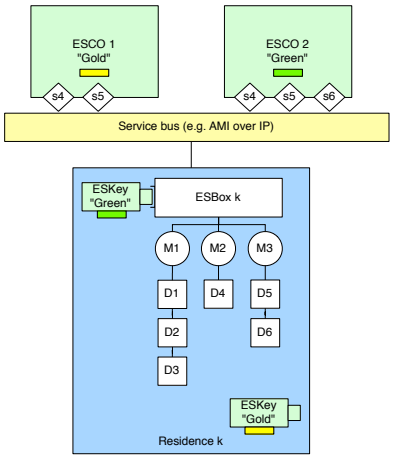

Fig. 2. Residential configuration – use of the ESKey

with a dedicated home control and management gateway called an *ESBox* (Energy Services Box), connected via the Internet back to an ESCO (typically via a cable or DSL broadband connection provided by an Internet Service Provider (ISP)). The ESBox uses Zigbee wireless networking to connect to a set of mini-meters (that include circuit breakers) that replace each of the household's circuit breakers. This allows each ESBox to both read and aggregates mini-meter readings and present them back to the ESCO, as well as allowing the ESCO to control (at the circuit level) the residence by switching on or off the circuit. This then allows the ESCO to accurately track the loads presented, and according to demand and pricing, either shed some aggregated load (by switching off, say, the air-conditioning units of its customers) or to offer cheaper pricing to the customer. Naturally, access and control of the customer's load is something that needs careful attention to agreed upon policy between the consumer, ESCO and energy provider. All transactions need to be suitably trusted, private and secure, and ideally, there should be a mechanism in place for detecting and dealing with exceptional and undesirable (from the point of view of the agreement in place between these entities) behaviors.

In the current prototype, the ESBox offers only routing, monitoring and switching capabilities and offers minimal support of trust, security or privacy. The ESBox is intended to be a common, standard Smart Grid platform, whose functionality is shared across all ESCOs. Which ESCO is selected and interfaces between the residence is determined by the ESCO issued hardware security device is (at any point in time) physically connected to the ESBox.

The functionality, and provision of controlling, monitoring, security and trust services lies within the use of the hardware security device we refer to as the *ESKey*. The current prototype demonstrated at CeBIT Australia 2010 contains the cryptographic keys, as well as implementing a set of basic cryptographic services (encryption, decryption, nonce generation, identity management) and ESCO's policy enforcement for the residential ESBox.

The envisaged scenario of use is that the consumer is issued with a ESKey from an ESCO. The keys and other sensitive information (e.g. ESCO servers IP address) are stored in secure storage on the ESKey, burnt in when it is created. Since the ESKey holds and locally enforces the energy policies of the issuing ESCO that the consumer agreed to upon purchasing the ESKey and service (through a contract), a consumer could easily change ESCOs by simply obtaining a new ESKey from the ESCO and replacing the currently connected one. We have designed and demonstrated that the ESKey hardware, as well as the authentication and security protocols are sufficiently robust that they do not result in a system failure or deadlock in the prototype.

III. The ESKey hardware security device

This hardware device, developed within CSIRO [5], is a USB sized "PC on a stick" and has a TPM cryptographic microcontroller adhering to the Trusted Computing Group's [6] version 1.2 of the standard. Although similar to the functionality offered by smart cards, it also differs from current smart cards in several respects. First, it is a flexible, open platform for testing and implementing smart card like functions, as well as offering a superset of smart card features, such as the ability to validate the complete operational integrity of the device (from applications to operating system to hardware). Our hardware device differs from the commonly found smart cards in that it does not require a specialized smart card reader, uses the ubiquitously available USB interface[1], and as mentioned, and offers rapid prototyping capability to develop new applications[2] suited to field trials and proof of concept demonstrations. Finally, and perhaps most importantly, it provides access to rich set of TPM-based cryptographic functions.

The implemented ESKey hardware security devices have the following features.

