33

Key management and the security of management in open systems: the SAMSON* prototype.

G.G. Endersz and R. Zamparo

Information Security Department, Telia Research AB,
S-136 80 Haninge, sweden

1. INTRODUCTION

The goal of project SAMSON is the design, development and experimental evaluation of a harmonised and flexible security management system for the efficient administration and supervision of of security services in open networks. Part of this goal is to specify and evaluate the security functions necessary for the adequate protection of management operations and information. Figure 1. gives an overview of the interrelationship between management, managed services and the security of management. Although, SAMSON activities focus on security related entities of a system it is obvious that properly designed protection of security management can easily be extended to protect all other managenet functions.

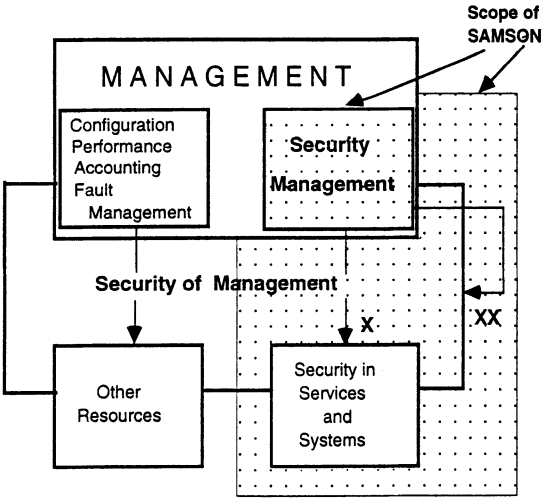

Figure 1. The relationship between security services, their management and the security of management.

*: SAMSON=Security and Management Services in Open Networks is a European collaboration R&D project within the CEC Race II (1992-95) program.

J. H. P. Eloff et al. (eds.), *Information Security — the Next Decade*

Key service, authentication, access control and security audit has been chosen as the targets of SAMSON security management. In order to avoid misunderstanding the following acronyms will be used. The process responsible to establish a secure association is called Key Service (KS), while the management of this process will be denoted as the Management of Key service (MKS).

This paper presents architecture and design issues of managed key services in open systems, and a prototype system based on the OSI environment and SAMSON management concepts.

One of the major design objectives for the SAMSON architecture was to support management of security services in different architectures from the same management platform. Figure 2. shows the chosen target architectures OSI, DCE and the X.500 Directory, with the actual security services that can be managed by the SAMSON protoypes. A short overview of the SAMSON concept is given in Section 3.1.

More detailes about the SAMSON architecture can be found in [1] and [2]. Reference [3] and [4] present more specific results on security management in the DCE and in the Directory (X.500) environments, respectively.

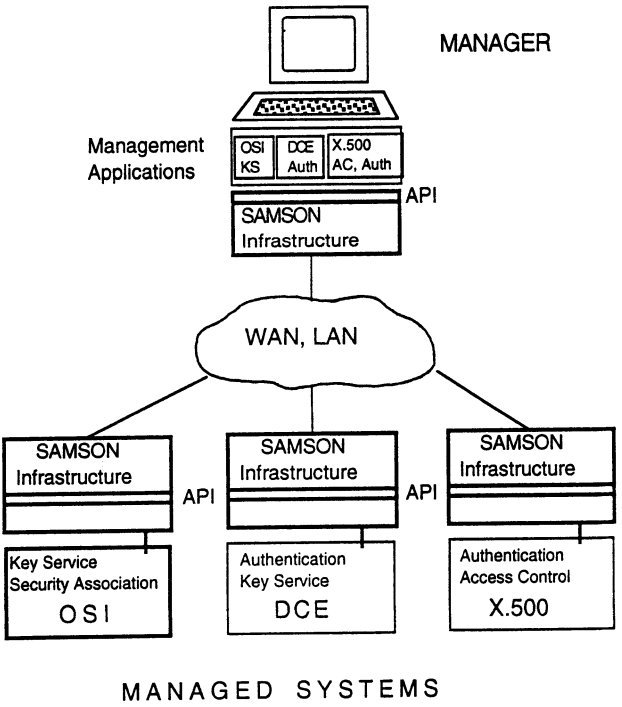

Figure 2. SAMSON management configuration with dissimilar target architectures.

