A Security Policy

for a

Distributed Utility Metering System

Christiaan Rudolph Burger

Submitted in partial fulfilment of the requirements for the degree
Master of Engineering (Electronic)

in the
Faculty of Engineering, Built Environment and Information Technology

University of Pretoria

March 2003

Summary

This dissertation describes a security policy for a distributed utility metering system. The system uses untrusted networks, such as the Internet, to communicate between service providers (water, gas, electricity etc.) and the gateway servers at customer premises. Within a building, the system uses a low-bandwidth mains-borne network, or Field-Area Network (FAN), such as Fieldbus, to communicate between the gateway server and each of the utility meters. The FAN is regarded as untrusted, and communications to and from each utility meter must be protected from all other meters and any possible outsiders on the network. It must also be assumed that the gateway server is physically vulnerable to attack, and that its loss must not jeopardise the security of the system.

Each service provider must be able to access each utility meter individually. Service providers can send commands to individual utility meters, and obtain individual meter readings applicable to their service. Service providers must not be able to interfere with one another's service. However, the gateway must be able to interpret communications initiated by individual meters, to ensure that the alarm can be raised to service providers if a meter reading appears to have been tampered with.

On high-bandwidth networks, well known symmetric and public-key cryptography techniques can easily provide the required features. However, with a low-bandwidth network such as FANs, the protocol must be carefully optimised to minimise the amount of data transmitted. This dissertation describes a new architecture, in which well-known cryptography principles are applied in the FAN field in a way that has not been described in the literature.

Keywords

cryptography applications, Fieldbus, utility meter, field area network, distributed metering

Opsomming

Hierdie verhandeling beskryf ’n sekerheidsbeleid vir ’n verspreide dienste-meterstelsel. Die stelsel gebruik onvertroude netwerke, soos die Internet, om tussen diensverskaffers (water, gas, elektrisiteit ens.) en die toegangsbedieners by kliëntpersele te kommunikeer. Binne die gebou gebruik die stelsel ’n lae-bandwydte netwerk wat op die kragtoevoerdrade loop, soos Fieldbus, om tussen die toegangsbedieners en elk van die meters te kommunikeer. Die kraglyn-netwerk word as onvertroud beskou, en alle boodskappe na en van die meters moet teen ander meters en vreemdelinge op die netwerk beskerm word. Die toegangsbediener self word ook as vatbaar vir fisiese aanvalle beskou, en selfs die diefstal van die rekenaar mag nie die sekerheid van die stelsel in gevaar stel nie.

Elke diensverskaffer moet elke meter afsonderlik kan bykom. Diensverskaffers kan bevele aan indiwiduele meters stuur en afsonderlike meterlesings vir hulle diens van elke meter kry. Diensverskaffers moet nie in staat wees om met mekaar se dienste in te meng nie. Die toegangsbediener moet in staat wees om kommunikasie van indiwiduele meters te verstaan, om seker te maak dat oënskynlike onreëlmatighede met meterlesings dadelik aangemeld kan word.

Op hoë-bandwydte netwerke kan bekende simmetriese en publieke-sleutelstelsels maklik die vereiste funksies lewer. Op lae-bandwydtenetwerke soos Fieldbus moet die protokol egter versigtig ontwerp word om die hoeveelheid inligting wat oorgestuur moet word te minimeer. Hierdie verhandeling beskryf ’n nuwe argitektuur waarin bekende kriptografie-beginsels in veldnetwerke toegepas word op ’n wyse wat nog in die literatuur beskryf is nie.

Sleutelwoorde

Kriptografie-toepassings, Fieldbus, dienstemeter, veldnetwerk, verspreide metering

List of Definitions and Non-Standard Abbreviations

- **MU (Meter Unit):** The unit installed inside a user's home or office, which directly controls the services being provided (water, electricity etc.). The MU must contain a switch of valve for each service, and its enclosure must be sufficiently robust to make it difficult to interfere with the operation of these switches or valves.

- **Field-Area Network (FAN):** A mains-borne network, using higher-frequency components than the 50 Hz power that is the prime content of the wires. Typical modulation frequencies can be from a few tens of kHz to perhaps 20 MHz, and bit rates of over 10 Mbps are advertised. However, practical throughput can be very low in practical networks, perhaps even as low as 10 bps or less. Fieldbus is one of the available solutions, and many of the references use this term interchangeably with the term "FAN".

- **Gateway:** The controller responsible for a group of meter units, typically within the confines of the premises. A Gateway might control all the MUs in a single multi-storey building, or a group of buildings in close proximity, and may have up to several hundred meter units. The Gateway communicates with the meter units through a FAN or other low-bandwidth LAN, and with the utility providers through the Internet.

- **Security Application Module (SAM):** The SAM is used as a cryptography coprocessor, providing encryption and decryption services and secure key storage. The SAM could consist of a tamper-proof module containing a secure microprocessor and some ancillary equipment, or a simple smart card module.

- **TTP (Trusted Third Party):** A contractor, responsible for programming and/or managing a scheme. This contractor must be trusted with the keys by all scheme participants, except in the case of a public key system. The scheme can be

operated without the existence of a TTP, if such a party cannot be found, but such schemes introduce unnecessary complication.

- **TC (Trust Centre):** The physical installation through which the TTP administers the Scheme.

- **SP (Service Provider):** A company providing a specific utility service to many MUs. Several SPs might require access to a specific MU, as different SPs might provide different services (e.g. water, gas, electricity) to that MU.

- **Scheme:** A collection of SPs, installers and users, according to which some service providers are given access to some or all MUs. A scheme would typically use a single TTP to administer its user base.

- **SO (Scheme Operator):** The SO might be the TTP, or might employ a separate entity to fulfil the role of TTP. The SO might even be one of the SPs, in which case a third-party TTP is imperative if other SPs are to be allowed to participate in the scheme. The SO takes initiative for the scheme, solicits customer participation, finds interested SPs and coordinates the supply and installation of MUs and Gateways. There is only a single SO per scheme.

Table of Contents

1. Research Overview

1.1 Introduction

This dissertation sets out to provide a security policy for a distributed utility metering system.

The system to be investigated consists of a number of meter units (MUs) on customer premises. The system is administered by separate utility service providers (SPs) through gateways installed in customer premises.

A more comprehensive description of the system is provided in Chapter 3.

The security policy does not make use of any novel cryptography techniques. Instead, it sets out to apply some well-known principles in an arena where these principles have thus far apparently not been applied. A thorough search of the literature has not unearthed any prior applications of similar architectures in bandwidth-limited networks.

Perhaps the most substantial existing reference in this field is the work of Sauter and Schwaiger [F10]. They address the security issues surrounding Internet-FAN interconnections by proposing a key-derivation scheme to determine keys for inter-unit interconnections. The basic strategy is sound, but perhaps the single biggest shortcoming is the lack of a suitable broadcast mechanism within the FAN. Their work also emphasises smart cards as security coprocessors, something that may prove too restrictive for a typical gateway application. They also do not provide a solution for untrusted gateways, something that one would almost inevitably have to contend with in practical systems.

This dissertation addresses some of the shortcomings of the proposed solution, specifically in terms of proposing a shared-key architecture in the FAN to facilitate broadcasting and the use of gateway security coprocessors not based on smart card platforms.

The dissertation will provide a discussion of all the relevant issues, along with a proposed architecture that meets all the stated requirements of security, cost and response time.

The work could be seen as an attempt to introduce the FAN engineering fraternity to sound security principles. All telecommunications technologies (telephone, commercial networks, the Internet and even mobile phones) have evolved in a similar way: The early emphasis was on affordable and ubiquitous interconnectivity to the exclusion of all other considerations. As ubiquity and affordability became achievable, it became evident that security was a real issue, and had to be addressed before commercial exploitation of the relevant technology would be possible.

This transition is not yet complete in FAN technology. The emphasis is still on interconnectivity and affordability. However, in due course the security emphasis will also come to FANs. Perhaps this work can provide some early exposure in that regard.

1.2 Scope

The distributed utility metering system is described in some detail in the references [F1, F2, F3]. This document is focussed only on a suitable security policy for the system, and does not address protocol or architecture design issues beyond those required to implement reasonable security for the system.

Because the full system is being implemented by a number of different students, and many members of the team are dependent on the existence of a security policy to be able to complete their own designs, this security policy must be published before full details on other system building blocks have been fully quantified. For this reason, a few issues must be left open for further investigation. As the platforms are chosen and their performance can be determined more accurately, some minor adjustments may have to be made to specific aspects of the policy.

1.3 Problem Statement

Cryptography applications have become widespread over the past decade or so, as interconnectivity between computers has spread. While interconnectivity brings great advantages to computers, it introduces ample scope for abuse that can only be addressed with a decent security policy, supported by suitable cryptography equipment. Such equipment is easily obtained, and the principles underlying suitable policies are well understood and widely published.

However, as networks have evolved and computers have become cheap and powerful, two things have happened:

- The network owner's ability to implement suitable safeguards has improved, as non-volatile key storage has become feasible and as computer power allows ever-increasing sophistication in algorithms with decreasing response times.

- The attacker has become more powerful, necessitating ever more powerful safeguards.

An analysis of the various factors influencing the balance of power in cryptography systems is beyond the scope of this text. Suffice to say, though, that cryptography is one field in which the good guy is favoured. Adequate protection is becoming easier and easier, even in the face of malevolent attackers. Briefly: A comparatively small increase in key size does not appreciably increase the computation or storage burden, but increases the attacker's search time significantly. As a brief example, increasing the key length from 100 to 110 bits would increase the storage and processing requirements by around 10%, while an exhaustive search would take just over 1000 times longer. In public key systems, where the attacks no longer rely on exhaustive searches but rather on factorisation, similar trends apply.

One can therefore say with confidence that equipment and knowledge abound, and that an adequate protective system can be implemented with very little effort in modern networks. In fact, it is feasible to implement a solution that will resist likely

attacks for years to come, well beyond the envisaged lifetime of the scheme that it protects.

However, the present network has some unique characteristics that make typical existing solutions unsuitable. The most important of these characteristics is the fact that a FAN (field-area network), if not installed in a controlled environment such as a laboratory or factory, can suffer from some very severe shortcomings in availability and bandwidth.

Furthermore, the kind of embedded processors suitable for meter units (MUs), in terms of cost, robustness and complexity, do not lend themselves to complex cryptography systems, in terms of processing power and non-volatile storage.

1.4 Research Approach

The approach taken in this dissertation emphasises implementation issues. The principles discussed are not new. However, their application requires considerable thought to circumvent some of the difficulties inherent in the system architecture.

It is hoped that this dissertation will serve to introduce some oft-used network security techniques to the world of FAN technologies. The early development of FANs has followed a path analogous to that of the Internet, where basic connectivity is being emphasised to the exclusion of security. It is expected that security will eventually become an integral part of FAN implementations, just like it now is in the Internet world.

Some of the issues to be addressed include:

- How do symmetric key systems and various public key systems compare in terms of required resources in processing, non-volatile storage and bandwidth?

- How can the requirement for *ad hoc* notifications when discrepancies are detected be reconciled with the financial constraints of network connectivity costs?

- How can the Gateway be given access to communications between MUs and service providers (SPs) without jeopardising the stated requirement for resistance against physical attack of the Gateway?

- What building blocks can be used to ensure that the trust requirements are satisfied without exceeding the cost constraints in the system?

- How can replacement units be made available without jeopardising the security of the system?

- Is a trusted third party (TTP) required for the operation of the scheme?

As these and other questions are considered and answered, a judicious and economical implementation will result.

1.5 Document Overview

This dissertation consists of the following sections:

Chapter 1 is a Research Overview, describing the overall problem and the intended approach to provide a solution, as well as this document overview.

Chapter 2 is a literature study, investigating existing solutions in this field.

Chapter 3 provides a system overview, including a description of the trust assumptions which have to be satisfied by the proposed security architecture.

Chapter 4 discusses the cryptography architecture in some detail. After an overview of the suitability of public-key and symmetric systems, some hardware

implementation issues are discussed. Brief mention is also made of possible choices for the symmetric algorithm, other than the assumed choice (3DES) elsewhere in the document.

Chapter 5 provides a detailed specification for each component of the system. A protocol for communicating between the different subsystems is also proposed.