- *Secure Storage*: Customer information (such as customer ID, address, etc) and ESCO details (such as ESCO IP address) is used to associate specific energy customers with a particular ESCO. Because of its sensitivity, it is important that this information cannot be read, altered, deleted or otherwise tampered with. In order to meet this security requirement, our ESKey first forms a hash measurement of the ESKey hardware and all software that it is running. This hash value is called the *Platform Configuration* and is computed and held in the TPM chip. The ESKey then binds the customer data and the ESCO details to the Platform Configuration by using the TPM *seal* operation, and at this point, the information is stored in the secure, sealed storage of the TPM chip itself. The customer data and the ESCO details can only be accessed through an *unseal* operation, and this requires that the current calculated Platform Configuration is the same as

[1]Gemalto does supply smart cards with USB interfaces rather than the usual card reader interface, however, at the time of writing, it has seen little market penetration.

[2]The development environment for the device is *OpenEmbedded*, with the device's operating system based on Linux.

the previously calculated value, now stored in the TPM Platform Configuration Register. By demonstrating that there have been no changes to the ESKey's environment, it is safe to say that there have been no external attempts to modify contents of the ESKey, including stored data and the execution environment between the time at which the information was stored, to the time it is retrieved for the further use.

- *Hardware-based cryptographic engine*: One of the roles of ESBox is to aggregate and send energy data from the mini-meter readings to the ESCO when requested. The integrity of this sensitive energy data, as well as the ESCO issued query and other control commands, must be maintained from unauthorized reading and alteration, and so requires that it is encrypted before it is transferred over the Service Bus which is running over the public Internet (see Figure 1). Our ESKey uses symmetric key encryption, where the symmetric key is created by ESCO and it is registered into the ESKey. Using the TPM cryptographic engine in our ESKey, the energy data is encrypted before being transferred on the network using secure HTTPS channel. Similarly, ESCO commands and queries are decrypted by the ESKey before being sent to the ESBox for interpretation and implementation. As a final note, the ESKey platform is equipped with a set of functions which could potentially support more complex and state-of-art encryption techniques than the one used for this demonstration.
- *Identity Management*: In our demonstrator prototypes, a consumer is issued with a ESKey by an ESCO. Each ESKey has a unique identity credential, and this is represented in our prototype by using the TPM cryptographic controller's Endorsement Key (EK), consisting of a public/private key pair. The EK value is embedded at ESKey manufacture time, with the private part of EK never leaving the device (i.e. being revealed or transmitted) and is used as a base of secure transaction such as providing identity and integrity of ESKey's platform configuration to ESCO.
- *Policy Management*: A set of security policies can be easily defined and embedded into our ESKey platform. Such security policies can be used to determine the level of access to certain resources between the energy consumer and the ESCO. For the purposes of the demonstration, we developed two separate ESKeys, one "Gold", one "Green", each having different energy policy statement to access different types of services provided by two different ESCOs.

Figure 3 shows a message sequence chart of the security protocol implemented for our demonstration prototype.

The protocol shown consists of four distinct "phases", *P*, 0, 1 and 2 in message sequence chart showing the transactions between a single consumer's residence and the ESCO that they have selected and entered into a contractually binding service level agreement with them. All residential information flows via the ESBox.

The first phase referred to as *P* in the message sequence chart, represents the final step in the delivery of an ESCO's issued ESKey to a customer's residence by using a physical mail. Each ESCO controls the issuing and revocation of the ESKey, which contains, amongst other things, the ESCO's (distinct) authentication information. Prior to this step, the consumer and ESCO entered into a contract, binding the two together, and allowing them control the access to sensitive information (whether from the consumer – e.g. readings, or from the ESCO – e.g. commands).

