Security Policy Implementation Strategies for Common Carrier Monitoring Service Providers

Carl A. Gunter
University of Illinois

***Abstract*—There are increasing capabilities and demands for the remote monitoring of homes and their occupants. There are a variety of options for the architecture of such monitoring systems entailing trade-offs between privacy, security, cost, manageability and other factors. This paper considers the virtues of building *Monitoring Service Providers (MSPs)* based on the concept of a *common carrier*. The goal is to provide policy to support monitoring with limited risk to the monitored parties, the users of their data, and the MSP. We argue that advances in distributed computing, cryptography, and trusted computing provide enabling contributions to building practical *Common Carrier MSPs (CCMSPs)*. We illustrate this with discussions of applications in the areas of assisted living and electrical power metering.**

I. Introduction

A *Monitoring Service Provider (MSP)* is an entity that supports communications between a monitored party and a user party that acts on the monitored data. An example of an MSP is a company that monitors homes and commercial facilities for security and fire emergencies. When an emergency arises that requires action, the MSP contacts the owner, police, fire department, or other parties. The value of such an entity is to relieve the monitored and respondor parties of responsibilities they cannot or prefer not to bear. The MSP can seamlessly connect parties that use heterogeneous and changing communication interfaces while providing value-added services like routing and prioritizing communications. Moreover, the MSP may create economies of scale by aggregating monitored and respondor parties. For instance, security services monitor homes in many different police districts; this enables outsourcing of MSP functions to parties that can provide good quality at low expense.

A *common carrier* is traditionally a party that transports goods for others, including, under U.S. law, a telecommunications company like an ISP. Elements of U.S. law[1] provide protections whereby a common carrier that enables communications between parties (such as voice switching or data), but does not carry out content-specific processing on this data, does not assume liability for the content of the communications they enable. For example, if a telephone provider is used to plan a crime, then the provider will not be considered a conspirator. MSPs are at a borderline between common carriers in this sense and parties that process data since they add application-specific services that make them more responsible for the data they collect. In some respects this places them in a category closer to bulletin boards and email relay providers, who have argued for common carrier status as a way to protect themselves from the liability for how they are used by the parties whose communications they carry. In many instances it is desirable, as a matter of policy, for an MSP to act like a telecommunications common carrier. This provides valuable risk management: if the MSP is not responsible for content-specific information, it limits its failure risk to its primary functions such as convenience, efficiency, and reliability. A *Common Carrier MSP (CCMSP)* is an MSP that manages policy so that it strives to limit its own access to the data it carries to the maximum degree possible consistent with its key functions. In particular, a CCMSP uses policy management to limit the sensitivity of the data on its relays, backups, and communication links. The aim of this paper is to discuss two MSP applications that might benefit from CCMSP policy management and consider three policy implementation mechanisms that will provide valuable tools for this objective.

There are a variety of applications of monitoring that could benefit from a CCMSP policy. The security monitoring application discussed above is one of these, and, indeed, the providers of such services are eager to avoid direct responsibility for, say, the consequences of a slow police response to the break-in of a monitored home. The sensors installed by a security MSP include ones like motion detectors, which detect private data, but this data is generally not relayed to the MSP since it is only used to trigger an alarm. By contrast some of the next generation of monitoring systems will collect more data: sometimes this will be necessary for the applications in question and sometimes it will not. One good example is a service that collects health vital signs from monitored parties. Another is a service that collects information and sends control operations using computerized electric power meters over a data network. Both of these applications are rapidly emerging and could limit risks by the use of CCMSP policy management. There are a variety of serious technical impediments to basing the next generation of MSPs on CCMSP policy. These can be addressed with existing technology to some degree, but may also benefit from the use of several emerging technologies. Among these are advances in distributed computing, cryptography, and trusted computing. We illustrate each of these by considering their use in implementing security policies for CCMSPs for monitoring vital signs and electrical power usage.

In: IEEE International Symposium on Policies for Distributed Systems and Networks (POLICY '09), London, UK, July 2009.