In the Telia prototype the managed key service is used as one of the components to provide secure associations between open systems as well as between manager and agent when managing the secure associations of the above open systems. The secure associations for these two different purposes are provided by the same security services and they need the same kind of management. The arrows corresponding to the management of each are marked in Figure 1. by * and **, respectively.

2 KEY SERVICES FOR OPEN SYSTEMS

2.1. Requirements for a generic key service for open systems

The Key Service is essential for the operation of most security services, such as confidentiality, integrity and authentication. The Key Service comprises a key management protocol. In the following requirements for a key management protocol in open system environments are listed.

The key management protocol has to be algorithm independent and support both public key and secret key cryptographic systems.

It must be possible to do key management across security domain boundaries.

The key management protocol shall support different kinds of applications with keys and attributes. For example, in an OSI environment the protocol has to be able to support different types of security protocols.

A data network may be expanded, therefore, the key management system must be scalable ranging from a small to a large number of systems.

The key management architecture shall support any party initiating a communication with any other party.

The same form of key management should be applicable across a wide range of networking environments (e.g., to both LAN and WAN environments).

2.2. Model and interactions of the generic key service

The main components and their interactions are shown in Figure 3. Applications like file transfer or management operations are secured by security services which in turn rely on key services (KS) for the provision of long-term and short-term keys.In SAMSON have been specified the KS and an API for the access to the KS by the key user.

The Key Service includes an OSI key management protocol that provides generic key services to different users within the system. Interactions between the key services of two systems include the establishment of secure association by mutual authentication and the generation/exchange of session keys for use, e.g. by a security protocol. Another important function is the replacement of long-term keys, such as the KS´ own private key for asymmetricbased authentication and key exchange. For this purpose a specific, on-line CA functionality has been developed.

The protocols chosen by Telia KS to KS and KS to AC, respecitvely, are similar, based on the one proposed by IEEE 802:10D.

In addition some functions and interfaces have been specified and are implemented for the above managed components in order to achive full functionality and meet general requirements.

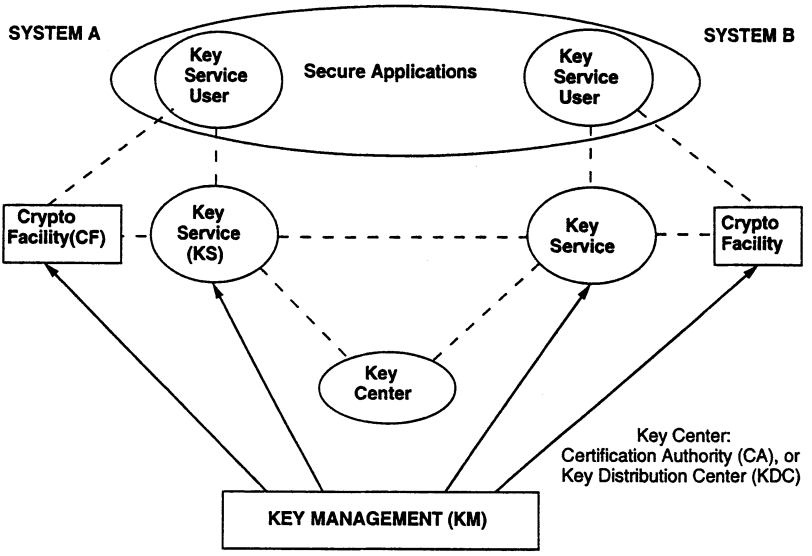

Figure 3. Basic model of key services and interactions

Both the KS and its user relys on the *CryptoFacility (CF)* for security sensitive processing, e.g. cryptoalgoritms. An API has been defined between the KS and the CF.

