Security in the Telecommunications Information Networking Architecture – the CrySTINA Approach*

Sebastian Staamann
Uwe Wilhelm André Schiper
Swiss Federal Institute of Technology
Operating Systems Laboratory
EPFL-DI-LSE, 1015 Lausanne, Switzerland
{staa, wilhelm, schiper} @lse.epfl.ch

Levente Buttyán
Jean-Pierre Hubaux
Swiss Federal Institute of Technology
Telecommunications Laboratory
EPFL-DE-TCOM, 1015 Lausanne, Switzerland
{buttyan, hubaux} @tcom.epfl.ch

Abstract

The article presents the first results of the CrySTINA project. We analyze and structure the security problem domain in the TINA-C architecture and present our approach to provide the necessary security functionality in the form of self-contained application-independent security services and security mechanisms as part of the DPE functionality. The DPE is assumed to be basically provided by CORBA products. Therefore, we introduce the CORBA security specification and investigate if and how the identified TINA security services can be implemented using the CORBA security functionality.

1. Introduction

An essential requirement for the Telecommunication Information Networking Architecture (TINA) is security. TINA is intended to provide a comprehensive architecture for multi-service networks that shall enable multimedia communications and access to information for business and private users. In traditional communications networks that are solely dedicated to the telephony service, not much attention has been paid to authenticity, integrity, and confidentiality of the voice data carried. Efforts for security have been nearly exclusively focused on the secure and safe operation of the network itself and the protection against toll fraud. In future multi-service networks, this situation has to be changed. If commercially valuable interactions shall take place over these networks, users will require authenticity, integrity, and confidentiality for the information transmitted. The provision of security is becoming an important issue in the competition between TINA technology and other multi-service networks, first of all the Internet. In the Internet world, several approaches on various layers, all based on cryptography, are currently discussed or already in use, e.g., [3] [10] [4] [26]. In order to be competitive, TINA networks must guarantee at least the same degree of security. An important field of application is electronic commerce. The Internet does not yet have a specific architecture for electronic commerce. On the other hand, TINA with its business model and service architecture [23] together with the know-how and the customer base of the established telecommunication network operators as one of the driving forces of the TINA effort inherently possesses the appropriate infrastructure for electronic commerce. To benefit from this advantage, the overall architecture must provide the functionality to protect the transactions and to establish the legal bindings.

In order to avoid redundancy of functionality and for the sake of interoperability, security in TINA networks has to be provided in a consistent way. The security problem domain should be structured and security functionality should be provided as much as possible through general security services. A prerequisite for these services in a multi-party environment, such as a TINA network, is the existence of a security infrastructure that provides long term keys and supports the negotiation of security mechanisms and policies between different administrative domains. The means are mainly provided by cryptography. All application-independent security functionality should be available from the Distributed Processing Environment (DPE), since application and platform independent functionality is in TINA provided at this architectural level.

The provision of the security infrastructure is the subject of the CrySTINA (Cryptographically Secured TINA) project, a joint research effort of the Swiss Federal Insti-

*Research supported by the Swiss National Science Foundation as part of the Swiss Priority Programme Information and Communications Structures (SPP-ICS) under project number 5003-045364.

tute of Technology Lausanne, Siemens Munich, and the Swiss Telecom. As expressed in the name, security is realized mainly by cryptographic means. CrySTINA is part of a broader research effort aiming at a Secure and Reliable Distributed Processing Environment for telecommunication networks. This article presents the results of the analysis phase, i.e. how the security problem domain must be structured in order to provide security functionality as application-independent services and how CORBA security can be used for that purpose. It is an improved and enhanced presentation of the analysis given in [20]. Similar to the approach stated in the technical report on the TINA security architecture [22], we also strive to reuse established concepts from other standardization work, such as OSI, TMN and the OMG specifications.

In the following section, we analyze and structure the security problem domain in TINA. Then we investigate how CORBA security can be used for TINA and we identify additional security services and mechanisms that must be provided by the DPE. Section 3 presents the CORBA security specification. In Section 4 we propose how CORBA security may be used in TINA networks and identify the open issues for the TINA-DPE. Finally, in Section 5 we give an outlook on our ongoing and future work.