[1] In particular, the Restatement of Torts (Rest. 2d Torts sections 581,612), the Digital Millennium Copyright Act (17 U.S.C. section 512), and the Communications Decency Act immunity for interactive computer service (47 U.S.C. sec 230.

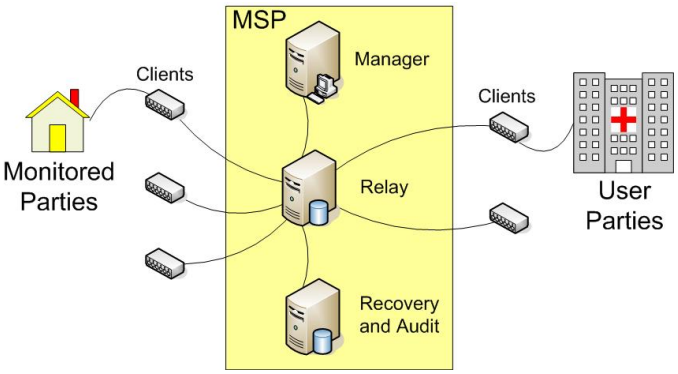

Fig. 1. Architectural Elements for a Monitoring Service Provider (MSP)

We consider the applications first, then sketch the enabling technologies and illustrate their potential relevance to the applications. A short discussion concludes.

II. Applications

Figure 1 shows a typical architecture for an MSP. A collection of client systems provide middleware that connects the User Parties to a Relay. The Relay assures that information from Monitored Parties is properly routed to the User Parties that need to see it. The figure illustrates *Assisted Living* in which a person is monitored at home for health vital signs which are then routed to a clinician who understands and acts on the received data. This type of MSP is an *Assisted Living Service Provider (ALSP)* [13]. Two important systems at the MSP are the Manager, which controls the addition of Monitored Parties and User Parties and governs the routing between them. The Manager also deals with communication and equipment problems that arise in the system, such as a failing client for a Monitored Party. The MSP provides a Recovery and Audit system whose functionality depends on the security and reliability policies of the MSP. For instance, only a portion of the exchanged data may be kept and it may be kept only for a limited time, or all data may be kept but only recent data is provided on line at the Relay. There are many possibilities for how the Clients connect. For instance, the system may have a publish-subscribe architecture where Clients poll the Relay for data, or there may be push communications at some nodes (like particular Clients) or on particular occasions (like emergencies). Data passed through the Relay may be batched like email relays or may provide a real-time feed like video surveillance. The aim of the MSP is to provide value-added services beyond what is typically provided by Internet Service Providers (ISPs), which forward network-layer packets on a best-effort basis. Additional services from MSPs may include security, reliability, recovery, and handling for errors and emergencies not currently or easily provided by Internet routers. A particularly important example of such a service is access control, since the MSP can manage the sharing of data in a more flexible way than routing.

In some cases monitoring is done without an MSP. For instance, in Assisted Living, one may have a Monitored Party who collects her own data into a Personal Health Record (PHR) on a home computer and conveys it directly to the clinician with a USB key carried to her next office visit. On the other hand, some hospitals have sophisticated IT staff who may be able to manage remote monitoring themselves so that the MSP is essentially a part of the clinician IT. These extreme cases may be common, but it will also commonly occur that Monitored and User Parties will want to avoid the burden of IT management of monitoring functions and will seek qualified third parties to provide this value. Such an approach may also improve scalability since the specialized monitoring function might be shared among a large number of parties. For instance, a single Monitored Party may use its MSP to supply information to several User Parties. MSPs may also improve reliability since the MSP has more knowledge of the requirements for this than an ISP would have for a TCP connection or a vendor for their comminicating equipment. For instance, a failure to hear from a node may cause the MSP to make a service call to an ISP, a vendor, or the owner of a client. Given this list of value-added services, the aim of policy management for a CCMSP is to limit access to only the information required to perform the functions required for monitoring without unnecessarily entering the domains best managed by Clients.