The associations between KS' of communicating systems A and B and likewise between the security services of A and B need a guarantee of trust, which is provided by the *Key Center (KC)*. The shape and functions of KC depends on whether the KS´ are based on symmetric or asymmetric crypto algorithms. In the symmetric case KC consists of Key distribution Centres (KDC) while in the asymmetric case we have one or more Certification Authorities (CA). In SAMSON the focus was on the latter case and functionality and protocol of an on-line CA have been specified.

The above model meets the requirements of independence, that is, independence of

-applications,
-underlaying networks and
-target architectures to be managed.

It is applicable to both symmetric and asymetric based KS schemes. In the prototype, presented in this paper, asymmetric-based key service has been chosen.

2.3. The SAMSON Key Service

The SAMSON Key Service can be used for establishing security associations for OSI lower layer security protocols. A security association denotes a security relationship between two or more communication entities. In the security association are included a number of security attributes. The security attributes tell whether a confidentiality and/or integrity service is included in a security

association, which encryption algorithm is used for providing the confidentiality service, etc.

The lower layer security protocol protects the traffic sent between two systems. The traffic can be confidentiality and/or integrity protected and a security label can indicate the sensivity of the data sent in the security encapsulated Protocol Data Unit (PDU). The lower layer security protocol can get a service from the Key Service through the API. The API has been defined in SAMSON. The API hides the underlaying security mechanisms for the security protocol. From the security protocol point of view it does not matter which security mechanisms are employed to create the security assiciation. The API can also provide the Key Services to several class of users, for example, several security protocols.

The services available to lower layer security protocols are:
- create security association
- delete security association
- update security association
- verify security association

3 MANAGEMENT OF THE KEY SERVICE

3.1. The SAMSON management architecture

At the logical level the main actors of security management are, in accordance with the OSI management model, the manager process (management application) at the manager side and the agent with the specific sponsors executing the orders of the manager on the agent side. The sponsors perform the mapping of abstract management operations onto real resources. The overall SAMSON management architecture is shown in Figure 4.

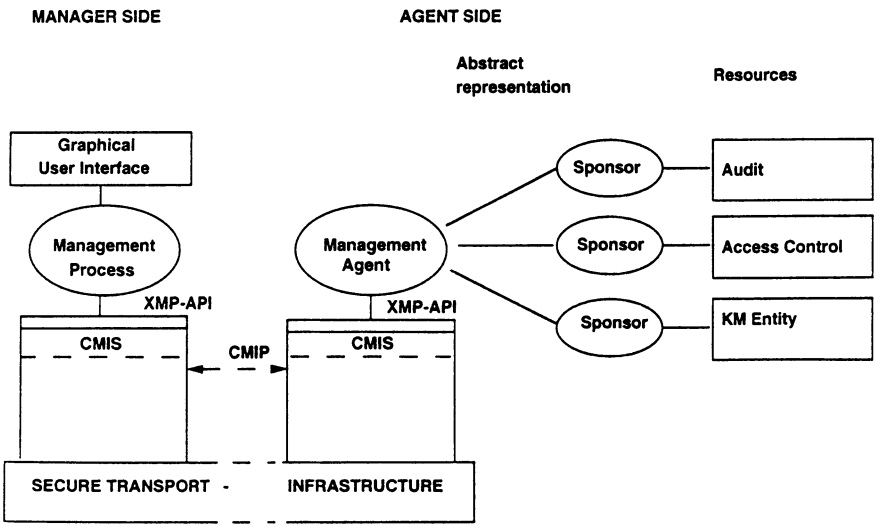

Figure 4. Security management using the SAMSON architecture

The common platform in SAMSON on both sides is the X/Open Management Protocol API (XMP-API) providing an application independent object oriented interface for all security management operations. XMP is a widely accepted industrial standard and it maps onto different management protocols. The one chosen by SAMSON is the international standard CMIS/CMIP.