Chapter 6 provides a Summary and Conclusions. After reviewing the proposed solution, it suggests further work that might prove fruitful in the same field.

The dissertation concludes with a list of references, divided into the three fields of FANs, cryptography and hardware platforms, and an appendix containing abbreviated specifications for a suitable microprocessor on which the security module can be based.

2. Literature Study

2.1 General Overview

Much of the available literature on FANs revolves around FANs in the industrial control [F4, F5] and domestic automation [F6, F7] domains. Only passing mention is made of utility metering [F7].

Several references make the point that there has been little concern for FAN security in the past, and that the concern has only now raised its head because FANs are increasingly being connected to the Internet. This argument has some merit, as security has certainly been accepted as an integral part of Internet systems, and questions are likely to be asked about any network connected to the Internet. On the other hand, in less-developed technologies the emphasis is more likely to still be on obtaining simple connectivity, without worrying about security. This situation is reminiscent of the situation on the Internet just a few years ago.

However, the literature on the subject is extremely limited. A search of over two million publications, using IEEE Xplore, Elsevier Science Direct and general Web searches produced only a handful of documents covering FAN security. Virtually all of these appear to emanate only from the Institute of Computer Technology at the Technical University of Vienna, from where the Iguana protocol originated, and most are taken from conference proceedings.

2.2 Summary of Available Publications

Palensky and Sauter [F7] provide a good discussion of security issues in FAN-Internet interconnections. They specifically make the point that a node (such as an MU) cannot be regarded as secure, even if it contains a smart card module, if the plaintext version of the counters can exist within the MU at any time. Their prime concern appears to be that the MU RAM must be accessible for remote administration purposes, and they postulate that a separate microprocessor without remote

administrative capabilities must be included in the MU to facilitate secure transmission.

The authors of several publications [F8, F9] spend considerable effort on comparing different protocols and their suitability for FAN/Internet interconnections. Lobaschov, Pratl and Sauter [F8] compare SNMP, LDAP, SQL, HTTP and IGUANA in terms of security, their ability to handle changes to the network structure, their initiation of an update to the node list, the FAN address mapping, data interpretation, the amount of overhead per request and their ability to handle asynchronous notifications in the direction opposite to the normal direction of communications. IGUANA compares well for these purposes in all but the last respect, and the problem has been addressed by running a notification protocol concurrently on the same FAN. IGUANA is therefore identified as a protocol particularly suitable to the kind of communications envisaged in the FAN.

Schwaiger and Sauter [F4, F5, F10] propose security architectures based on both smart cards and usernames/passwords. The latter approach is not suitable for a utility metering system, given the varying levels of literacy in the user base and the likelihood that user collusion could influence the SPs. An interesting aside is that a comparison of various block sizes for block ciphers is presented, indicating that the total processing time remains virtually constant for all block sizes from 128 to 2048 bytes. This paper also proposes key generation schemes, although the first two papers appear to suggest that the generating master keys be retained in each MU, and the keys then be generated in the coprocessor before a session. The final version [F10], though, indicates clearly that the generating master keys are retained and safeguarded by the TTP, except for diversified keys for access to different MUs on the FAN. In the latter case, keys are derived from the respective MUs' serial numbers. This paper also refers to the use of AES as a viable alternative to DES and its derivatives.

Several articles [F4, F7, F10] use the term "proxy" to describe the Gateway, and present approaches not too dissimilar from Internet firewalling architectures for

solving the Internet/FAN interfacing dilemma. None of these articles mention the influence of restricted bandwidth on security policies in the FAN.

Gordeev [F11] provides a useful introductory discussion to the issues surrounding information security in an Internet/FAN interconnection. However, his discussion revolves only around smart cards. He specifically mentions that the Gateway/smart card interface is vulnerable, but suggests that this problem can be solved by integrating the card into the Gateway server. He stops short of suggesting that an untrusted Gateway server can be used.

The reliance on a trusted Gateway server is a common thread through all the available literature. As such, it can be argued that none of the articles provide a satisfactory solution for the present system, as its trust model assumes an untrusted gateway server that could be subject to physical attack or the remote installation of malicious applications.

Finally, Borst, Preneel and Rijmen [H1] provide a useful overview of smart card architectures and performance issues. They provide a section on attacks on smart cards that would be useful reference for a system designer intending to use these devices as building blocks in a system design.

3. A System Overview

3.1 General Overview

The dissertation describes a security policy for a distributed utility metering system [F1, F3], providing a number of different service providers (SPs) the ability to individually manage a large number of subscribers.

The subscribers must each have a Meter Unit, controlled in groups by gateway servers. Untrusted networks, such as the Internet, can be used to communicate with these gateway servers. The networks inside the buildings are also assumed to be untrusted, so that a rogue subscriber cannot interfere with other meter units in the same premises or using the same local network.

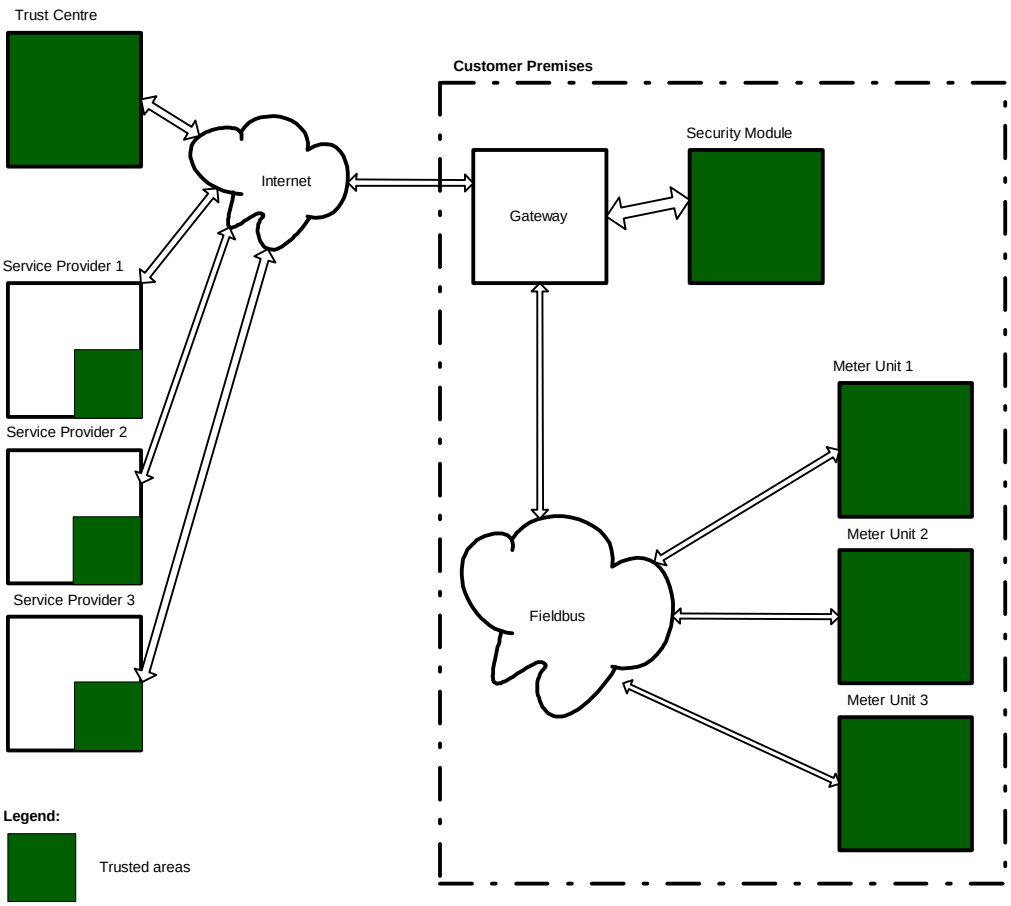

Figure 3-1: Block diagram, showing trusted areas.

The diagram shows the basic components of the system. In principle, the number of service providers can be extended to any reasonable number, as can the number of customer sites. Each customer site must include a gateway and a number of MUs. However, depending on available network infrastructure, a customer site as intended here could be distributed across any number of buildings or sites, as long as all MUs are in contact with the Gateway through a suitable bi-directional network. The architecture has been designed to take account of the fact that a typical FAN cannot provide wide bandwidth. Although vendors claim speeds of "up to 14 Mbps" in domestic installations [F12, F13], speeds in multi-phase systems and buildings with multiple households are much slower. The same manufacturers claiming 14 Mbps in house wiring cannot guarantee 50 bps in office buildings.

The most severe problems for FANs are associated with capacitive loading from PC power supplies, inter-phase isolation by transformers and noise sources due to poor connections and noisy loads. In large buildings, the wavelength of the modulated signals (4,3 to 20,9 MHz, with wavelengths of under 15 m) could also play a role, as typical building wiring could contain standing waves with a series of nulls and nodes at various frequencies in the modulation band. In practice, many pairs of powerline modems in a large building cannot mutually communicate due to these effects. Some commercial solutions have even been built around additional "repeater" nodes that can be installed in buildings to address some of the most severe point-to-point problems.

Thus, the protocol has been designed to minimise the amount of data transmitted between the Gateway and the MUs wherever possible.

While the specifications provided [F1, F3] do not call for the presence of a Trust Centre, the reasons for its introduction will become clearer as the relevant cryptography issues are discussed.

3.2 Trust Assumptions

The following trust assumptions are regarded as the basis for the system design:

- **The MU (inside the house or apartment) is trusted.** This assumption is the only reasonable one, as a user who can gain access to the MU can bypass its operation in basic ways (such as shorting the input and output connectors together). The enclosure of the MU must be sufficiently robust to ensure that tampering is difficult, and can be detected if it does happen.

- **Communications between different MUs and the Gateway must not be visible to other MUs.** Other users within the same building must not be able to interfere with or eavesdrop on communications between an MU and its Gateway.

- **The Gateway must be able to interpret messages from and to any of the MUs.** The Gateway must be able to check whether parameters being returned by an MU are reasonable, and must be able to raise the alarm if they are not.

- **The Gateway is not physically secure, and could be accessed by attackers.** A Gateway might reside in a basement, where it cannot be counted on to survive untouched forever. It must be assumed that attackers can install malicious code inside the Gateway server, and can monitor any and all interactions with its SAM.

- **All networks used for communications are untrusted.** Both the FAN used inside the building and the network used with the SPs are open to attack. The Internet might even be used as a carrier between the Gateway and the SPs, or even the TC.

3.3 Operational Requirements

The following operational requirements have been stated:

- Each MU must be able to remain off-line for up to 30 days, while retaining all its records and meter integrity.

- Each SP must be able to interrogate each MU individually. No SP must be able to interfere with the operation of another service in the same MU.

- Each Gateway must be able to check each packet from each MU for integrity (e.g. reasonable meter readings) before forwarding the packet to the relevant SP. If apparent discrepancies are noted, the Gateway must immediately establish a connection to the SP and report the problem. This ability is intended to safeguard systems that are not permanently online, so that fraud cannot be perpetrated during the offline periods.

4. Cryptography Architecture

4.1 Public-key vs. Symmetric Systems

In principle, implementing a system where several service providers can communicate with a multitude of individual meter units is not hard. Either a symmetric-key or a public-key system can provide any level of security required, with a suitable choice of key lengths and algorithms. Any good introductory cryptography text provides enough information to make the relevant choices [C1].

Because of the number of MUs involved, public key systems offer some very attractive benefits in terms of key distribution. However, the modest computation capabilities of microprocessor platforms suitable for the MU are a severe restriction in typical public-key systems.

Elliptic Curve Cryptography (ECC) offers some hope of reasonable key lengths and a reasonable computation power requirement when using small embedded microprocessors [C2]. For commercial applications such as this, the table below provides a comparison of processing power and key length requirements for a symmetric system, an RSA-based public key system and an ECC-based public key system.

	Symmetric	**RSA**	**ECC**
Key length required (bits) [C3, C4]	112	2048	224
Processing power required (relative) [C2]	1	»600	»9000

The processing power figures are rough indications only, and are based on a set of benchmarks using a similar platform to implement 3DES, RSA-2048 and ECC-155. The 3DES benchmark speed for a block of 128 bits (16 bytes) is 1,46 µs. The RSA-2048 encryption time is 0,9 ms, with a decryption time of 64 ms. The ECC-155 encryption and decryption time are 24 and 14 ms respectively. Using these figures as a comparative basis produces the best-case figures mentioned in the table. In real life, the ECC figures would be much worse because ECC-224 would have to be used, and

RSA figures would be much worse because decryption is also involved. Decryption involves a factor of around 39 000.