A key technology trend that is driving the development of MSPs is the increasing availability and use of networked sensor devices. Another example that illustrates MSPs and is driven by this trend is *Advanced Meter Infrastructure (AMI)* which enables networked reading of electrical power meters to provide billing information to *Electricity Service Providers (ESPs)*. AMI reduces costs for reading meters, provides better distribution grid status reports, and enables advanced features like demand-response *Meter Data Management Services (MDMSs)* assist ESPs in monitoring meter usage. MDMSs may be owned by ESPs, but may be associated with meter vendors or act as an independent intermediary between these types of parties. For example, an ESP may wish to diversify the types of meters it deploys and hire an MDMS to deal with the resulting heterogeneous sensor grid. The MDMS in turn may manage data for many ESPs in different regions, providing an economy of scale by specialization of function and broad geographical coverage.

ALSPs and MDMSs differ in many respects, but they also have some key commonalities. In particular, each type of MSP will need to deal with the risk that it either fails to serve its function of providing reliable and timely monitoring, or it will itself be subject to security violations that compromise the data that has been entrusted to it. For this reason, it seems that sound risk management recommends the principle that the MSP should have access to as little cleartext data as it can manage. In this way the risk of a compromise is minimized as far as end-to-end encryption will allow, aiming for the kinds of protection that a common carrier telecommunication provider enjoys. Achieving the status of a CCMSP using end-to-end encryption of this kind relies on middleware systems at the clients that store and use cryptographic keys. The CCMSP may be involved in this management (such as issuing certificates and keeping backup escrow keys) or may be largely independent of it (providing the greatest protection for the MSP but the least assistance

to the clients). Architectures like that of a common carrier MSP have been goals of security-conscious engineering for decades. For instance, Internet messaging (email) has striven for end-to-end security guarantees that avoid decryption on the email relay through a variety of strategies, especially public key infrastructure such as X.509 certificates or the PGP web of trust. Because of email's emphasis on high connectivity, this has been difficult to deploy. However, prospects are much stronger for applications that provide more focused services. For example, it is not difficult to see how a given MDMS might manage keys for its monitored parties and users.

III. Enabling Technologies

Given the challenges raised by end-to-end security in applications like email, one is led to ask whether there are new technologies that might smooth the way to practical implementation of CCMSPs. We consider three of them, spanning the areas of standards for distributed systems, new types of public key cryptography, and advances in operating systems and security co-processors that improve confidence in remotely monitored systems.

Service-Oriented Architectures. Service-Oriented Architectures (SOAs) [12] are a rapidly developing approach to the structuring of distributed systems based on course-grained standardized data formats characteristically based on web service protocols standardized by W3C and Oasis. This approach is fundamental to many modern software development systems such as Microsoft .NET. The standards are based on XML and SOAP (typically used for XML-based remote procedure calls) and provide an emerging foundation for B2B commerce on the web. In particular, they provide a good interoperable foundation for MSP communications in general and common-carrier MSP communications in particular. To see this in a concrete illustration, consider the problem of avoiding decryption at the MSP relay. Protocols like TLS [4], which are ubiquitously used for security of B2C ecommerce communications, provide only point-to-point encryption and therefore cannot directly support the desired common-carrier end-to-end encryption for a system like the one in Figure 1. However, SOA support for encryption (XMLENC [6]) can satisfy this requirement using off-the-shelf software from many different vendors.

Attribute-Based Encryption. Public key cryptography has done a great deal to reduce the key management problem for complex inter-domain systems. However, a next generation of public key technologies is emerging from ideas arising from Identity Based Encryption (IBE) [3]. IBE enables the use of common names as public keys (thus easing the burden on parties of remembering keys). Its decedent, Attribute-Based Encryption (ABE) [1] uses extensions of these techniques to encrypt data so that it can be read only by parties having a given set of attributes. This family of cryptographic techniques allows encryption to more effectively support access control on remote systems while also maintaining robustness against collusion among key holders. With ABE it is possible to use encrypted repositories to support Attribute-Based Access Control (ABAC) and by this means link encryption directly to the management of attributes of data and users.