As a communication infrastructure different choices are possible, e.g. OSI or TCP/IP. Managed objects and the structure of managed objects are defined in accordance with the ISO Guidelines for the Definition of Managed Objects (GDMO). More about the SAMSON concept and the relevant standards are to be found in the References.

The scope of SAMSON management of security services covers for each of the selected security services the specification and implementation of

management applications including graphical presentation,
agents /sponsors to carry out management operations and
managed objects.

3.2. Managing the Key Service

The Key Service needs to be managed. Example of a management action is the replacement of the certificate of the Key Service. When the certificates is about to expire it has to be replaced. Note that associated with a certificate there is a private key. SAMSON deals only with management actions that are performed over the network; off-line management actions are not within the scope of SAMSON. So in SAMSON has not been specified how to manually replace certificates. Because of the asymmetrical relationship between manager and agent in CMIP the protocol has not been used for the certificate replacement management action. Instead an on-line certification authority has been defined which upon a request issues a new certificate.

In SAMSON a number of managed objects have been defined in order to manage the Key Service. In this case the security manager uses the SAMSON management infrastructure. In the following paragraphs some of this managed objects are explained:

KM Entity Object:
The KM ENtity Object contains general information about the actual key management protocol used.

Authorization Object:
The Authorization Object has been defined so the security manager can define the key exhange algorithm used to create the session keys; specify whether a confidentiality service hase to be included in the security association; specify whether an integrity service has to be included in the security association; the security mechanism used to realize the confidentiality service; and the security mechanism used to realize the integrity service.

Security Association Object:
The Security Association Object has been defined so the security manager can monitor the security association thar are active within a system at a specific point in time. The security manager can also determine the security attributes that are included in the security associations.

Asymmetric Credentials Object:

The Asymmetric Credentials Objects is the top object of the managed objects related to a Key Service which provide services by employing public key technology.

Symmetric Credentials Object:
The Symmetric Credentials Objects is present in a Key Service which provide services by employing secret key technology.

Cryptofacility Object:
The Cryptofacility Object is the top object of the managed objects related to the cryptofacility. Examples of management operations related to the cryptofacility are to activate/deactivate an algorithm, to set minimal key sizes, to list the key identfiers of the session keys in order to determine how many keys are active at a point in time and specify whether keys have to be archived.

3.3. Requirements for security to protect management

Management operations and information need to be protected. While this statement is valid to some extent for all types of management, it is particularly important for security management, as all other security in the management system can be compromised if security management is or becomes unprotected.

In addition to the obvious requirement for protection of the managed resources of a distributed system, such as application servers, databases, routers and switches, there are a number of specific needs and objectives for the security of management in the telecom environment. The complexity of requirements is due firstly to the increasing number of different actors involved and secondly to technological developments in the area of distribution and large.scale integration of open systems.

The business and regulatory environment is undergoing radical changes. Deregulation of the telecom market will open access to network resources and to management information for service providers, customers and also for competing network operators. Correct accounting and the integrity of customer-specific information are two fundamental components of trust to be provided.

From the technology point of view the distributed management environment has certain characteristics that will influence security requirements and the possible choices to meet these requirements [5]. Open systems, such as the Telecommunication Management Network (TMN), means that proprietary interfaces and protocols are replaced by standardised and well known ones, which are easier to attack. Interconnection of management networks increase the overall vulnerability. In addition, there are administrative and contractual aspects of the interoperation of different security domains.

More than half of the computer crimes are committed by insiders. A high level of accountability will not only protect the system but also the integrity of honest members of the operating staff.

With the above in mind it was decided to introduce a certain level of security into the Telia SAMSON prototype, already at an early stage of development. In the large-scale distributed management environment of a network operator the protection of the manager-agent communication accross the network is one of the high priority targets. In the next section is described the concept and integration of security into the management infrastructure.