The key lengths are based on the minimum required to obtain reasonable integrity in a symmetric system [C5], with approximate equivalents in the public-key systems to obtain the same level of integrity [C3]. Although a key length of 100 bits is adequate for the purpose, the most commonly used symmetric cipher (DES and its derivative Triple DES) use key lengths in multiples of 56 bits.

Each of these solutions offers some advantages and suffers from some drawbacks in this context:

	Advantages	**Drawbacks**
Symmetric (e.g. 3DES)	High assurance of integrity Low processing burden Low RAM requirements	Difficult key distribution
RSA	Public domain High assurance of integrity	High storage requirements
ECC	Modest key length Easy key distribution	Low assurance of integrity

The items labelled “assurance of integrity” revolve around the fact that cryptography solutions are not provably secure, and users rely to some extent on the algorithms’ track records. With many symmetric systems and with RSA public-key systems, several decades of attacks from a myriad different sources have not resulted in known successes. There is therefore very high certainty that these systems do not suffer from design flaws that could lead to them being jeopardised.

While one occasionally sees claims that industry standard algorithms such as DES have been "cracked", these claims are generally unfounded. Most often, attackers simply build powerful machines to do an exhaustive search quickly [C6]. They have not found a short cut, and have simply proved what the designers said in the first place.

Similar claims have been made about RSA, when the 512-bit key that was originally published as an attack target was factored, yielding the private key. Again, while this team did make some advances in factoring techniques, their computational effort was very much in the same order of magnitude as that originally predicted by the designers, and considerable processing power was brought to bear to find this relatively modest key [C4].

Clearly, in principle, there is a lot of flexibility in the choice of cryptography solutions, and any of a number of systems in common use can be implemented.

However, the system under discussion has two problems that make conventional solutions unuseable:

- **The low bandwidth** available in the FAN [F12, F13] precludes the use of long certificates and/or large encrypted blocks. Large packets would prove cumbersome in a network with a bandwidth of perhaps a few tens of bits per second.

- **The need for the Gateway to inspect the contents of packets** destined for SPs means that end-to-end cryptography solutions cannot be used. There must be communications between the MU and the Gateway using one key, and communications between the Gateway and the SP using another.

The first problem imposes a limit on packet size inside the FAN of perhaps a few dozen bits. In general, small block sizes (significantly below the key length) could jeopardise the integrity of the system, as not many such blocks would be required to

launch an attack on the system. For symmetric key systems, this constraint would dictate a minimum packet size of at least 100 bits. For hybrid systems using public keys for session key distribution, the minimum packet size would be similar, but for pure public-key systems the block size would have to be significantly greater.

The second problem precludes the obvious solution, where the communications between an MU and an SP are encrypted with a symmetric or public key. Such solutions, in general, are known as End-to-End security systems, as the nature of the intervening channel is assumed to be untrusted, and no-one sharing the network has any form of access to the communications. Because the Gateway must be able to inspect each packet, it must either have the SP's secret key (a requirement which is clearly not reasonable) or the packet must be re-encrypted for the Gateway. In secret-key systems, the entire packet would have to be re-encrypted using the Gateway's secret key. In public-key systems, the MU would have to re-encrypt the session key under the Gateway's public key, adding a number of bits roughly equal to the key length (in the order of 100 bits) to the packet. In both cases, given the bandwidth restrictions, such duplication is clearly unacceptable.

A solution therefore has to be found that allows the Gateway to share a secret key with the MU and with each SP, without jeopardising its own trust model. If we remember that the Gateway is assumed to be untrusted and could be physically accessed by an attacker, this requirement is not simple.

4.2 Hardware Implementation Issues

The proposed solution uses a series of coprocessors. These coprocessors can encrypt and decrypt communications, and will only provide the plaintext information to the host processor if the packet is intended for that particular processor. In this context, the coprocessors are in the form of SAMs (Security Application Modules), as their prime purpose is to handle cryptographic functions.

In this system, the Gateway and all MUs on the customer site (i.e. on the same FAN) share a secret key. This secret key is stored only in the SAM, and cannot be read

from that SAM. The security level of each SAM must clearly be adequate to ensure that the key cannot be read from the SAM, even by an MU owner who has unlimited access to his own MU's SAM.

Modern security processors provide such immunity. Given the amount of data to be stored and the cost constraints, MU SAMs will probably be in the form of smart card processors. This approach is easy to implement, as a SIM-type card socket could easily be incorporated in each MU. The smart cards would then also be used to distribute keys, and the system administrator would simply provide a set of pre-programmed smart card SAMs for each customer site.

The Gateway's storage requirements, especially if a large number of MUs will have to work with a single Gateway, are beyond the means of current smart cards. The Gateway has to store a separate key for every SP, as well as the secret key it shares with its MUs, and several parameters for each MU. A typical minimum set of parameters might include:

- Serial number
- Last received meter reading for each SP (electricity, gas etc.).
- Last transmission sequence number received (see discussion on packet composition).

The Gateway might also have to store separate administrative keys, used by the scheme operator for disaster recovery and general administration. These secret keys are shared with the Trust Centre, or with individual SPs if there is no scheme operator.

Because current smart card processors are not up to the task, the Gateway SAM will have to us a more substantial processor. A more comprehensive discussion of

possible approaches to implementing the Gateway SAM is supplied in the following section.

4.3 Alternative Algorithm Choices

The preceding discussion has thus far used 3DES as an example of a suitable symmetric algorithm. 3DES (Triple DES, after the Data Encryption Standard favoured by the US government), in its commonly-used EDE (encryption-decryption-encryption) mode, provides full backwards compatibility with DES if two identical 56-bit keys are used, and provides 112-bit integrity with two different 56-bit keys. A 112-bit key length is regarded as adequate for commercial purposes [C5]. Because DES is a well-known and well-trusted algorithm that can be implemented fairly efficiently in software and hardware, it has become pretty much ubiquitous.

However, there are alternative choices that might provide considerable advantages in speed and storage requirements.

Keating [C7] implemented all the AES candidates and DES in a Motorola 6805 CPU, and provided benchmarks of their performance at a pre-selection conference. AES (Advanced Encryption Standard) is a US government initiative to replace DES with a more flexible and possibly more efficient alternative. The newcomer had to allow different key lengths, and lend itself to efficient implementation in hardware and software [H7].

Rijndael, the algorithm that was eventually chosen as AES, was the hands-down winner on this particular platform. Interestingly, DES and even 3DES outperformed all but two of the candidate algorithms. Three of the candidates had a speed deficit of around ten times relative to AES.

Although no exhaustive review has been conducted on the implementations or the comparisons, the point is that there can be substantial differences in implementation speeds between different algorithms on a particular platform, and a different choice of algorithm might yield vastly superior results.

3DES does have the advantage that dedicated 3DES engines are now included in many smart cards. These engines provide much better performance than would be obtainable with software implementations on any standard microprocessor platform, including smart cards.

Because of AES's status as a US Government encryption standard, it is likely to start appearing in crypto-coprocessors in the near future. At this point, it may surpass 3DES in usefulness, even when using off-the-shelf building blocks.

5. Specification of System Components

This section describes each component of the total system, and a protocol for communicating between them.

It is assumed that a symmetric key system is used between the SPs and the Gateway, and that a TC is used. If no TC is used, each SP would have to include TC functionality.

5.1 The Service Provider

The SP has perhaps the simplest operational requirements, as it is the only entity that is assumed to be fully trusted. Obviously, it is in every SP's interest to provide the necessary arrangement to ensure that its servers remain trusted, failing which it will suffer commercial loss. These arrangements would include physical safeguards (access control, building security, alarms etc.) and administrative arrangements (rules, staff screening, key update intervals etc.).

The preceding discussion assumes that the SP servers need no physical security. However, in practice this assumption is not really used, because of concerns over insider collusion. Corporate governance requires that a company not unduly rely on the trustworthiness of individuals within the organisation, and for good reason.

A practical SP server would therefore include some safeguards that might end up being not too dissimilar from the trust model in a Gateway server.

The secret key might be protected in a SAM, similar to the Gateway device. Any key management activities might be subject to controls that include multiple responsible persons that all have to be present to initiate the transaction, as well as multiple keys or smart cards held by different individuals.

If the Internet is used as a communications medium with the Gateway, the SP server would also have to include suitable precautions against attack from the Internet. The

server or servers connecting to the Gateways must be separate from the database servers that generate accounts, and must be protected from malicious access through suitable firewalls and security policies. There must also be suitable firewalling between these servers and the database servers, so that the database servers cannot be attacked even if the SP servers have been jeopardised.

Finally, normal precautions must be taken to ensure sufficient uptime to enable normal operations to continue to the satisfaction of consumers. In most cases, round-the-clock operation is all but a minimum.

Each SP must have the ability to communicate securely with each Gateway. In addition, the SP must know the serial numbers of all MUs in the Gateway's domain. This requirement is not unreasonable, as an SP would generally have a contractual relationship with each MU owner or user. The MU serial numbers can be associated with the Gateway during installation.

The SP does not need to know the keys for each MU, as the communications with the MU will be set up by the Gateway.

The SP premises can be assumed to be fully trusted, as it is not in the SP's interest to disclose the keys used for communicating with Gateways. That being the case, an SP's cryptographic operations can all be done in a standard PC-type platform. If the speed offered by a PC platform is inadequate for the number of Gateways, a separate cryptographic coprocessor can be used. There are no specific security requirements for this coprocessor, as secure key storage is not required.

The SP must have a database server that keeps track of MU identities, MU meter readings and account holder details. The system must generate monthly or other periodic accounts. It must also administer the termination process for clients that are not fully paid up.

The SP must typically await connection requests from Gateways, as the Gateways themselves are unlikely to be permanently attached to the Internet. It must also keep track of connection schedules, so that a Gateway that is behind schedule can be pursued otherwise (e.g. by a mobile inspector that can investigate its status).

Finally, an SP must be able to issue instructions to a Gateway. These instructions might include routine maintenance tasks, including meter readings, and occasionally specific instructions to an MU that might include cutting off or turning on the relevant service.

5.2 The Gateway

The Gateway server acts as an intermediary between the SPs (through a high-bandwidth network such as the Internet) and the MUs (through a low-bandwidth network such as a FAN).

The Gateway is privy to all communications passing through it. However, because the Gateway server is physically unguarded, one cannot assume that all communications passing through the Gateway may exist in plaintext form within it.

The Gateway must therefore have a SAM within which all messages are decrypted and re-encrypted as required. This SAM must be tamper-proof, so that the keys stored within cannot be obtained from the unit. Even when the SAM has been stolen and is in possession of the attacker, it must not be possible to obtain the keys and use them to attack the system.

A single Gateway has to retain several keys securely:

- **Admin Key:** A symmetric key for routine communications with the TC.

- **Special Admin Key:** Another symmetric key used infrequently for special communications with the TC. This key can be used for emergency action when a normal key has been jeopardised, or for routine key replacement

activities. Because the Special Admin Key is very important to the security of the system, its use must be restricted to the absolute minimum to ensure that cryptographic attacks cannot be launched against packets encrypted under this key.

- **SP Keys:** A separate key is required for every SP with access to MUs on the Gateway's FAN. In principle, there is no limit to the number of SPs potentially involved. However, in practice the number of SPs is likely to be limited to the number of services (gas, water, electricity etc.) being offered. More than one SP might provide a specific service, but the number of SPs per gateway is still not likely to exceed half a dozen or so.

- **FAN Key:** A symmetric key for encrypting traffic within the customer premises, including all communications to and from any MU on the FAN. All FAN communications are between the Gateway and an individual MU, as no peer-to-peer communications are possible.

Furthermore, the Gateway must retain other parameters, some of which may be regarded as sensitive:

- **MU Serial Numbers** for all MUs on the FAN network.

- **Meter Readings:** Last known readings for each service on each MU. Given that there could, in principle, be many hundreds of MUs on the network, and that each MU could support half a dozen services, up to several thousand meter readings might have to be retained securely. With a meter reading of perhaps four bytes, several kB of secure non-volatile storage might be required.