It is important to note that from the point of view of a consumer, the message sequence chart given above is showing the case where they are using the Green ESCO. Our prototype allows (and was demonstrated) for contracts between the consumer and one or more ESCOs. If the consumer wishes to enter into another contract with another ESCO, the message sequence chart would simply included a separate *P* phase time-line. To use the energy services of a particular ESCO, the customer, upon receiving the ESKey, plugs it into the ESBox. The ESKey is powered up by the ESBox and starts to run the protocol proper from phases 0, 1 and 2. If the consumer has entered into two or more contracts, and has received separate ESKeys, then they can choose to change ESCOs by simply swapping over the ESKeys at any time during the execution. For the successful CeBIT demonstration, we had the consumer choose between a "Green" ESCO and a "Gold" ESCO, and were able to repeatably and reliably, swap their respectively issued Green and Gold ESKeys at any time during the execution of the protocol and without any system failure.

Following this initial phase, phase 0 executes after the ESKey has successfully powered up. Phase 0 represents the start of a new session, and within that session phases 1 and 2 can execute independently.

Phase 0 is the authentication phase, during which ESKey and ESCO mutually authenticate against each other by exchanging three messages that are routed by the ESBox. This sequence of messages is:

1) Upon booting up, the ESKey sends the first message
$$(enc(cid, nonce1, PK), IP_E, EPK)$$
where cid indicates the customer identity registered in the ESCO that issues the ESKey, $nonce1$ is a random number generated by the ESKey for the new session, IP_E, EPK are the IP address and public key of the corresponding ESCO, respectively. In this message, cid and $nonce1$ are encrypted using EPK, and so only the corresponding ESCO can obtain the encrypted cid and $nonce1$.
2) Upon reception of the first message, the ESBox configures the IP address of the ESCO for routing messages between the ESCO and ESKey. The public key EPK is used by the ESBox to build a secure channel such as HTTPS between the ESBox and ESKey, when the encrypted blob $enc(cid, nonce1)$ is routed to the ESCO.