4 THE TELIA SAMSON PROTOTYPE OF MANAGED KEY SERVICES

Components of the implemented prototype can be divided into two groups. The first group includes the security protocol (TLSP) and the key service entity that will establish the security association between the communicating systems.Retrieval and verification of public key certificates of communicating parties is also supported, based on X.500-series standards. The implemented key service and security services are generic in the sense that they can be used to protect communications both by the management applications and other, general purpose applications

In the secong group, the prototype integrates all necessary components for remote management of the key services and security associations, such as the user interface, the key management application, the managed object classes and objects and the sponsors for mapping management operations onto the real resources.

4.1. The Telia Key Service prototype

The key management protocol used for handling security associations is based on the IEEE 802.10 key management proposal. According to IEEE 802.10 the lifetime of a security associtions consists of four different phases which are:

session key establishement
security atribute negotiation; testing of cryptographic compatibility
session key usage
session key destruction

The session keys can be established either using secret or public key technology. The IEEE 802.10 key management proposal has been submitted to ISO SC 21 foir becoming an OSI key management protocol standard.

The Telia prototype supports only that security associations are created by using public key technology for establishing the session keys. The prototype includes the Transport Layer Security Protrocol (TLSP). The certificates are stored in the X.500 Directory. As crypto facility the software package SECUDE has been used. The OSI-stack used for the implementation was developed within the Swedish R&D collaboration program IT4. This stack has been enhanced with the TLSP protocol, the IEEE 802.10 Key Management Protocol and tha API defined in SAMSON. It is possible to run the OSI-stack over ethernet and X.25. As X.500 Directory a Telia propriatary implementation has been used. The Key Service module consists of the key management protocol, the API and the X.500 Directory User Agent. The following figure describes the different security components:

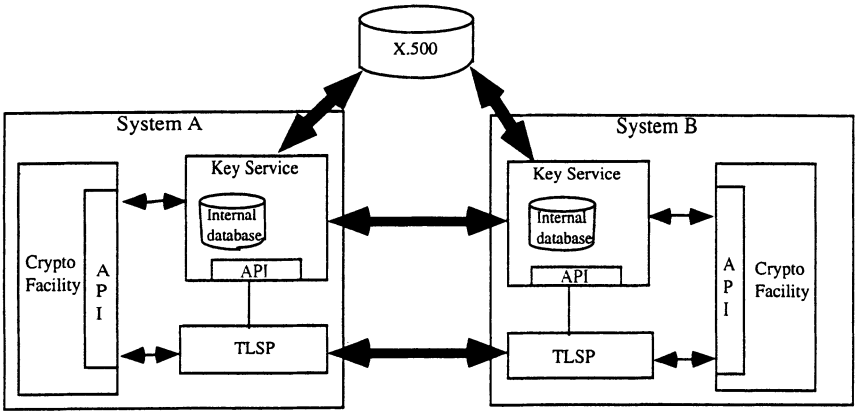

Figure 5. The integrated Telia SAMSON prototype

The thick lines in the figure above represent communication according to an OSI protocol specification. The communication between the Key Service entities is in accordance with the IEEE 802.10 protocol specification; the communication between the two TLSP entities is in accordance with the TLSP protocol specification; and the communication between the Key Service and the Directory is in accordance with the Directory Access Protocol (DAP). The thin lines in the figure above represent communication locally at the system.

When a transport connection has to be created it is disçovered that the transport connection has to be protected. The implementation makes this decision based on the transport selector used. The initiating transport entity sends a request to the Key Service to create a security association to an identified peer entity. The request, a create-security-associtión-request, is sent to the Key Service API. The Key Service performs a lookup in its internal database to find out which security services that has to be included in the security association, which security mechanisms that have to be used for realizing the security services and which key exhange algorithm that has to be used for establishing the session keys. The key exchange algorithm that has been used in the prototype is RSA based.

In the Key Service is included a Directory User Agent (DUA) which sends a request to is Directory Service Agent (DSA) in order to get the certificate of the peer entity, that is, the remote Key Service entity. The two Key Service entities can thereafter establish the session keys. Phase 1 is thus completed.