- **MU Logging:** A log of communications with each MU, to ascertain whether a specific MU is due for routine communications.

- **SP Logging:** A log of communications with each SP, to determine whether a specific SP is due for routine communications.

- **External communications parameters:** A series of IP (or other) addresses and communications protocols for the TC and all SPs.

The Gateway must have a dial-up or permanent connection to the Internet (or other high-bandwidth network). If a permanent connection is available, the ability for each SP to interrogate any MU is considerably simplified. However, if only dial-up access is available, the Gateway must schedule all connections to each of the SPs on a predetermined schedule.

If parameters obtained from an MU indicate abnormal activity, the Gateway must immediately contact the relevant SP and report the abnormality. Abnormalities might include:

- A meter reading lower than the previous one.

- Non-zero meter increments for an MU that has been disabled.

- Impossibly high consumption figures.

- Abnormal usage patterns, outside of the client's credit profile.

The SP will then determine a course of action, based on database rules, and issue appropriate commands to the MU. The Gateway will decrypt the command, make a suitable log entry, re-encrypt the instruction for the relevant MU and forward it on the FAN.

5.3 The Meter Unit

The Meter Unit has a rather nebulous trust model.

On the one hand, there is only one sensible trust assumption: The MU is trusted, as anyone who can open the casing has access to the switches and valves controlling the services themselves. There is therefore no incentive to tamper with the electronic communications, as the services can be circumvented directly.

On the other hand, anyone with access to the insides of the device could eventually get access to the cryptographic keys. These keys are useful beyond the MU itself, as all MUs on a specific subnet (i.e. FAN) use a common key for encryption and decryption.

Any sensible cryptosystem is assumed to be designed under the assumption that a potential attacker knows everything about the system design, and only the key is kept secret. For this reason, a disclosed key can be assumed to destroy all security that a system has, as the attacker is already assumed to know enough design details to enable him to read all traffic on the network.

Clearly, therefore, a secret key for use between a Gateway and its MUs must be very jealously guarded.

There are two approaches to secure key storage:

- A **secure microprocessor** is used. Such microprocessors would typically have internal non-volatile storage for keys, or must encrypt keys stored externally, with a key that is not used to encrypt routine memory contents. If the encryption key is shared with routine or predictable parameters, a known-plaintext attack [C1] could be launched against the system. Furthermore, if an attempt is made to access the memory contents, the keys must be erased. A more comprehensive discussion of Gateway SAMs is provided in a subsequent section.

- A **smart card-based cryptography coprocessor** is used. In this case, keys are stored very securely in this SAM. Modern high-security devices [H2]

include comprehensive protection against a wide variety of attacks, including PA, DPA, DFA, direct probing and electron microscopy.

In both cases, the important issue is that no plaintext must be allowed to exist in a part of the MU that is open to attack. If one keeps in mind that it is relatively easy to insert an in-circuit emulator onto a PC board, there are very few solutions that are actually fully secure, unless an integrated (single-chip) solution is used.

A fully integrated SAM would analyse all received packets, check to see if the packet content is destined for the MU, and only return the relevant parameters to the MU processor as required. Transmissions intended for other MUs are simply discarded, with possibly a notification to the MU processor that the packet is not relevant. Even valid transmissions, intended for that MU, are not fully decrypted and passed to the MU processor, as not all parameters can be trusted to exist in plaintext format in the MU.

Achieving such a scheme is relatively easy. The MU processor can be a relatively simple device with relatively modest processing power and possibly even without non-volatile data storage. The SAM would contain the necessary non-volatile storage. The MU processor would take care of communications, including bit-by-bit transmission and reception from and transmission to the powerline modem chip. The MU processor would also be able to communicate with the SAM. Suitable commands might include:

- **Increment Counter *n***. There might be several counters, for several services, each of which must be maintained separately.

- **Decrypt Packet**. This command would be issued once a complete packet has been composed. The complete packet is presented to the SAM for decryption. The SAM could respond with a status code named *NotForMe*, or with an instruction to the microprocessor.

Valid responses from the SAM to the MU controller might include:

- **Turn On *n***. This instruction turns on one service, identified by the variable *n*.
- **Turn Off *n***. This instruction turns off one service, identified by the variable *n*.
- **Report *n***. This response causes the processor to check the state of a specific service (*on* or *off*).
- **Not for Me**. This response is issued if a packet has been decrypted, and is not intended for the specific MU.
- **Invalid Packet**. This response is issued if a packet cannot be sensibly decrypted using the secret key in the SAM.

In both commands (controller to SAM) and responses (SAM to controller), the variable *n* refers to a specific service (water, gas, electricity etc.), and might take on one of up to half a dozen possible values within a specific system.

5.4 The Trust Centre

The TTP might be the SO, or a separate entity acceptable to all parties. The TTP's role in the scheme is to administer all the keys required to make the scheme work, and possibly to perform some housekeeping duties to keep the scheme working. The TTP's hardware installation that can be used to contact all Gateways in the scheme is known as the Trust Centre (TC).

If there is only one SP, or if a single SP is trusted by all other SPs, there is no need for a separate TTP. However, the operational requirements to be met by that SP are no different from those applicable to a TC, and this discussion makes no distinction between a separate TC on the one hand, and SPs acting as TTPs on the other.

A separate TTP is desirable, as clients who query the integrity of the system can make it very difficult for the SP if there is no impartial party to testify as to the security status at the time of the complaint. Depending on the contractual relationship between the client and the SP, the TTP may also act as arbiter or adjudicator in disputes between the client and the SP.

Unfortunately, there is still some user scepticism about meters that do not have rotating dials and visible counter digits. It could be argued that the electronic versions are more secure and more robust, but the basic instincts of consumers and legal eagles might not quite have evolved to that point yet.

The TTP operates a Trust Centre (TC) that serves the following purposes:

- **Program key carriers** (including smart cards, SAMs etc.) with the correct keys on the production line, before distribution.
- **Retain keys** for future reference, if required.
- **Issue replacement key carriers** or additional key carriers if new subscribers must be added to a system.
- **Adjudicate disputes** between users and SPs, or act as expert witness in such disputes if they result in legal action.
- **Conduct routine maintenance of keys and software** in the field. This function might involve regular key swaps or *ad-hoc* firmware upgrades of gateways and even MUs, although the latter would involve physical visits in the field and would more typically be conducted through local authorities or agents. However, the TTP would remain responsible for the process and would supply suitable code carriers and installation devices.

A TC would typically require very secure facilities and procedures to ensure that keys are not jeopardised. The basic security of the TTP's facilities is extremely important when the integrity of the system is questioned, such as when a customer claims that the meter readings are incorrect. If the TTP cannot prove in a court of law, beyond reasonable doubt, that the system does not allow such discrepancies, a floodgate of claims will result. The solvency of the SO is therefore greatly dependent on the TTP's astuteness in information security.

To simplify key issuance and replacement, a TTP would typically use derived keys. These keys are derived from a serial number or other known parameter, using a secret key common to many units. Using an example where an MU behind a particular Gateway must be addressed, the TTP could generate the key in the following way:

- Use the Gateway's serial number (let's call it *G*) and *n* SP Master Keys (one for each SP) to derive the secret keys to be used when communicating with Gateway *G*. These keys might be called K_{Gn}, and there would be *n* of them for each Gateway. $K_{Gn} = f(G, n, K_{SPn})$. Depending on the technology used, the SP Master Key could be kept secret even from the TTP, and could be generated by each SP and programmed into a module that is used on the TTP production line, but only under closely-supervised access control. Strong auditing mechanisms would typically be in place to ensure that no additional key carriers (smart cards etc.) can be programmed without the SP's knowledge and consent.

- Use each Gateway's serial number *G* and a Master Key K_{TTP} to program key carriers for the MUs behind the Gateway, and the Gateway's own SAM. $Key_G = f(G, K_{TTP})$. Each MU is programmed with Key_G and its serial number.

- Use each Gateway's serial number *G* and an Administration Master Key K_{Adm} to derive a Special Admin Key K_{GA} for that Gateway. $K_{GA} = f(G, K_{Adm})$.

This key is programmed into the TC server and into the Gateway's SAM, for use only during key management and emergency procedures.

The key derivation scheme outlined above does reduce the security of the scheme in the sense that it uses a single key of which the disclosure could render the entire system vulnerable. A good example is K_{SPn} for a specific SP. If this key is jeopardised, an attacker can issue commands to any MU behind any Gateway, as if that attacker was the SP.

One might initially think that this situation is completely unacceptable. However, if suitable security measures are implemented at the TC and the SPs' premises, and the master keys are not stored on customer devices, there is no substantial risk, and this architecture is used in almost exactly this form in many payment systems. The secret (sic) is in the procedures used to safeguard the generation and use of the master key, and the fact that the master key does not exist outside the premises of the TC. If sufficiently strong protection exists in the issuing process and the hardware used, no significant threat to the scheme results.

If done properly, the derivation scheme offers tremendous benefits in terms of managing the scheme. If a specific landlord wants a replacement SAM for a specific MU, the TTP can simply program a new SAM, using only the serial number of the Gateway and the serial number of the MU within the premises. If the same landlord wants to add another subscriber to the same scheme, the TTP can simply look up the serial numbers already used in that building and issue a new MU SAM, using the next available serial number, the Gateway serial number and the relevant master key.

Indeed, if the scheme is administered properly, it offers lower risk than the alternative. If derived keys are not used, every key in every MU would have to be generated randomly and stored. It is virtually impossible to implement the same level of protection for such keys as one could for master keys, simply because of sheer numbers. One could argue, therefore, that derived keys offer the highest security

level in practice, because it is actually possible to control derived keys very securely because of the very limited number of master keys in existence.

As an aside, the derivation scheme proposed by Schwaiger and Sauter [F10] results in separate keys for every MU. These different keys make it very difficult to broadcast messages to all MUs, and require real-time derivation or storage of all the unique keys on a specific FAN. The approach proposed in this dissertation (a common FAN key for all MUs behind a specific gateway) facilitates broadcasting by the Gateway, and requires only that a single FAN key be stored, without the processing and storage constraints of having to derive a key for every MU on the network. The approach is no less secure than that proposed by Schwaiger and Sauter, as their scheme also requires a single common master key for all MUs. As keys are stored securely in each MU, and considering that only a single FAN is jeopardised if the key is found, finding a key is not worth an attacker's effort.

The final function of the TTP is maintenance of the Scheme. Both firmware and cryptographic keys might require routine maintenance:

- If the key lifetime is shorter than that of the Scheme, routine key replacement might be required. This key replacement is done using the Special Admin Key. "Routine" in this context must not mean "frequent", as frequent use of the Special Admin Key could eventually allow attackers to compromise its security. The Special Admin Key for each Gateway can be stored in the TTP server, or can be generated each time it is needed, using the Administration Master Key and the Gateway's serial number.

- If a key is jeopardised, the TTP can replace the keys in the field on request. The Special Admin Key can be used for this purpose, as key cracking is likely to be a very infrequent occurrence. In systems with unusually stringent security requirements, a second Special Admin Key can be programmed each Gateway, so that a way out is available even if the Special Admin Key is jeopardised.

5.5 Security Application Modules

5.5.1 General Considerations

A SAM could serve several purposes:

- Secure key storage.
- Encryption and decryption without disclosing the plaintext to the microprocessor.
- Additional cryptographic functions such as signatures, authentication, checksums and certificates.
- Secure storage of sensitive parameters such as counters.
- Non-volatile storage, where no such storage is available in the microprocessor (this function is not security-related and is only a secondary benefit).

In the context of this system, SAMs are used in several locations:

- The TC Server, to safeguard several master keys.
- The Gateway, to safeguard keys for FAN and Gateway to SP or Gateway to TC communications, as well as secure storage of MU counters.
- The MU, to safeguard keys and decrypt FAN packets without disclosing the contents to the microcontroller.
- The SP server, if the SP environment cannot be regarded as entirely trusted. Modern corporate governance principles encourage scepticism in management that make a SAM highly desirable, even where the environment is regarded as trusted.

All these applications share some requirements:

- The need for secure key storage.
- The need for encryption and decryption without disclosing the plaintext to the outside world.

In addition, both the MUs and the Gateways need to store counter values securely.