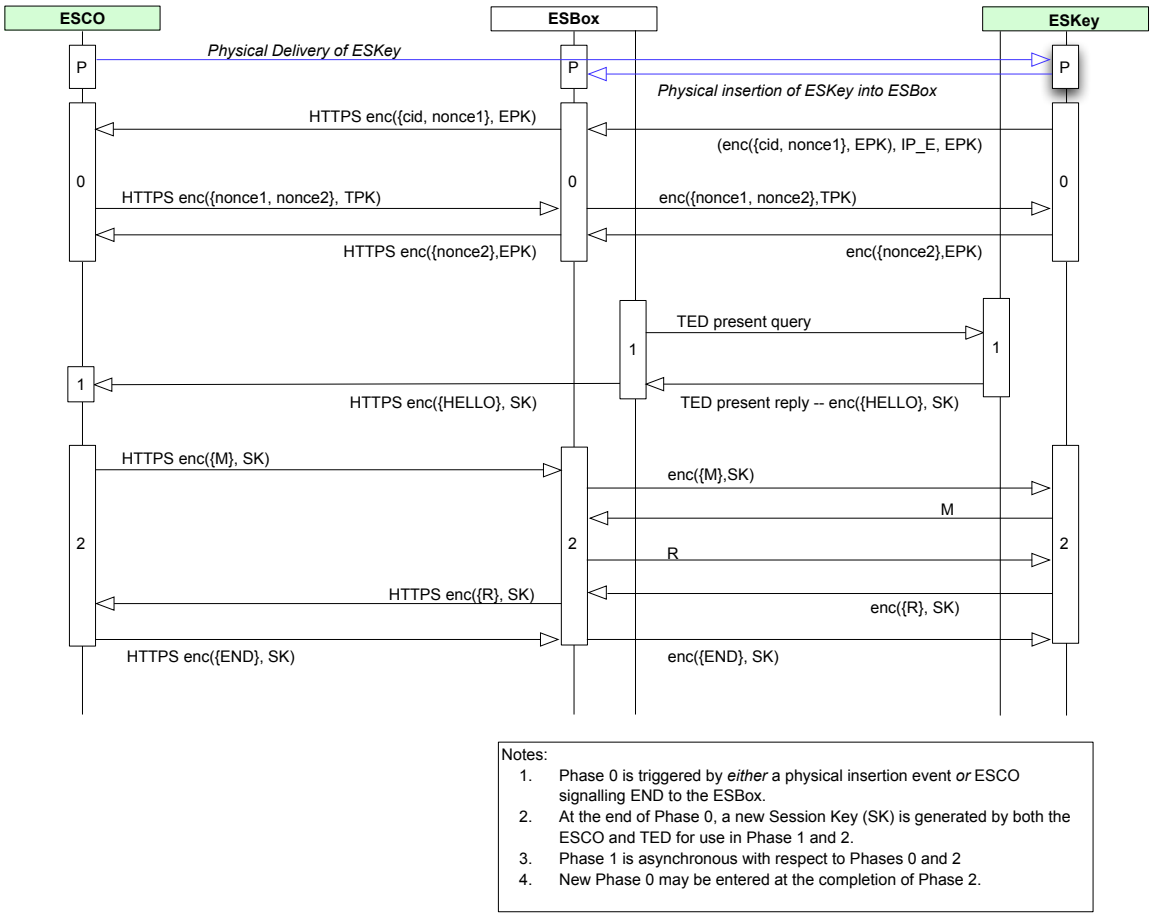

Fig. 3. Use of the ESKey in the smart-grid application: Green ESCO issued ESKey plugged into residential ESBox

3) The ESCO replies with the encrypted blob

$$enc(nonce1, nonce2, TPK)$$

where $nonce2$ is a new random number generated by the ESCO, and TPK is the public key of the customer indicated by cid.

4) As the ESKey has stored the private key of customer cid in its TPM chip, it can decrypt the second message. If the $nonce1$ in this message matches the $nonce1$ it sent in the first message, then the ESCO is authenticated. The ESKey then generates the third message and the ESCO finishes the authentication of the ESKey by checking $nonce2$ in the third message.
5) At the successful completion of phase 0, a new session key SK is generated based on $nonce1$ and $nonce2$.

The protocol now can either enter phase 1 or phase 2 (they run concurrently).

Phase 1 is a simple check to see whether the ESKey is alive. Since we assume that ESKey might be unplugged by the customer at any time, the ESBox polls the presence of the ESKey. The ESBox starts this phase by sending a query, and ESKey replies with the message $enc(HELLO, SK)$, where SK is the session key and $HELLO$ is a predefined constant. This message is then routed by ESBox to ESCO based on the IP address IP_E obtained in the phase 0, and indicating the presences of the ESKey at the customer's residence.

Phase 2 deals with protecting the commands and data exchanged between the ESCO and ESBox during the current session (indicated by the session key SK) by using the ESKey to perform symmetric encryptions. We chose symmetric encryption for this prototype because of its efficiency over asymmetric encryption (that is used during Phase 0). When receiving a message encrypted with SK from the ESCO, the ESBox passes on the message to ESKey to decrypt it, since the ESBox does not have the session key. On the other hand, when the ESBox needs to send energy data back to to the ESCO, it uses the ESKey to encrypt the data with the current session key. At any point during this Phase, an ESCO may choose to terminate the current session by sending the ESBox an $enc(END, SK)$ message. The ESBox passes this onto the ESKey, forcing it to restart the authentication protocol in Phase 0, and establish a new session.

IV. Further Work – an Accountability Service for Maintaining Trusted Interactions

The ESKey provides identity information, as well as offering cryptographic functions for assuring that all information shared between an ESCO and a customer's residence is

secured. However, it is only partial, technological solution to the issues of trust, security and privacy in a residential scale energy services system.

Additional mechanisms need to be put in place to ensure the good behaviour of all participants in the system, and according to their agreed upon behaviors when they are in a business collaboration between the energy provider, the ESCOs and the residential customers.