2. The TINA security problem domain

Security concerns all parts of a TINA system; it is pervasive and cannot be addressed in isolation. To cope with this complexity, it is necessary to structure the security problem domain in an appropriate way. All services and resources may be the subject to attacks. Attacks may be the illegitimate use of components or the modification of data, state or programs. They may occur through direct access to systems, data, or services from outside or through modification of messages exchanged between interacting components. Potential attackers are outsiders, but also other stakeholders in the TINA network. Motives of attackers may be the illegitimate use of services, fraud (e.g., in online businesses), toll fraud, eavesdropping on and observation of consumers or providers, or the deliberate prevention of service provision (denial of service attack). The ultimate goal of an attack may be achieved directly or indirectly. In the latter case, an attacker may install a backdoor during a first successful attack, which enables him later on (and possibly at multiple times) the actually intended misuse. Examples for backdoors are the modification of programs or access rights.

Each stakeholder in a TINA network has his or her own administrative domain [24]. We make the assumption that the administrative domain is the trust domain of the stakeholder. This assumption is based on the fact that in the regular case the installed hardware is under the physical control of the stakeholder and the software is installed by him or herself. The trust domain may in fact consist of various nodes under the physical control of the stakeholder that are connected by physically unsecure communication links. These links can be turned into secure channels by the use of symmetric cryptography without sophisticated management of keys, so that the connected nodes form a single trust domain. Since TINA supports personal mobility for consumers [23] [1], the current administrative domain is not necessarily administered by the consumer currently using it. However, in this article we do not cover the additional trust relationship between the consumer administrative domain and the consumer. This relationship is subject of our ongoing work. In this article, we assume complete trust between the user and his or her current administrative domain. Security within the administrative domain (intradomain security) is domain specific and is achieved by local means. Within his or her domain, the stakeholder trusts in the correctness of the installed software. Towards the outside, the administrative domain must be protected against illegitimate access. For interactions with other domains (interdomain interactions), limited trust relationships must be established. The communication channels between domains cannot be assumed to be secure. Therefore protection must be achieved by cryptographic means. Security must be provided to all parts of the TINA system that are involved in interdomain interactions. The security of each part is usually dependent on the security of many other parts in the administrative domain, nevertheless, all single parts have to be protected on their own. Figure 1 shows our structuring of the security problem domain.

System Security:
System security shall ensure that systems, mainly the hardware and the operating system, are not subject to intrusions. This concerns networking resources (e.g., network switches) and computing resources. It also includes the Native Computing and Communications Environment (NCCE) (operating system and communication ports), since intrusions may not only occur over communication ports of the NCCE that are used by the DPE, but also over other ports of the NCCE. The latter point concerns mainly the administrative domains of end users (consumers) whose Customer Premises Equipment (CPE), e.g., Personal Computers (PCs) or workstations, cannot be assumed to be exclusively used as the endpoint of the TINA network.

Service Security:
Service security is mainly concerned with the preservation of the integrity of service control. Service control includes among others the verification of whether a user is allowed to use a service (subscription) and the accounting for billing purposes. Both rely on the authenticated identity of the user. This must be supported by a protocol for the authen-