To understand the way in which design parameters for SAMs interact, we need to investigate design tradeoffs in any security system [C8].

Any security system, whether it be a physical access control system, an information security system, a safeguard or indeed a domestic security system, constitutes a tradeoff between security, convenience and cost.

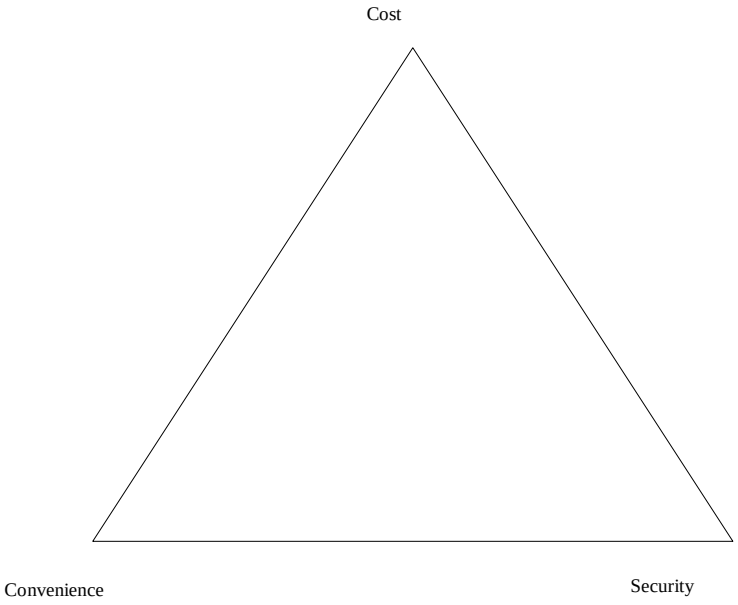

Figure 5-1: The tradeoffs in a security system

A domestic system would typically favour convenience and cost, rather than security. An extreme example is the common habit of leaving a key under the doormat. This solution favours convenience, as one doesn't need to carry a spare key. It also

favours cost, as again there is no need to have another key cut. However, it rates very badly in the security stakes. If a thief doesn't think of looking there in the first place, he only needs to see the key being hidden once, to have guaranteed access for life.

A banking system, on the other hand, favours security to a greater extent, with less emphasis on convenience and cost. In fact, a typical bank branch displays a variety of different tradeoffs. The foyer, where customers enter to conduct their business with the tellers, features only moderate convenience (or is it fun to go through a recalcitrant swing door?), moderate cost and moderate security. If one wants to enter the area behind the counter, more access control is involved, with an attendant increase in security, a decrease in convenience and an increase in cost. Entering the safe involves a lot of inconvenience, with multiple key-bearers present at once, complicated sequences and time-consuming multiple locks. It also costs a lot of money, but in return hopefully provides high security. In this case, the emphasis is clearly on security at the expense of convenience and cost.

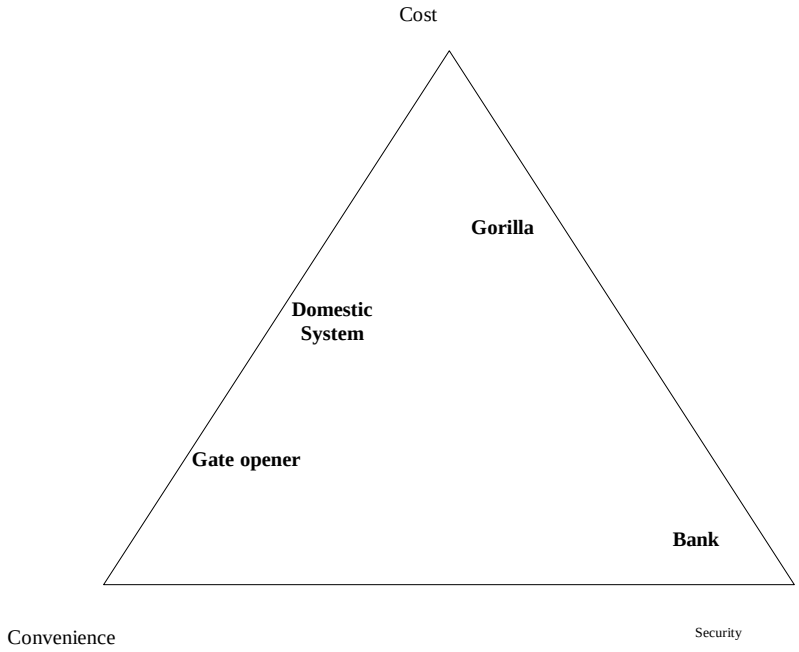

Figure 5-2: Specific examples of security tradeoffs

The figure above shows specific examples of tradeoffs. Apart from the banking system, which favours security, and the domestic system, which favours cost and convenience at the expense of security, two other examples are quoted. A remote gate opener adds convenience, at the expense of cost and security. A "Gorilla" steering wheel lock, used by many motorists, clearly favours cost, and provides a little security, but at the expense of convenience.

This simple triangle model is very useful in distilling one's thoughts about the tradeoffs in any security system. In some specific market milieus, other dimensions might have to be added to form a pyramidal shape. These factors might include safety (in the automotive and aviation industries) or even some less tangible parameters such as political acceptability or environmental impact.

SAMs are no different, as they are subject to exactly the same tradeoffs. When considering the current system, the three parameters in the triangular model are probably adequate, as the system is not subject to severe safety, political or environmental constraints.

In systems of this nature, convenience takes the form of quick response times, easy replacement of lost or broken modules, and redundant storage with easy recovery of corrupt data. However, each of these requirements comes at the expense of security. Achieving a reasonable level of security without sacrificing the specified level of convenience would cost a lot of money. One approach might be to use biometric systems to allow the legitimate owner a lot of flexibility in administering the system. Clearly, though, this solution would not be cheap.

It therefore follows that an affordable system with a fairly high level of security is going to involve some inconvenience. This inconvenience comes in the form of:

- **TTP intervention** if a module is lost or broken. This intervention is both inconvenient and expensive.

- **Relatively slow response times** because of encrypted traffic. This problem can be overcome by using more sophisticated processors, at a price.

- **Lack of interoperability between units.** An MU SAM can only work with a specific Gateway SAM, and *vice versa*.

The figure below shows a depiction of the tradeoffs involved in this metering system:

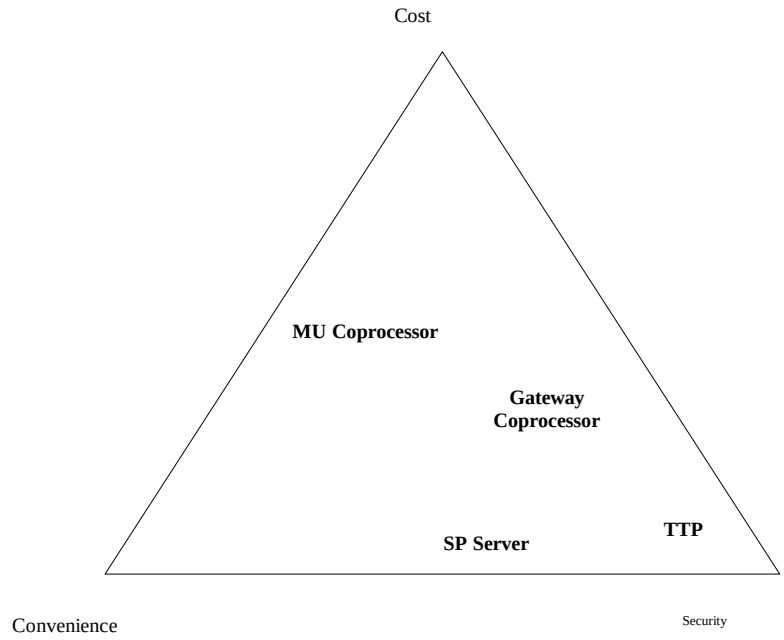

Figure 5-3: Security tradeoffs applicable to the SAMs

To enforce reasonable security, the SAMs must be able to protect the keys, even against physical attack. In simple low-security systems, a microswitch can set a non-volatile latch if the enclosure is opened. Such arrangements are commonly found in PC enclosures. SAMs must go one step further, and actually erase the sensitive data, including keys, when the enclosure is tampered with. Two terms are important here:

- **Tamper resistance** implies that the system is not prone to tampering. The mentioned strategy of erasing all sensitive parameters when the device is tampered with provides a high degree of tamper resistance. However, it comes with inconvenience, as even superficial tampering will cause the

module to cease functioning. High-security SAMs involve secure battery back-up, along with embedded wires, light detectors and possible even vibration sensors. They are normally based on high-security processors [H3]. Any attempt to open the enclosure results in the erasure of the keys and counters.

- **Tamper-evidence** means that tampering can be detected subsequently, either on the enclosure or through the network. A tamper-evident system might be as simple as a sticker that cannot be peeled off, and that must be damaged to enter the enclosure. However, it does not prevent the damage to the system in the first place. It only seeks to address damage after the fact. Clearly, this solution favours convenience and cost rather than high security.

In a situation where many utility meters might be compromised by the loss of a single key (such as the FAN Key in a large user installation), it is important that all keys must be strongly safeguarded. A tamper-resistant solution is called for, rather than mere tamper-evidence.

5.5.2 Considerations Specific to the Gateway

The Gateway must retain the following information about each SP:

- The relevant symmetric or public keys used to communicate with that SP.
- Restrictions on that SP's access to each of the MUs in that Gateway's domain.

The Gateway must retain the following information about each MU:

- The serial number of the MU within the Gateway's domain.
- The most recent meter reading for each SP from that MU.

The Gateway must also fulfil the following functions:

- Interpreting information returned by MUs to SPs to assess the realism of parameters returned, and raising the alarm if out-of-bounds parameters are detected.

- Polling MUs to determine whether they are still operational.

- A mechanism to prevent replay attacks, for Gateway/MU, Gateway/SP and Gateway/TC communications. This mechanism could be based on meter readings, time stamps or sequential counters. The sequential counter approach is probably simplest, but could be subject to intellectual property constraints.

These requirements rule out the use of smart cards as a SAM on all but the smallest of building schemes. This situation will change to some extent in due course, but for the moment the available smart card platforms do not offer the required combination of storage space and speed. For the moment, the requirements of ISO7816, which dictates the characteristics of all smart cards, also restrict the bandwidth with which a smart card can be addressed, placing a lid on the number of users that can be served by a single smart card SAM.

In a building with a sizable number of users (perhaps in the order of 100), a microprocessor-based module with tamper-proofing is probably a more sensible choice. Such a module would contain:

- A microprocessor.

- Some RAM with battery backup for key storage. This memory arrangement is preferable to EEPROM, as it has no lifetime limits and facilitates easy key erasure in the event of an attack. However, EEPROM or Flash RAM can also be used, provided that a suitable lifetime analysis is conducted during the design cycle. More information on memory lifetimes is provided in a later section.

- Enough non-volatile storage to store the parameters of the associated MUs. The lifetime of non-volatile memory must be assessed to ensure that the unit will not malfunction in its design lifetime.

- A mechanism to ensure that the keys are erased if the module is tampered with.

These requirements are all relatively easy to assess and design, except for the erasure mechanism. This mechanism requires a lot of thought if it is to comply with all the design requirements.

Traditionally, metal enclosures with microswitches have been used for many SAMs. Some even include some form of light detection to determine when the casing has been opened. These modules may stand up to casual attacks, but are certainly nowhere near secure enough for concerted attacks by assailants who have stolen the entire computer containing the SAM.

A Gateway SAM must be designed under the assumption that the entire Gateway computer can be stolen and taken to the attacker's premises. The attacker then has an unlimited amount of time and considerable facilities to investigate the computer and its SAM. The attacker may also have a full understanding of the system design. This assumption is not an unreasonable one, as a determined attacker might obtain participation from one or more members of the design team, or might steal several servers with the purpose of reverse engineering them methodically and obtain enough design information to finally crack one completely.

Against this class of attacker, simple microswitches and photo-detectors will clearly not make the grade.

Apart from more robust physical security, a SAM must also stand up to electrical and other non-physical attacks.