The session keys are created over an unprotected transport connection. The session key material is protected by security services provided in the OSI application layer. After the session keys have been created the two Key Services negotiate about which security attributes to include in the security association. The negotiation is protected by the created session keys. It is negotiated about whether a confidentiality service has to be included in the security association and whether an integrity service has to be included in the service association. The created security association is recognized by two security association identifiers. One identifier serves as a local identifier and the other as a remote identifier. The negotiation is performed over an unprotected transport connection, in this case too the security services are provided in the OSI application layer. At this point the security association has been created, that is, phase two is over.

The inititating Key Service sends a create-security-association-confirm to its TLSP-entity. The TLSP-entity is now able to open a secure transport connection to

its peer entity. The transport PDU (TPDU) is protected by the created security association. When the responding TLSP-entity receives the first TPDU (CR-TPDU) it is not able to recognize the included security association identifier. Therefore it sends a verify-security-association-request to its Key Service. The Key Service returns a verify-security-association-confirm. If the Key Service could verify the said the receiving TLSP-entity is provided with the necessary security attributes so it can remove the TLSP security encapsulation and apply the security encapusulation when it has to send a TPDU to the identified peer TLSP-entity. This is part of phase three.

When the transport connection has been terminated the initiating Key Service informs its peer entity that the security association has been deleted. This is the last phase of the lifetime of a security association.

4.2. Managing the Telia Key Service prototype

The following managed objects have been implemented in order to manage the Telia Key Service prototype:

KM Entity Object
Authorization Object
Security Association Object

The implementation with regard to management is represented by the figure below:

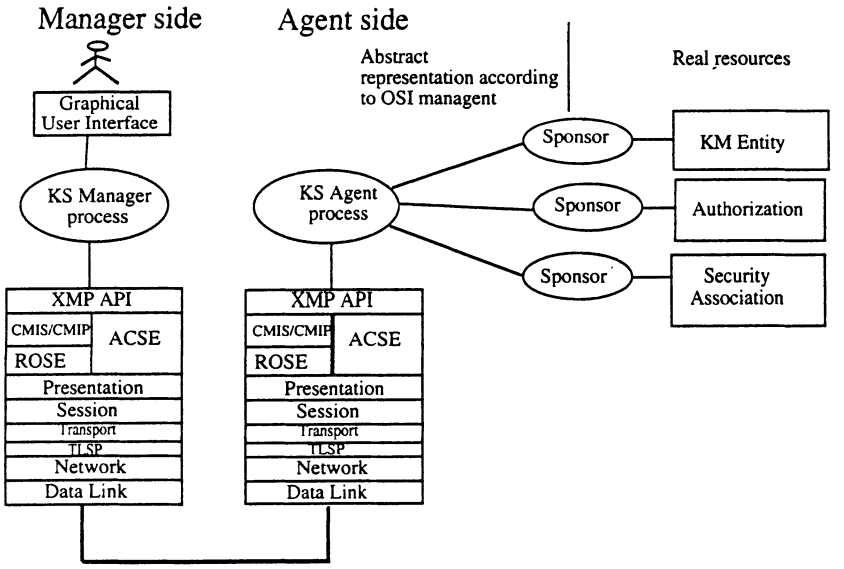

Figure 6. Management of the Key Service prototype

The manager accesses the SAMSON management infrastructure through the Graphical User Interface (GUI). The implementation of the GUI is based on MOTIF. A sponsor performs the mapping from the representation of a managed resource according to OSI management concepts to its representation in the real world and vice versa. Initially the manager opens an application association to the system he

or she wants to manage. As a result the actual Naming Hierarchy Tree will be represented on the GUI. The Naming Hierarchy Tree consists of the managed objects that are active at the point in time the manager opened the application association. The manager can navigate in the Naming Hierarchy Tree in order to change the value of one or several attributes of a managed object or to display the values of the security attributes of a managed object. The communication between the manager and the agent side is protected by TLSP. It should be noted that this is a recursive definition since the resource to manage is used to protect the management information sent over the communication network.