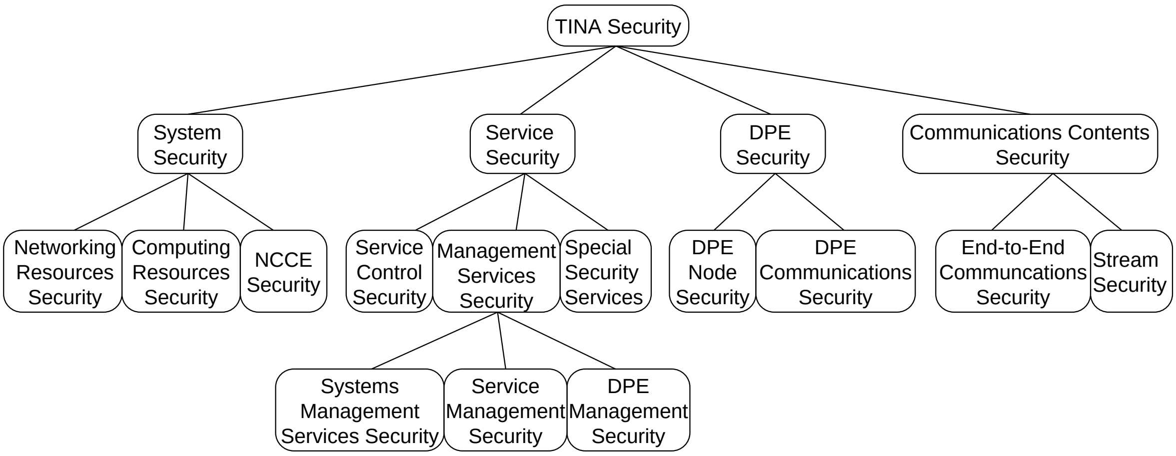

Figure 1. TINA security problem domain

tication of the user. Anonymous users of a chargeable service may be authenticated using anonymized, but chargeable (e.g. pre-paid), identities. The authentication protocol must guarantee that no secret authenticating information is revealed. This can be achieved best by mutual authentication of the user and the provider. The integrity of service control includes integrity of subscription verification and accounting. Access to the service functionality is controlled at two levels, the DPE level and the service level. At the DPE level, a coarse-grained access control based on the authenticated identities of the users involved in a session prevents attempts by others to invoke operations of the service components involved in the session. At the service level, the service logic implemented in the service component controls the access to service specific information and functionality based on the authenticated identities, context, and state information. Integrity and confidentiality of the messages exchanged between the service component's operational interfaces is achieved by the activation of the appropriate features of the DPE security services. These features must provide not only the protection of the integrity of the messages and their temporal order but also protection against interruption of the control connection itself, as we have demonstrated in [19]. Special cases of services are management services and special security services. Both require a potentially higher degree of security (e.g., stronger authentication mechanisms, longer cryptographic keys, or a physically better secured DPE node for their implementation). The special security services provide specialized security features, e.g., digital cash support, that are not present in every DPE node but are supported by dedicated providers (retailers or third party service providers) at the service level. The management services are concerned with the management of systems, services and the DPE. The security of management services is crucial, since illegal access to management functionality may be used for the implantation of backdoors. Of particular concern is the management of the DPE, which includes the management of the DPE security services.

DPE Security:
DPE security is mainly concerned with the prevention of illegal access to computational objects (CO) and CO groups as well as the protection of transmitted messages containing arguments, results, and exceptions of object invocations and notifications. DPE node security also provides the means to audit and report security relevant events on the node according to the audit specifications defined by the administrator (see also [6] [7]). DPE security includes the security of the DPE implementation and its basic services, such as the Object Services in CORBA. Since our architectural placement of security functionality allocates the general security services and mechanisms to the DPE (see Section 4), also the security of the security services themselves is part the security of the DPE.

Communications Contents Security:
Communication contents security is concerned with the authenticity, integrity, and confidentiality of the service contents information. Since all service content information is delivered in the form of streams, it deals only with streams. Streams are protected using cryptographic mechanisms, preferably stream ciphers [17] [18] or special ciphers for certain information formats, e.g., voice or video data. If the service implemented in the provider's domain does not require any modification of the stream between two users, they can have end-to-end security. Otherwise, only user-provider security can be provided. The management of the necessary keys is part of the service control.

3. CORBA security

The CORBA Security specification [13] has been released by the OMG to provide the model, architecture, as well as usage and administration interfaces for security in CORBA [12] systems. There are two levels of conformance. Some features are optional. Thus, not all secure CORBA implementations will provide the complete functionality specified.

The basic notion is the secure object invocation. For each object invocation, the request from the client object to the target object is subject to access control by the ORB security implementation. This access control may take place at the client side, the target side, or on both sides. Figure 2 shows a secure object invocation. The access is decided based on information bound to the target object, and/or information linked to the client object's request. The latter information is referred to as credentials. A credential consists of unauthenticated and authenticated attributes. Authenticated attributes are identity and privilege attributes. This general model enables a large variety of access control schemes, ranging from access control lists over capabilities to label based schemes. The scale of access control is not specified, but it can be assumed that implementors will provide access control down to the granularity of operations.