The following measures could be considered to enhance the security of a module intended for Gateway applications:

- **Power supply monitoring** to detect attempts to manipulate the power supply to corrupt the memory contents. A particularly vulnerable item is the program counter in a microprocessor that could be manipulated to execute code that is not intended for public scrutiny, such as testing code included only for diagnostic purposes. Proper brown-out protection is very important.

- **All memory must be protected against scrutiny.** Internal memory must be physically screened with metal layers, in accordance with modern practice in the microprocessor industry, and must feature randomly encrypted address and data lines to ensure that individual physical memory cells cannot be mapped to the appropriate bit locations in memory. The contents of external memory must be encrypted, using keys unique to that microprocessor. Different keys should be used for encrypting routine and sensitive data (such as keys), to ensure that very sensitive information is communicated infrequently, thereby denying an attacker the opportunity to gather enough data for a cryptographic attack on the FAN Key. Data should not be retained in memory for any length of time without being re-arranged, to ensure that attacks against RAM contents cannot succeed. Even after erasure, unpowered RAM can be attacked to retrieve information stored in that RAM for a long time, because of migration of semiconductor impurities and modified properties of oxide insulator structures [H4].

- **The physical tamper-protection must include potting.** The potting must include impregnated wiring, using suitably fine wire that cannot be individually identified without very specialised equipment. Wires should be run in twisted pairs, so that any attempts to cut or drill the potting compound or to fiddle with a single wire will result in a short-circuit with the adjacent wire, which in turn will lead to erasure of the memory contents. Potting compounds should be chosen to be susceptible to the same chemicals that can

be used to dissolve the potting compound, so that attempts to remove the potting compound must inevitably result in damage to the wire insulation, with resultant erasure of the memory contents.

- **Fail-safe operation:** The device should fail into a known safe state if the battery runs flat.

In short, given that the Gateway module retains several secret keys essential to the operation of the system, it must be designed to favour security. The inconvenience associated with losing a key and having to have the unit reprogrammed even when routine maintenance is conducted on the system is the price that has to be paid for achieving an acceptable level of security that will not jeopardise any other parts of the system.

A Gateway SAM, because of its relatively high level of activity, presents particularly serious memory endurance problems. Endurance has to be very carefully assessed, using the principles outlined later in this Chapter and the data sheets of the particular memory devices. In general, battery-backed RAM is likely to provide the most elegant solution with a large user base.

5.5.3 Considerations Specific to the Meter Unit

The Meter Unit is assumed to be physically secure. No other assumption makes sense, as an attacker with access to the casing is able to bypass the switching for any service in any case, without having to resort to tampering with the microprocessor's operation.

While this assumption is true in terms of the operation of the MU itself, it does not hold for the keys. Because a secret key is shared between many different MUs and their gateway, an attacker who jeopardises the key can affect many other MUs. For this reason, the keys themselves must be protected from scrutiny, even by the legitimate MU owner.

In principle, therefore, the cryptographic processing in the MU could be done inside the microprocessor, provided a way can be found to distribute and safeguard the keys, with possibly tens or thousands of MUs.

One option would be to use a secure processor with non-volatile storage in the MU, and to program each MU before it left the factory. The MU would then have to comply with similar security requirements to those outlined for the Gateway SAM in the previous section.

However, this approach is inflexible, as it makes it impossible to adapt the number of users for specific premises in the field, and also makes field replacement extremely cumbersome. It does have the advantage that the costs of a separate key carrier (e.g. a smart card) are avoided.

A smart card-based SAM offers a far more convenient carrier for keys. Identical generic meter units (i.e. without individual identity) can be kept in stock by the installer, along with groups of smart cards for every user group. If the TTP has retained details of the keys and serial numbers programmed into each card, additional cards for new users in a group or replacement cards for existing users can be ordered. The TTP would simply program the cards and supply them to the installer, who would then install those cards inside the meters and seal them in a tamper-evident way.

The problem with card-based key distribution is that the transfer process, with which the key is transferred into the MU, must be sufficiently secure that even the installer, who can easily monitor the communications between the key carrier card and the MU, must be unable to obtain the key. This requirement is not impossible to meet. However, because a secure communications session must be established between the card and the MU microprocessor, it requires substantial cryptographic processing capability inside the key carrying card, ruling out the use of cheap memory-only cards for this application.

If a relatively expensive smart card with cryptographic capability is to be used in any case, it makes far more sense to use this card as a SAM, such that no packets will ever exist in any MU in plaintext form. Only packets intended for that particular MU are decrypted, and only those parameters intended for the microprocessor are passed to it. Counters are maintained inside the SAM, using a series of instructions from the microprocessor to increment those counters when required. The counter can never be read by the microprocessor.

The SAM can also control all communications with the outside world, with or without intervention from the microprocessor. All communications intended for the Gateway or any of the SPs can be encrypted using the relevant FAN Key, for decryption by the Gateway SAM, and possible re-encryption for outside parties.

Finally, a hybrid system, in which a sophisticated key carrier is used to program many MUs, can also be used. In such a system, the carrier (which may or may not be a smart card) would carry many keys for many different MUs. The installer would then use this carrier to personalise each of the MUs. During this process, the MU would securely obtain the key from the carrier, and store the keys in its own internal secure storage. Only self-contained secure processors with internal key storage would be suitable for this application. The advantage is that one would not need a separate SAM for each MU. However, the advantage is limited, as each MU would still have to have a means of communicating with the carrier. If the carrier is a smart card, the MU would have to include a card holder. If the carrier is an external unit, such as a hand-held calculator or PDA, a connector for a serial port would be required.

If the key is to be stored in the processor, suitable security precautions must be taken to safeguard the keys. These precautions are very similar to those for the Gateway SAM, and the reader is referred to the relevant section for a more comprehensive discussion.

In the light of the operational requirements for the system under discussion, a smart card-based SAM with internal secure key storage is installed in every MU. The choice is made to facilitate field maintenance and replacement of MUs.

A secondary advantage of the SAM approach is that the microprocessor itself requires no non-volatile storage, possibly allowing the use of a cheap processor in the MU. However, this advantage is of limited practical use if the MU is to have on-line field-upgrade capabilities.

The requirement for field-upgradeability in the MU is not strong, as the bandwidth available in the FAN is inadequate for uploading of new firmware. The only way in which field upgrades would be feasible is if the processor included provision for new firmware in non-volatile memory, and that such upgrades could be uploaded to the controller using a handheld device such as a smart card or small computer. This approach would have to be orchestrated very carefully, as MUs with different firmware versions would exist inside the same network for a considerable amount of time while the upgrade is being conducted. Because manual intervention is required for field upgrades, a simple drop-in replacement microprocessor with EPROM or PROM program memory might be a better and more cost-effective choice.

Finally, memory endurance is unlikely to present as severe a problem in the MU as it does in the Gateway. Because only one MU's parameters have to be updated, write operations are not as frequent and are not as likely to result in destruction of the memory. However, especially if several services are controlled by each MU, careful analysis of memory lifetime parameters, particularly endurance, is required during the design phase.

5.5.2 Memory Lifetime Limitations

Virtually all types of non-volatile memory (magnetic disks, EEPROM, Flash RAM etc.) have lifetime limits. These limits are generally called "endurance" in manufacturers' specifications. Endurance generally applies only to write cycles, as most memories have practically unlimited lifetime when only being read.

Endurance figures of 10^4 write cycles for EEPROM and Flash RAM are typical, although some manufacturers quote or even guarantee 10^5 or 10^6 [H5]. Other technologies such as FRAM promise infinite lifetimes [H6]. However, FRAM is not universally regarded as a mature technology, and cynics might argue that even 10 cycles cannot be guaranteed.

Apart from endurance, data retention is another important parameter. Retention refers to the memory's ability to retain information without power and without re-write cycles. Retention is typically not a major problem, with most devices offering retention of at least a decade at room temperatures.

Clearly, therefore, endurance is the major factor to be considered. Two types of fatigue-related failures plague non-volatile memory devices:

- **Global failures:** The device charge pump or address decoder fails, leading to a complete device failure. Such failures could result in total inaccessibility of information, or to the inability to write new information into the device even though the contents can be read. Global failures are more likely to result when many different memory locations are written to regularly. Thus, common elements in the writing process are likely to succumb before the specific memory cells do.

- **Localised failures:** A specific location fails because of fatigue. This kind of failure is normally related to microscopic breakthrough in the device itself. This type of failure is more likely when a specific memory location is used repeatedly, rather than the generalised usage characteristic of the global failure. Most generally, this kind of failure can be simply circumvented by no longer using the failed address. This kind of failure can also be prevented by a suitable rotation strategy, writing information into different addresses alternately. This strategy cannot be used indefinitely, as the lifetime of global components will eventually limit the number of write cycles. However, on typical devices at least a dozen locations can be rotated with a commensurate increase in lifetime. Such a rotation

strategy requires careful thought to ensure that power interruptions can be adequately handled without corruption. A semaphore is often used for the purpose, to indicate whether a writing process was in progress when the device last turned off. However, the semaphore itself can easily become the lifetime-limiting factor that the rotation strategy was trying to avoid in the first place.

The analysis required to plan device lifetimes adequately is not always trivial. In some cases, especially in Flash RAM, the memory is written block by block, and even though the addressing scheme might make it appear as if only a single location is being re-written, an entire block might in fact be internally re-written during every writing operation. Writing sixteen words into a block in memory might actually involve re-writing each location in that block sixteen times. In such cases, the system programmer can greatly enhance memory lifetime by using caching strategies that make optimal use of the actual addressing scheme in the memory device, and by dividing the storage areas into blocks that coincide with the internal blocks used by the memory device.

However, the caching strategy must be designed to ensure that no irrecoverable information is lost if there is a power interruption. Such strategies would typically involve some kind of power supply monitoring and sufficient power accumulation (e.g. in a large capacitor) to allow the relocation of all cached information into non-volatile memory before the microprocessor itself loses power.

5.6 Protocols for Communicating between Components

5.6.1 Considerations Common to all Protocols

This section specifies security issues common to all protocols in the system. Operational issues are not addressed, except where they have an impact on the security of the system.

Because of the requirement for the Gateway to read all communications from each MU to the SPs, all packets are decrypted by the Gateway and re-encrypted with a

different key. The Gateway therefore maintains a key for each SP and for the FAN. All FAN packets contain the destination MU's serial number to enable MUs to distinguish between packets destined for them and packets for other MUs.

Provided that the relevant symmetric keys are not jeopardised, and key lengths and lifetimes are chosen well, all packets in the system travel between trusted devices, and should not be open to scrutiny.

However, provision must still be made in each of the protocols for protection against replay attacks. A replay attack is launched by recording packets on the network (whether FAN or the Internet) and then replaying them on the network later. As the packets are valid and correctly encrypted by the legitimate originator, they will again be accepted by the addressee, unless specific protection is included.

Replay protection can be provided relatively easily by encrypting a counter with every transmission. This kind of protection is easy in the Internet (i.e. between Gateway and SP or between Gateway and TC), where a few additional bits of information can do no harm. However, in the FAN, where bandwidth is at a premium, the decision on whether to encrypt a few more bits must be taken with great care.

The tradeoff in counter design is again something where convenience must be weighed up against security. A short counter (perhaps eight bits) provides some protection against replay attacks, and perhaps even a rudimentary retransmit request capability, without undue transmission overheads. However, it suffers from a short repetition cycle (only 256 transmissions), making it possible for deliberate replay attacks to succeed.

A long counter (perhaps 32 bits), on the other hand, provides very strong protection against repetition, perhaps even throughout the entire lifetime of an MU. However, this security comes at a price, as the additional 32 bits appended to the encrypted packet might increase transmission time by several seconds, in addition to the additional decryption time required at the sending and receiving ends.

A 16 bit counter might represent a good compromise, with around half a second of transmission time and a cycle time of around 65 000.

A counter scheme to protect against replay attacks has a secondary benefit. If one or more packets disappears between two devices, the recipient can immediately request retransmission when the next packet in the sequence arrives. In this way, all devices can ensure that no packet is omitted from the sequence.

An alternative strategy for providing replay protection is to include a time field (from a real-time clock) in every packet. This mechanism does not provide protection against missing packets, and is dependent on occasional synchronisation of real-time clocks. However, it is a strategy that has been suggested in some of the references, with a 150 s margin to account for badly synchronised clocks or transit delays.