Remote Attestation. Trusted computing aims to develop a base of highly reliable and secure software and hardware. One contribution in the area is technologies for remote attestation to enable a monitoring party to demand cryptographically protected evidence from a remote monitored party of the nature of the software and data being used on the monitored party. The Trusted Platform Module (TPM) [11], [10] implements remote attestation with co-processor and is being included in personal computers. The relevance of this feature to achieving CCMSPs lies in the goal of limiting the amount of sensitive data that the MSP needs to handle in the first place. Recalling the example in the introduction, the way a security MSP avoids handling sensitive motion detector data is to cause the data to be processed locally. In new types of MSPs it will be desirable to localize processing at the monitored site as much as feasible, and this feasibility will be enhanced by remote attestation.

IV. Illustrations

Let us now consider to some degree the potential value of these technologies in application contexts, specifically for ALSPs and MDMSs.

In the *drop box* architecture for ALSPs the clients 'push' vital signs readings into a repository from which they are later collected by clinicians. This is similar to the approach often used in hospitals where samples (like blood) are put in a drop box and collected periodically by the labs for analysis. It is possible to implement the drop box architecture as an CCMSP using web services [9]. The calls between the major components are all be done in SOAP and the key data items are placed under security from web services protocols. In particular, the data from the medical device passes through an Assisted Living Hub (ALH) where it is encrypted for both the ALSP and the clinician (User Party). The combination of signatures and dual encryption provides hop-by-hop security, end-to-end security, and non-repudiation. The implementation in [9] was done using standard protocols and their implementations from major vendors. A formal version of the protocol was analyzed under the assumption of a fully compromised ALSP and shown to preserve the integrity and confidentiality of medical readings (although routing information was compromised). For the medical data this is like a proof that the ALSP is a CCMSP.

ABE offers a way to send messages to groups of recipients described by their attributes while preserving end-to-end confidentiality. Existing techniques based on S/MIME provide a way to encrypt end-to-end, but only for a specific list of subscribers. By contrast *Attribute-Based Messaging (ABM)* [2], sends messages addressed to a receiving policy formed from a description, using attributes, of intended recipients. This can be secured end-to-end by encrypting (with ABE) under a policy for which anticipated recipients have the necessary keys. When one takes collusion into account, ABE is the only existing encryption technology that can provide this CCMPS-like end-to-end guarantee for a recipient list where the sender does not necessarily know the exact recipients and the server never learns the cleartext of the message. One thing that makes this approach interesting is the potential efficiency of establishing

a policy for the relationship between the 'receiving' policy (parties to whom the message is sent) and the reading policy (parties for whom the message is encrypted). For instance, if Alice needs to send a lab result to Bob's attending and GP physicians, then the reading policy might just encrypt under the attribute doctor rather than these more specific designations. This would provide some practical control over key management while satisfying legal regulations for end-to-end encryption of medical data.

A final example arises from the challenge of metering electricity usage without obtaining excessive amounts of private information about the monitored residence. In the future advanced meters will be able to gather quite a lot of data based on short time interval measurements and measurement capabilities like reactive power that will provide an ability to profile electrical usage to a fine degree. There are a variety of drivers for this trend such as *demand response*, which is the ability to buy electricity at variable prices so parties can intelligently shift their usage (for instance, running the dryer when demand and hence price are lowered). However, the resulting information can be privacy sensitive. Remote attestation provides the ability for the MDMS to leave much of this sensitive data at the monitored site (for possible use by interested Monitored Parties) while limiting the data that needs to pass to the MDMS. That is, the MDMS and ESP accept only the data they need but can use remote attestation to assure that metering is done by software they trust [8].

V. Discussion

Common carrier policy management for an MSP has something in common with the Internet 'end-to-end principal', which pushes functionality to the hosts in order to keep the functions of the network simple and avoid redundancy between network elements and hosts. This is an architectural consideration for the Internet; by contrast, the CCMSP objectives also include a policy management and enforcement aspect since MSP functions go beyond network connectivity. They vary with different application domains and even within a given application. For instance, it is not only the case that the policy issues for ALSPs are quite different from those for MDMSs, it is also likely that there will be significant policy differences between ALSPs in different contexts or between different parties in a given ALSP. This diversity applies to MDMSs as well. Adhering to a policy that the MSP retains as little information as possible provides a general guide that will minimize risk even in the face of this diversity.