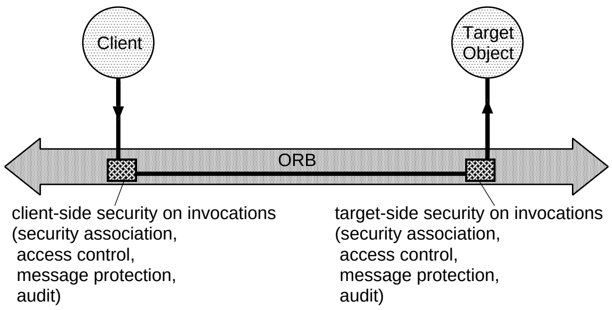

Figure 2. Secure object invocation

The client object acts on behalf of a principal. In most cases, the principal is a human user. In some cases, it may be a system entity, comparable to a system account on UNIX computers. For an object invocation, the credentials of the client object are contained in a dedicated credentials object which is referenced by the object representing the current execution context. The credentials object is created for the principal as the result of the authentication process of the principal.

Access control is only one of several concerns of a secure object invocation. A secure object invocation requires a security association between client and target object. In a security association, both parties trust the claimed identity of each other. This may require additional, mutual authentication with the creation of additional credentials, particularly if the two objects do not reside in the same ORB system. A security association will normally persist for many interactions. Depending on the security policy, integrity and/or confidentiality of requests and responses within a security association may be protected by cryptographic means. For security auditing, security relevant actions may be logged. As an option, CORBA security implementations may provide support for non-repudiation to the applications programmer.

The CORBA security document does not specify the underlying security technology itself. Instead, it defines interfaces for the use and administration of the security service(s) and interfaces to integrate security technology into ORB implementations. In a security enhanced ORB system, security is imposed at two levels, the administration level and the application level. Security policies for access control, message protection and audit are specified by the administrator. A security policy is represented by a security policy object. One security policy may be valid for one or more security domains. Each security domain contains a domain manager object which references the valid security policy object for this domain. The affiliation of an object to a security domain is determined by the administrator. The respective policies are in each case enforced on the object. At the application level, additional security measures may be enforced by the applications themselves. This may be done by the additional enforcement of administrator defined policies and/or the direct use of security features such as non-repudiation. The application enforced security measures cannot override administrator enforced policies.

There are four types of domains regarding security: security domains, security policy domains, security environment domains, and security technology domains. A security domain is the domain that is administered by one authority. A security policy domain is the scope over which a security policy is enforced. In most cases, it is identical with the security domain. A security environment domain is the domain in which the enforcement of the security policy may be achieved by local means, e.g. objects on the same machine. A security technology domain is a set of objects for which the same technology (e.g., Kerberos [11]) is used to enforce the policies.

Secure interoperation between objects depends on the membership of the objects to security technology domains, ORB technology domains, and security policy domains. Interoperability between objects in different security policy domains can only be achieved if both domains agree on a cooperation security policy for the respective interactions. This cooperation policy may be negotiated at invocation time or in advance. In the simplest case, only the security policy for the target is applied. Objects in different ORB technology domains can, technically, interact without problems using the Secure Inter-ORB protocol (SECIOP) as specified in [13], as long as the same security technology is used on both sides. (We do not consider the gateway approach for inter-ORB-interoperability to be used in TINA. However, the following would also apply in this case.) According to the CORBA security specification, interoperability between objects in systems with different security technology requires a security technology gateway. This is obviously not a trust problem if both security technology domains are in the same security domain (one common security administration). It may cause trust problems if the boundary between security domains (with different administrations) is also a boundary between security technology domains. For instance, assume one security domain (and security technology domain using asymmetric cryptography) providing the non-repudiation service using digital signatures and another security domain (and security technology domain using only symmetric cryptography) providing non-repudiation using notary servers. In general, a security technology gateway cannot be realized without the administrators of both security domains trusting each other or a third party that runs the gateway. A less restrictive solution would be to negotiate the security technology used.

The specification for secure interoperability between ORBs extends the CORBA 2.0 standard which specifies interoperability. The information which security technology the target requires and which security mechanisms it supports is part of the interoperable object reference (IOR). The Common Secure Interoperability Specification (CSI) by the OMG [14] allows the protocols of three security technologies within the SECIOP, namely SPKM, Kerberos, and the ECMA security protocol. If the interoperability between the ORBs is based on DCE [16], the DCE security technology based on the Kerberos protocol can also be used [13]. A recent proposal [15] wants to allow to base inter-ORB security on the Secure Socket Layer (SSL) [10]. Based on the information in the IOR, a security context acceptable for both sides can be determined. The establishment of the respective security association and the protection of messages are controlled by security tokens which are added to the Inter-ORB-Protocols. Key management is not explicitly dealt with in the CORBA security specification.