5.6.2 Between Gateway and MU

The link between Gateway and MU is typically a narrow-bandwidth FAN, with a bandwidth in the order of 10 bps. The protocol design is therefore being done with a narrow-bandwidth link in mind, even though the prototypes might be built on high-bandwidth networks.

To secure communications between Gateway and MU on the FAN, a symmetric system is a must. Public-key schemes are not feasible, given the maximum block lengths dictated by bandwidth constraints on the FAN. The key length and block length required for reasonable security are in the order of 100 bits [C5], and should remain so for some time to come. An algorithm such as 3DES should prove more than adequate over the lifetime of the system. Its 112 bit key provides a safety margin of around 2^{12}=4096 over the minimum key length of 100 bits, providing another 18 years of protection against exhaustive search attacks under the assumptions of Moore's Law [C5].

When discussing communications between the Gateway and the MU, one must bear in mind that the Gateway cannot issue commands directly to the MU microprocessor.

Instead, this microprocessor passes all received packets to the SAM for decryption. In this sense, the SAM is the master controller for the MU, even though the microprocessor is coordinating all communications with the Gateway and with other MUs.

The following elements must be passed between the Gateway and the MU:

- **The destination MU serial number.** This serial number need only be unique within the Gateway's domain. If an arbitrary maximum of 1000 MUs per Gateway is imposed, ten bits will suffice for this field. One of these serial numbers (e.g. 0 or 1023) must be assigned as an "All" address to enable the Gateway to address all MUs simultaneously.

- **A flag indicating the direction of travel** (MU to Gateway, or Gateway to MU). A single bit is required.

- **An operation code, specifying operations to be executed** (e.g. returning a meter reading, turning specific services on or off etc.). These operations are similar to those issued to the SAM by the MU microprocessor, and a total of sixteen codes should suffice for all future events. Four bits should therefore be assigned for the opcode. A list of possible instructions is provided later in this section.

- **Service number to which the operation or request refers.** No more than four services should be required, therefore only two bits are needed.

- **Clock readings for synchronisation purposes.** To ensure compatibility with typical high-security microprocessors [H3], these fields should allow around seven bytes (i.e. 14 BCD digits) to specify hours/minutes/seconds, year/month/day and the day of the week. However, given the constraints on bandwidth, such extravagant fields are not feasible, and a more modest timekeeping scheme might have to be devised. Two possible approaches spring to mind: Firstly, a granularity of an hour might well prove adequate for the application. If so, and

assuming a system lifetime of no more than 30 years (or 263 000 s), 18 bits should suffice. Alternatively, bearing in mind that the MU never needs to remain autonomous for more than 30 days according to standard utility company rules, the lifetime of the clock could be reduced. A 16-bit counter is adequate for a one-minute resolution for a period of more than thirty days. This approach relinquishes the luxury of absolute time, but still retains enough information to time-stamp transactions within the MU for future reference. The protective mechanism employed against replay attacks must also be borne in mind with this choice. If the clock rather than a simple sequence counter has been chosen to provide replay protection, far more accurate clock resolution is required.

The following elements must be passed between the MU and the Gateway:

- **The source MU serial number**, unique within the domain (10 bits).
- A one-bit **flag indicating the direction of travel** (MU to Gateway, or Gateway to MU).
- **Device status** (service states etc.). Six bits should suffice for up to four services.
- **Meter readings.** These readings could be up to 24 bits in length [H7].

Provided that the key storage precautions mentioned in a previous section are complied with, a single key can be used to encrypt all communications on the FAN. Each unit (MU or Gateway) need then only respond if it is being addressed. This approach could be used even if the SAM or smart card in the MU is only used for cryptographic functions, as each MU's SAM can decode all transmissions, yet only make available the decoded transmissions intended for its associated microprocessor. All other transmissions would simply be decoded internally, and never sent back to the microprocessor for action. The SAM might simply issue a "*Not for Me*" code after successful decryption of messages not intended for its MU.

To reduce the processing overhead, it might make sense to append some of the flags (e.g. the Direction flag and the serial number) to the packet in unencrypted format. These "open" data would enable an MU to instantly recognise those packets intended for it, and to decrypt only those packets. Considerable time could be saved, as the SAM would not have to decrypt every packet on the FAN.

However, it must be remembered that these parameters must also be included in encrypted form to verify their authenticity. Apart from the obvious bandwidth penalty, this duplication presents the attacker with information useful in a known-plaintext attack, reducing the effective integrity of the encryption scheme by the same number of bits included in the plaintext. For a ten bit serial number and a one-bit direction flag, the effective key length is reduced by 11 bits, reducing the search time by a factor of $2^{11} = 2048$. Given that the FAN block lengths are extremely limited by bandwidth constraints, sacrificing several bits of integrity and including several bits of redundancy does not seem to make sense.

The relative importance of brevity versus processing time, which in turn depends on the availability of bandwidth versus the processing speed of the SAM, will determine design choices in this regard. If there is very little bandwidth but the SAM is fast, it makes sense to transmit only the ciphertext, and decode each packet in the SAM to determine whether the packet is intended for that MU. On the other hand, if the FAN has relatively high bandwidth and the SAM is slow, it makes sense to transmit some plaintext along with the ciphertext to prevent having to decode each packet in the SAM.

A list of possible commands from the Gateway to the MU might include:

- **Turn on service *n*:** Service number n in the particular MU must be turned on.
- **Turn off service *n*:** Service number n in the particular MU must be turned off.

- **Turn off all services:** All services in the particular MU must be turned off. This command might be used in case of dire emergency, or when a particular subscriber's contract has terminated. If there is a cooperative agreement between SPs on a specific Scheme, this command might also be used to apply pressure to clients who are in arrears with one or more SPs, although legal restrictions on interruptions to basic services would have to be borne in mind.

- **Report service *n* state:** Report whether service *n* is on or off in that MU.

- **Read meter *n*:** Provide the counter value for service n in the MU.

- **Read all meters:** Provide the counter value for all services in the MU.

5.6.3 Between Gateway and SP

The connection between Gateway and SP is far more flexible, as the network used would probably be the Internet, or at least a similar network with relatively wide bandwidths. Normal speech-grade lines can produce reliable communications at speeds of at least tens of kbps, and most countries offer ubiquitous infrastructure with even more bandwidth.

In any case, the bandwidth constraints in the system revolve around the FAN, as its bandwidth is likely to be at least two orders of magnitude lower than that of the interconnecting network.

There is therefore considerable flexibility in designing the communications between the Gateway and the SP.

In line with modern design trends, a markup language such as XML might be used, facilitating easy integration of passed parameters into the databases used for administering the system and issuing statements to customers.

Strong authentication is required, as tampering with the communications by outsiders must be prevented. As all Gateways are programmed by the TTP, all SP keys can be programmed into the Gateway SAM on the production line. For reasons of performance, symmetric keys are strongly preferred, and within the constraints of the system, with its manageable key distribution problems, a symmetric key system is easily implemented.

Public key systems also offer some advantages. Specifically, if not all SPs enter the Scheme simultaneously, public keys offer a way for new SPs to add themselves to the scheme later, without TTP intervention. The SO can simply provide the new SP with the Gateway's public key, after which the SP can introduce itself to the Gateway and upload its own public key. All subsequent communications can then be protected.

There is some risk associated with this approach, in the unlikely event where a service is already in operation before an SP is assigned. In this case, a rogue "operator" can connect to the Gateway and introduce itself as a new SP. It can then manipulate the relevant counter and issue rogue commands.

Public-key systems are rather demanding from the point of view of the Gateways' SAMs. A much less demanding approach is to simply store as many SP keys as required in each Gateway SAM before shipping, and to assign those keys to new SPs as they join the Scheme. If derived keys are used, the TTP only has to retain the relevant SP Master Key and the Gateway serial number, and then program a suitable key carrier for the new SP as required.

Most Gateways will only connect to the SP intermittently, as its Internet access is likely to be a dialup connection. To save costs, the Gateway only needs to connect to the Internet at off-peak times when telephone charges are lower. The Gateway would therefore routinely connect to the Internet at regular intervals, connect to each of the SPs and the TC to determine if there is any traffic, and upload any possible traffic. It would then disconnect, and execute all pending requests by communicating with each

of the MUs on the FAN. Once all the replies have been gathered, it will await its next scheduled connection.

There are two exceptions to this *modus operandi*. On the one hand, an SP or the TC might request an urgent response. In this case, the Gateway would remain on-line until the relevant information has been obtained, and report it back to the requesting SP or TTP. On the other hand, the Gateway might detect a discrepancy in a meter reading from an MU or a replayed transmission from that MU that indicates possible shenanigans on that FAN. In this case, the Gateway would raise an immediate alarm by connecting to the Internet and alerting the relevant SP and the TC of the discrepancy. This situation would warrant making a phone call to connect directly, even if it is during peak periods with consequent higher charges.

The connection between SP and Gateway must allow the following functions and commands:

- **Read meter *m***: This command is passed to the relevant MU. A reading of the relevant meter counter is returned to the SP.

- **Reset meter *m***: This command is passed to the relevant MU to reset the counter value. This command should not be required during normal operations, and may only be used during initialisation to exclude power drawn during the installation process from the customer's account.

- **Turn on service *m*:** The service provided by the particular SP is turned on in meter *m*.

- **Turn off service *m*:** The service provided by the particular SP is turned off in meter *m*.

- **Verify service *m*:** The SP wants to know the current state (*on* or *off*) of service *m* on the particular MU.

- **Read all meters:** The Gateway must gather and return the counter values from all meters, applicable to the service provided by the specific SP. This command could involve considerable delay, and a mechanism must be included to allow the replies to be retrieved after a suitable delay.

Each of these commands is received by the Gateway, and passed to the Gateway SAM. The message is then decrypted, interpreted and re-encrypted with the FAN Key. The messages are then placed on the FAN, where they are received by all MUs and passed to their SAMs for decryption. Each SAM will then decrypt the packet and check the destination serial number and the direction flag to determine whether the packet is intended for its specific MU. If it is, the SAM passes the relevant instructions to the MU microprocessor for execution.

Responses follow the opposite route: Each answer initiated by an MU is encrypted by its SAM and transmitted on the FAN. All MUs and the Gateway receive the packet and pass it to their respective SAMs. Each SAM then decrypts the packet, and checks the serial number and direction flag. In this case, only the Gateway SAM will find a match, and will then issue appropriate instructions to the Gateway control processor or. One possible course of action, depending on the nature of the received packet, is to re-encrypt the packet with the relevant SP's key and to pass the resulting packet to the Gateway processor for transmission.

The following responses and commands can be passed from the Gateway to the SP server:

- **Responses** to the instructions from the SP (read meter, reset meter, turn on service, turn off service, verify service and read all meters). These responses can be passed singly or in batches. Batch mode is used when the Gateway operates in offline mode, and has accumulated responses during an offline period.

- **Alerts**, where a suspect meter reading is received from an MU. This command would generally only be used when a Gateway is in offline mode, and has detected a suspect meter reading and decided to connect to the SP to issue an Alert.

5.6.4 Between Gateway and TC

The communications between the Gateway and the TC are not dissimilar from those with the SPs. However, as TC communications are infrequent and can leave room for gross abuse of the system, longer key lengths and more elaborate authentication mechanisms can be considered.

The following functions are served by TC-Gateway communications:

- Addition of keys for new SPs being added to the system.
- Administration of Gateway servers or SAMs that are suspected of having been compromised.
- Routine key replacement after the key lifetime has elapsed.
- Replacement of keys known to have been compromised.
- Software upgrades in the server.

Although, in principle, the ability to upgrade software remotely looks like it might introduce unacceptable security risks, the increase in risk is minor. The Gateway server has no access to any plaintext messages, and can merely hand encrypted messages to the SAM for decryption and re-encryption. Under a suitable security architecture, anything that happens in the server can cause no damage to keys or messages, except to lose such messages.

Key replacement likewise does not introduce appreciable risk, as the key replacement protocol is conducted directly between the TC and the Gateway SAM. As the Gateway Server has no access to plaintext at any stage, it also cannot interfere with the key exchange protocol or obtain keys, even if rogue software is installed in the Gateway Server.

The following commands could be issued by the TC to the Gateway:

- **Replace SP Key *m*:** Each SP key can be replaced individually. As all SP keys are unique between one SP and one Gateway, this command must include a command issued to the relevant SP server to update the key for that particular Gateway. These keys must typically be derived from a new SP Master Key, or be randomly generated and shared with the SP in advance.