4.3. The demonstrator

The integrated protoype has been successfully tested and demonstrated in a remote management configuration. Two communicating systems, protected by a security association were managed by a third, manager system, controlling among others the activation/deactivation of the integrity and confidentiality services.

The demo-configuration was in accordance with the figure below:

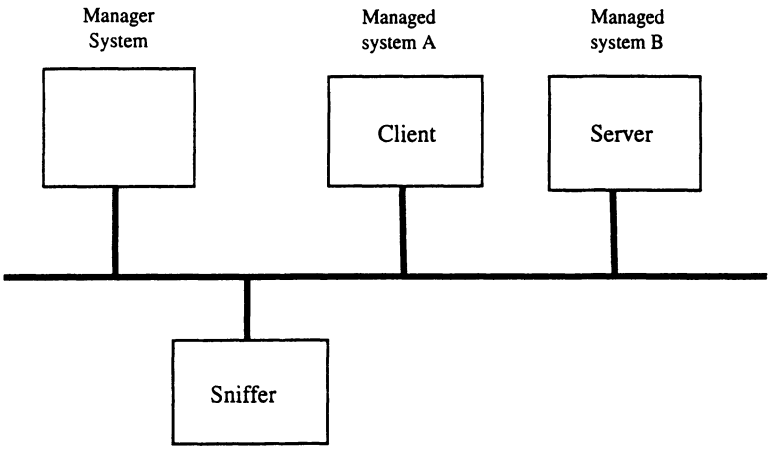

Figure 7. Demonstration configuration

A security association was established between the client system and the server system. The sniffer was then able to print out the clear text of the data sent between the client ans the server. The manager then opened an application association to the client system, that is, managed system A. The GUI displayed the active managed objects on managed system A. A managed object represented the security association created between the client and the server. An attribute of this object revealed that integrity but not confidentiality was included on the security association therefore the sniffer was able to print out the clear text of the data sent over the network.

The manager then changed the characteristics of a security association to be created between the client and the server systems by changing the appropriate attribute values of the Authorization object of managed system A. The application association between the manager system and managed system A was then closed. The Authorization object of managed system B was then changed in the same fashion. The security association between the client (managed system A) and the server (managed system B) was then terminated. A new security association was

established between the client and the server but this time the sniffer could only print garbage, because a confiidentaility service this time was included in the security association.

5 TMN and SAMSON

The Telecommunications Management Network (TMN) has been developed and standardised in order to provide a distributed, homogeneous and open network infrastructure for the management of large scale heterogeneous telecom systems. The architecure satisfies requirements of modularity and scalability and it builds upon and extends existing international standards within communications and management.

There are a number of commonalities between the architecures and concepts of SAMSON and TMN. Both rely on CMIP/CMIS as the management protocol and on GDMO as the basis of the definition of objects. Also the need for the security of management operations and information is similar. It is likely that a suitable solution for one of these architectures will be useful for the other. The benefits and synergies can be further enlarged by bringing in suitable elements of SAMSON into TMN.

Manager applications, agents/sponsors and managed objects have been specified and implemeted for authentication, access control, audit and key services by SAMSON. A great part of these results can be re-used in open management architecures based on OSI management.

The TMN standard presently does not include the necessary security architecture, service, protocol and interface definitions. There are two important interfaces (reference points) of information exchange in TMN. One is the so called Q3 (q3) interface, or reference point, between the managing system and the managed remote Network Element (NE). The management protocol at the Q3 interface is CMIP communicating through the OSI stack. The other important interface is the X-(x) interface, defined as the common point of information exchange between two managing systems, like in the case of cooperating managers of two different, but interconnected security domains.