We have previously described the concept of an eContract within a dynamic collaboration system [7] that identifies and specifies agreed upon "good behaviors" in terms of access policies, shared resources, participants and so on. Its adaptation and application to the Residential Scale Energy Services system will be investigated in the future.

Based on the concept of having participants agree to a contract, it is clear that in order to enforce its terms, an accountability service needs to be put in place within the system. This accountability service is responsible for monitoring and providing irrefutable history of critical transactions within a dynamic collaboration that is bound under an agreed eContract.

For example, an ESCO is capable of switching on or off circuits of a residential household remotely via the ESBoxes if the household allows the ESCO to do so in the contract. The household may be benefited from reduced energy bills. However, there is no mechanism to justify the decisions of an ESCO made on behalf of its users currently. Ensuring security alone cannot address this problem. There is a need for supporting accountability in such a market.

We discussed in [8] on how to use an external party to maintain a state machine for a service consumer and a service provider to make their interactions accountable according to the contract. Under the context of Residential Scale Energy Service system, an external party can play a role of collecting market information and sampling the anonymized mini-meter readings, and then validating the control commands sent from an ESCO based on these data. In this case, an ESBox keeps a log recording its interactions with the corresponding ESCO. A minimal set of information can be extracted from the log and shipped to an accountability service. By aggregating such information from a set of ESBoxes subscribed to a same ESCO, the accountability service will be able to detect anomalies when problems occur. With such a special type of service, the market can be enhanced with the ability to hold a party responsible for its actions.

V. Conclusions

We have demonstrated the use of a specialized hardware device, the ESKey, and associated security features in establishing a trusted, secure Residential Scale Energy Services prototype system that we regard as representing a dynamic collaboration between energy providers, ESCOs and consumers.

The ESKey carries cryptographic keys, identity certificates, policies and policy enforcement mechanisms. The ESKey offers similar functionality to that of smart cards, but also differentiates itself from a smart card by offering ease of development, a rich set of TPM-based trust and security operations, and a USB interface (rather than the usual smart card interface). These set of features allow rapid prototyping and evaluation of concepts such as those behind the Residential Scale Energy Services demonstration presented at CeBIT Australia 2010. Further refinement and possible implementation into traditional smart card form factor may be possible from this prototype stage.

We believe that the addition and integration of accountability services, would add substantial value to the trustworthiness of a system that includes a hardware security device such as an ESKey or smart card.

References

[1] C. Cuijpers, "No to mandatory smart metering: does not equal privacy." [Online]. Available: http://vortex.uvt.nl/TILTblog/?p=54

[2] "Saturn South Pty Ltd web site http://www.saturnsouth.com." [Online]. Available: http://www.saturnsouth.com

[3] C. Wang, M. de Groot, and P. Marendy, "A service-oriented system for optimizing residential energy use," in *Web Services, 2009. ICWS 2009. IEEE International Conference on*, 6-10 2009, pp. 735 –742.

[4] C. Wang and M. de Groot, "Managing end-user preferences in the smart grid," in *e-Energy '10: Proceedings of the 1st International Conference on Energy-Efficient Computing and Networking*. New York, NY, USA: ACM, 2010, pp. 105–114.

[5] S. Nepal, J. Zic, H. Hwang, and D. Moreland, "The Trust Extension Device: Providing mobility and portability of trust in cooperative information systems"," in *International Conference on Cooperative Information Systems*, 2007, pp. 253–271.

[6] "Trusted Computing Group web site http://www.trustedcomputinggroup.org." [Online]. Available: http:www.trustedcomputinggroup.org

[7] J. Chan, S. Nepal, D. Moreland, H. Hwang, S. Chen, and J. Zic, "User-controlled collaborations in the context of trust extended environments," in *WETICE*. IEEE Computer Society, 2007, pp. 389–394.

[8] C. Wang, S. Nepal, S. Chen, and J. Zic, "Cooperative data management services based on accountable contract," in *Cooperative Information Systems (CoopIS) 2008 International Conference*, 2008, pp. 301–318.