- **Replace TTP Key:** This operation is done with the secondary TTP Key. This key must be used extremely infrequently, to avoid providing enough ciphertext for cryptographic attacks. Under routine operations, the TTP Key is itself used relatively infrequently, and this key replacement option is likely to take place only in case of dire security breaches.

- **Replace FAN Key:** The symmetric key used on the FAN can be replaced. If this option is implemented, provision must be made for key changes in all units attached to the FAN too. Each MU must have a secondary administrative key in its SAM, and test each packet against this key if decryption with the primary FAN Key does not result in a valid decryption, or if the valid decryption contains a command to discontinue use of that FAN Key.

- **Set Date:** This command is self-explanatory, and can only be used for SAM microprocessors with internal clocks. A previous section contains an analysis of possible date/time schemes, including the required granularity and counter rollover time.

- **Set Time:** Similar to Set Date in its operation.

- **Render Transaction Log:** This command causes the transaction log to be uploaded to the TTP. Transaction logs must contain details of all SAM operations, including transaction details, handle of key used, party with communications and possibly more exhaustive details such as opcodes and counter values.

- **Global MU Shutdown:** This command would generally only be used in cases of dire emergency, and would cause the Gateway to turn off all services on all MUs.

6. Results

6.1 Summary

The security policy proposed in this document can be summarised as follows:

- A common shared secret key, unique to each FAN.

- MUs communicating through the FAN to a gateway server. Each MU has an integral security coprocessor, in the form of a smart card module. All received packets are passed to the coprocessor for decryption and interpretation. No plaintext is ever retained in the MU processor or memory, apart from inside the coprocessor.

- The Gateway decrypts and re-encrypts all traffic passing between an MU and the SP or the MU and the TTP. Plaintext only exists inside a security coprocessor, so that access to the gateway server (either physically or through a malicious applet) cannot jeopardise the system.

This proposed security architecture fully addresses all requirements of the metering system. The use of symmetric keys provides adequate security with reasonable storage and processing power requirements.

The proposed strategies for key management and distribution are also regarded as adequate for typical systems, as they provide a good balance between security, cost and convenience.

Although broadly similar solutions are described in the literature, the existing solutions fail to address one major requirement of the Iguana system: the use of an untrusted Gateway server. Because Gateway servers will typically reside in unguarded basements or meter boxes, reliance on a trusted Gateway server is probably naïve. Consequently, the use of secure coprocessors is vital.

The existing described solutions also make use of only smart cards as security coprocessors. Although smart cards provide an excellent platform for secure key carriers, they cannot address the communications bandwidth, processing power and storage requirements of a Gateway, where details of every MU on the FAN must be stored. The use of a SAM based on a secure microprocessor and with adequate tamper-proofing is therefore regarded as imperative.

The principles for implementing suitable coprocessors are described in this dissertation. Although exact details can only be determined once the operational parameters are fully determined, enough information is provided to enable the system designer to make suitable design choices and implement a solution that will provide the correct balance between cost, security and convenience.

6.2 Conclusions

The objective set at the commencement of the work has been achieved.

The proposed architecture complies with all the requirements, including the ability for the gateway server to raise the alarm if unreasonable meter readings are returned from any of the utility meters, without introducing any excessive sacrifices in cost, convenience or response time.

6.3 Further work

Several items must be investigated before a full-scale implementation can be considered:

- **SAM performance:** Quantify the response delays associated with the SAMs (including smart cards) in the system. Specifically, determine whether the use of a smart card is feasible in the Gateway, and if so, up to which number of MUs. Smart cards will probably suffice for very small premises. Larger user bases, with perhaps more than a few dozen users, will require a coprocessor with wider bandwidth for access by the server.

- **Find a crypto-algorithm:** Assess several symmetric cryptographic algorithms to make a sensible choice in the light of the actual microprocessor platform used for the scheme. The final choice might depend on the availability of a suitable smart card platform with a dedicated coprocessor for the chosen algorithm, and would have to take into account the capabilities of the microprocessor chosen for the SAM in the Gateway server.

- **Investigate the need for a TTP in a practical scheme:** The answer to this question would be highly dependent on the situation in a specific country. Issues to consider would be the extent to which SPs mutually trust one another, and especially the extent to which the expert testimony of the TTP might be required in potential altercations with customers. In a society where dishonesty is rife, or a particularly litigious society, a TTP is practically essential.

- **Design a suitable SAM:** The design would have to be resistant against all envisaged physical and logical attacks, including brown-outs, physical attempts to access the PC board and memory, attempts to remove potting by chemical and mechanical means, interception of communications between the SAM and the server or between the microprocessor and memory on the SAM and exhaustive-search attacks.

References

Field-Area Networks

F1. Lobachov, M and Pratl, G (Technical University Vienna): personal discussions, February 2002.

F2. Lobachov, M: *A secure Fieldbus/Internet communication system*, Technical University Vienna, unpublished.

F3. *Project Introduction: Iguana,* Technical University Vienna, unpublished.

F4. Schwaiger, C and Sauter, T: *A Secure Architecture for Fieldbus/Internet Gateways*, Proceedings, 2001 8th IEEE International Conference on Emerging Technologies and Factory Automation, 2001, vol. 1, pp. 279 to 285.

F5. Schwaiger, C; Sauter, T: *Security strategies for field area networks*, 2002 28th Annual Conference of the Industrial Electronics Society, vol. 4, 2002, pp. 2915 to 2920.

F6. Sauter, T & Palensky, P: *Feldbusse und das Internet—eine kritische Betrachtung*, Technical University Vienna, unpublished.

F7. Palensky, P and Sauter, T: *Security Considerations for FAN-Internet Connections*, Proceedings, 2000 IEEE international workshop on factory communications systems, 2000, pp. 27 to 35.

F8. Sauter, T; Lobashov, M; Pratl, G: *Lessons learnt from Internet access to Fieldbus gateways*, 2002 28th Annual Conference of the Industrial Electronics Society, IEEE, vol. 4, 2002, pp. 2909 to 2914.

F9. Lobashov, M; Pratl, G; Sauter, T: *Applicability of internet protocols for fieldbus access*, 4th IEEE international workshop on factory communications systems, 2002, pp. 205 to 213.

F10. Sauter, T and Schwaiger, C: *Achievement of secure Internet access to fieldbus systems*, Microprocessors and Microsystems, vol. 26, issue 7, 2002, pp. 331 to 339.

F11. Gordeev, M: *Security Architecture for Field Area Networks Connected to Internet*, Proceedings of the Fieldbus Conference FeT '99.

F12. Intellon Inc., manufacturers of powerline modem chip sets: http://www.intellon.com/

F13. Capelon, a Swedish manufacturer of powerline modems: http://www.capelon.se/pdf/ds-PD01-a.PDF

F14. Pratl, G; Lobachov, M and Sauter, T: *Highly Modular Gateway Architecture for Fieldbus/Internet Connections*, Technical University Vienna, unpublished.

F15. Wollschlaeger, M: *Mapping of Fieldbus Components to WWW-Based Management Solutions*, Proceedings of the Fieldbus Conference FeT '99.

Cryptography

C1. Schneier, B: *Applied Cryptography: Protocols, Algorithms and Source Code in C*, John Wiley and Sons, 1994.

C2. Crypto++ 4.0 Benchmarks, http://www.eskimo.com/~weidai/benchmarks.html. 3DES is taken to process a 128-bit (16-byte) block in 1,46 µs. RSA-2048 encryption/decryption take 0,9 and 64 ms respectively. ECC-155 produces figures of 21,5 and 13,5 ms respectively.

C3. Johnson, Don B.: *ECC, future resiliency and high security systems*, Proceedings of Certicom PKS '99.

C4. Certicom Responds: RSA 512-bit Challenge Factored, http://www.certicom.com/resources/news/news_082299.html.

C5. Matt Blaze, Whitfield Diffie, Ronald L. Rivest, Bruce Schneier, Tsutomu Shimomura, Eric Thompson, Michael Wiener: *Minimal key lengths for symmetric ciphers to provide adequate commercial security—a report by an ad hoc group of cryptographers and computer scientists*, published on the Internet, January 1996.

C6. Network Security—DES cracked! http://www.nim.com.au/security/se02001.htm#se02003

C7. Keating, G: *Performance analysis of AES candidates on the 6805 CPU core*, Proceedings of the second AES Candidate Conference, 1999.

C8. Burger, Chris R.: *Tradeoffs in security systems: Selling to the customer's needs*, presentation to the Microchip Inc. FAE Conference, Phoenix AZ 1996.

Hardware Platforms

H1. Borst, J; Preneel, B; Rijmen, V: *Cryptography on Smart Cards*, Elsevier's Computer Networks 36, 2001, pp. 423 to 435.

H2. Infineon SLE66C640P Short Product Information 08.01, http://www.infineon.com/cmc_upload/documents/028/863/SPI_SLE66C640P_0801.pdf

H3. Dallas Semiconductor (now part of Maxim) Web site: www.dalsemi.com

H4. Gutmann, Peter: *Secure deletion of data from magnetic and solid-state memory*, Proceedings of sixth USENIX Security Symposium, San Jose CA, 1996-07-22 to 25, http://www.uncwil.edu/Ed/INSTRUCT/burt/edn416/secure_del.html.

H5. Microchip Inc.http://www.microchip.com. A randomly-chosen product http://www.microchip.com/download/lit/pline/memory/ic/21709a.pdf shows 1 000 000 cycles endurance and 200 years retention.

H6. Ramtron FRAM products, http://www.ramtron.com.

H7. Standard Transfer Specification, STS Association Home Page, http://www.sts.org.za.

Appendix A: The Dallas DS5000 Microprocessor

An early device suitable for SAM use is the Dallas DS5000. The device uses a standard 8051-type instruction set, but includes some features specifically aimed at high security applications.

Dallas Semiconductor has been acquired by Maxim, and the product family has grown to include several members, the most secure of which is the DS5420. The present application is likely to favour the modest devices with low price-performance ratios. The description below covers one such device, the DS5000FP, and was obtained from the Dallas-Maxim Web site at http://www.maxim-ic.com/quick_view2.cfm/qv_pk/2945.

The DS5000FP is the original Secure Microprocessor chip in an 80-pin QFP. Based on nonvolatile RAM rather than ROM, it offers programming flexibility not possible with ordinary 8051s.

The program and data resources can be partitioned and managed by the user during and after installation. This facilitates frequent software upgrades, adaptive programs, customized systems, etc.

Different versions provide for internal memory space of 8, 32, or 64kbytes of NV CMOS SRAM. With a user-supplied lithium battery, the on-chip power management circuitry can maintain all nonvolatile resources for greater than 10 years in the absence of external power. A DS5000FP also provides crashproof operation in portable systems by saving the operating state and providing a power-fail reset, power-fail interrupt, and watchdog timer.

Optional security features include real-time memory encryption using a 48-bit user-selected encryption key [*Editor's note*: Contradicted by a later claim for a 40-bit key] and a security lock that over-writes memory if it is disturbed. The DS5000FP uses

dummy cycles on the memory bus to foil pattern detection. The interrupt vector table is hidden on-chip.

The DS5000FP is ideal for any 8051-type application that requires versatile programming, reliability, and portability, such as data logging and battery-powered applications. It connects easily to a Dallas real-time clock for obtaining time-and date-stamped transactions.

Key Features

- 8-bit 8051-compatible microcontroller, with an 8051 instruction set, four 8-bit bi-directional I/O ports, 128-byte scratchpad RAM, two 16-bit timer/counters and 1 UART.

- Reliability enhancements, including 10 year nonvolatile data retention, early warning power-fail interrupt, power-fail reset, precision watchdog timer, user-supplied lithium battery-backed SRAM.

- Optional software security, including encrypted software, security lock to prevent download (unlocking destroys contents).

- Ease of use and adaptability, including 8, 32 or 64 kB nonvolatile SRAM for program and data storage, ROM-based serial bootstrap Loader, on-chip serial port for software download, partitioned code and data segments and nonmultiplexed byte-wide address/data bus for memory access

- 5 V ±5% supply, up to 16 MHz system clock speed, 0 to 70°C operating range.