Based on experience and recent analyses there is wide agreement among experts that the following security and management services for TMN form a minimum set of necessary measures [6]:

-authentication of operators and communicating entities;
-access control to protect resources and sensitive management information;
-integrity (in part confidentiality) of transmitted and stored data;
-log and audit of all security relevant events;
-security recovery.

The option of creating security associations in the Telia SAMSON prototype allows the protection of communication between manager and agent. The security services data integrity and confidentiality, with the option of peer-to-peer authentication, are realised by standard conformant key services [9] and security protocol [10] and the resulting security architecture is easily applicable to the q3 reference point of the TMN model.

The tasks for continouing work are twofold. Firstly, more work has to be carried out in order to specify, integrate and verify further security services in the management environment, such as user authentication, access control and security audit. Secondly, an appropriate set of standardised security functions and

interfaces has to be selected and integrated with the management architecture in order to define security profiles.

Both technical tasks described above are addressed by ongoing work within the SAMSON project and contributions from that work can be expected in the near future. However, only international standards bodies can start activities and provide the framework to develop the common security architecture and profile, which will allow interoperable implementation of security services in management systems, e.g. in TMN, by different vendors. Presently, initiatives have been taken within ETSI to begin work on standards for TMN security.

One of the tasks tasksIn the TMN context is to investigate how existing OSI-security standards, e.g. X.509, GULS, etc, and evolving security architectures like SESAME [11] can be used to secure network management. The main objective of the work should be to define security profiles for the TMN Q3 and X interfaces.

6 ACHIEVEMENTS AND CONCLUSIONS

Management of key services and security associations has been successfully implemented, tested and demonstrated as part of the SAMSON security management environment. The prototype includes user interface, manager application, agent and sponsor and the managed object classes and objects related to the handling of keys and secure associations.

The successfull integration and operation of components originating from other R&D groups and from different programs proves the virtues of the open, modular concept and the extensive use of APIs in the SAMSON prototype environments. In the Telia protoype the XMP-API, agent module and the crypto facility SecuDe have been ported from SAMSON partners, while the key service and TLSP modules came from other programs within Telia Research. Various implemented security management modules have been exchanged among partners and are successfully operated in different SAMSON demonstrators [4].

The introduction of generic security services into the management infrastructure using a security protocol and an application-independent key service represents a step forward in the process of securing management opreations and information. The results show that existing and emerging international standards and collaborative programs can provide the necessary components to build and standardise a secure management infrastructure.

Some results of the work described in this paper are closely related to the standardisation process in the security area. One such result is the implementation and experimental verification of the IEEE key service and protocol. The key service API and the managed object specifications developed by SAMSON represent possible contributions to this specific area. Future development of a TMN security architecture can benefit from the concepts and components applied to secure the Telia protoype.

REFERENCES

[1] SAMSON - Managing Distributed Security Services, C. Capellaro, RACE IS&N Conference, 21 November 1993.

[2] SAMSON: Management of Security in Open Systems, S. Lechner, Computer Communications, Vol 17, Nr 7.

[3] Security, Authentication and policy Management in Open Distributed Systems, R. Hauser, at al, Proceedings of The 10th International Information security Conference.

[4] Secure Multimedia Applications and Teleservices, H Bunz, at al, Proceedings of IWACA`94, 26-28 Sept. 1994.

[5] Information Security in a Telecom Perspective, G. Endersz, Proceedings of ITU Conference "Telecom Europe" , Oct. 1992.

[6] Security in TMN, R. Hagen.

[7] Key Management in Open Systems, R. Zamparo,Research Report, published by Telia Research AB, Sweden and by NIST, USA, December 1992.

[8] Key Management in Open Systems in Race II SAMSON, G.Endersz and R. Zamparo, Proceedings of the COST225 Workshop on Secure Applications with Public Key Certification, 22-23 Febr. 1994.

[9] IEEE 802.10D Key Management Protocol.

[10] ISO DIS 10736 Transport Layer Security Protocol (TLSP), 1992.

[11] SESAME Technology, An Overview, D. Pinkas and T. Parker, July 1994.