Perhaps the most compelling motive for CCMSP is to limit losses in the case of an MSP relay compromise. An ALSP that never collects cleartext medical data will suffer limited exposure of private data if its relay is compromised. However, protection from involvement in law enforcement is also a significant concern. For ESPs there is already pressure to provide usage profiles for 'grow houses', which use heat lamps to cultivate marijuana [7]. There is reason to be concerned about an expansion of this type of investigation into information that could show when a consumer is at home and what types of appliances she uses [5]. Limiting the collection of these details will reduce potential liabilities for the utility and MDMS.

Acknowledgements. This work benefited from contributions of Omid Fatemieh, Michael LeMay, and other students in Illinois Security Lab, and from Jim Speta. This work was supported in part by NSF CNS 07-16626, NSF CNS 07-16421, NSF CNS 05-24695, ONR N00014-08-1-0248, NSF CNS 05-24516, NSF CNS 05-24695, DHS 2006-CS-001-000001, and grants from the MacAruthur Foundation and Boeing Corporation. The views expressed are those of the author only.

References

[1] John Bethencourt, Amit Sahai, and Brent Waters. Ciphertext-Policy Attribute-Based Encryption. In *IEEE Symposium on Security and Privacy*, pages 321–334, Oakland, California, 2007.

[2] Rakesh Bobba, Omid Fatemieh, Fariba Khan, Arindam Khan, Carl A. Gunter, Himanshu Khurana, and Manoj Prabhakaran. Attribute-based messaging: Access control and confidentiality. *ACM Transactions on Information and System Security (TISSEC)*, to appear.

[3] D. Boneh and M. Franklin. Identity based encryption from the Weil pairing. *SIAM J. of Computing*, 32(3):586–615, 2003.

[4] T. Dierks and C. Allen. The TLS protocol. IETF RFC 2246, 1999.

[5] G.W. Hart. Residential energy monitoring and computerized surveillance via utility power flows. *IEEE Technology and Society Magazine*, pages 12–16, June 1989.

[6] Takeshi Imamura, Blair Dillaway, and Ed Simon. XML encryption syntax and processing. W3c recommendation, December 2002.

[7] Robert Lamb. How grow houses work. *howstuffworks*, 2009.

[8] Michael LeMay, George Gross, Carl A. Gunter, and Sanjam Garg. Unified architecture for large-scale attested metering. In *IEEE Hawaii International Conference on System Sciences (HICSS '07)*, Big Island, Hawaii, January 2007.

[9] Michael J. May, Wook Shin, Carl A. Gunter, and Insup Lee. Securing the drop-box architecture for assisted living. In *ACM Formal Methods in Software Engineering (FMSE '06)*, Alexandria, VA, November 2006.

[10] J. Mccune, S. Berger, R. Caceres, T. Jaeger, and R. Sailer. DeuTeRium: a system for distributed mandatory access control. *Research Report RC23865, IBM TJ Watson Research Center, Feb*, 2006.

[11] R. Sailer, X. Zhang, T. Jaeger, and L. van Doorn. Design and implementation of a TCG-based integrity measurement architecture. In *Proceedings of the 13th USENIX Security Symposium*, pages 233–238, August 2004.

[12] David Sprott and Lawrence Wilkes. Understanding service-oriented architecture. *Microsoft Architect Journal*, 2004.

[13] Qixin Wang, Wook Shin, Xue Liu, Zheng Zeng, Cham Oh, Bedoor K. AlShebli, Marco Caccamo, Carl A. Gunter, Elsa Gunter, Jennifer Hou, Karrie Karahalios, and Lui Sha. I-Living: An open system architecture for assisted living. In *IEEE International Conference on Systems, Man and Cybernetics (SMC '06)*, Taipei, Taiwan, October 2006.