4. Providing security for TINA

Security features in TINA are implemented at various levels. In our approach, the DPE offers general security services and security mechanisms[1] to the applications. In the following, we will introduce our approach for TINA security, which is based on a layered structure. Figure 3 illustrates the layering. The usage relations are as follows. DPE security services are exclusively based on the DPE security mechanisms. The implementation of these mechanisms may directly use cryptographic mechanisms or may be built on available higher level security technology, such as Kerberos [11]. The underlying security technology may use the same cryptographic mechanisms as the DPE security mechanisms or proprietary implementations. The use of cryptographic mechanisms and/or higher level security technology may be accomplished through standardized interfaces (e.g., GSS-API [2]) to facilitate the integration of existing products into the DPE.

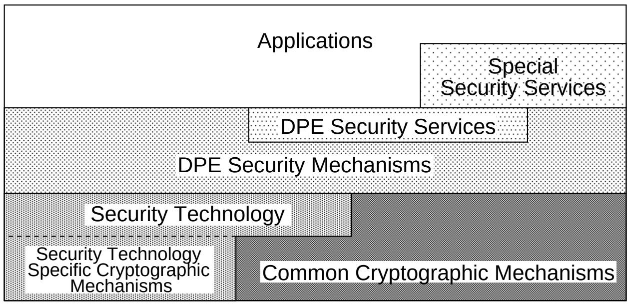

Figure 3. Layering of TINA security features

Above the DPE level, Figure 3 shows the special security services, which also rely exclusively on the DPE security mechanisms and services. They are used by TINA applications, but are application services themselves. The special security services are not implemented on each DPE node. Examples are electronic retail banking functions or notary services.

General security functionality is offered to the applications on each node as part of the DPE functionality. To ease integration in applications, as much functionality as possible should be provided as self-contained security services, i.e. the application is not concerned with how the security functionality is provided (e.g., which security mechanisms are used and on which cryptographic mechanisms or security technology they are based). The application is only required to use the DPE security mechanisms directly if the handling of the security mechanisms is service specific, e.g., if the verification of a digital signature is directly needed. The necessary DPE services and mechanisms must be provided by CORBA security and additional parts of the DPE that are addressed in the CrySTINA project. In the following, we investigate for each of the security topics introduced in Section 2 whether CORBA security functionality is sufficient and propose additional features if needed.

[1] Unlike the CORBA security specifications [13] [14], we use the term security mechanism in the sense as introduced in the OSI security architecture [5], i.e. for an abstract mechanism that can be used to provide one or more security services (e.g., a digital signature), but does usually not provide all necessary security functionality for the system.

System Security:
System security cannot be provided by CORBA security. It can only be guaranteed by a proper design, implementation and installation of the respective hardware and software as well as the protection of communication over unsecure links at the NCCE level. In operation, system security can be supported by auditing of security relevant events at the NCCE level and alarm reporting in case of serious incidents.

Service Security:
Service security relies extensively on DPE security features. In principle, the security of service control can be provided by the use of the CORBA security services in cooperation with the security relevant service logic. A crucial point is the mapping of security relevant domain types defined in the TINA-C architecture and in the CORBA architecture. Based on observations of trial implementations of the service architecture (e.g, [25]), we assume the mapping of each administrative domain in TINA onto one ORB system. This is reasonable because resulting from this mapping the interface between ORBs is a protocol interface, i.e. in the interaction with another administrative domain no potentially vendor-specific executable code from the other side is needed at the DPE level. For security reasons, we propose the following mappings of TINA domain types to security relevant domain types at the CORBA level:

- Each TINA administrative domain is mapped onto one security domain. This domain is also exactly one security policy domain. The mapping reflects that each stakeholder (i.e. administrator of a TINA administrative domain) has specific security interests and limited trust in other stakeholders and that a security domain should not contain equipment that is not under the physical control of the security authority. This mapping further implies that all interdomain interactions between service components (e.g., UAP-USM) rely on inter-ORB security and on the negotiation of a cooperation security policy.

- Each TINA administrative domain (also CORBA security domain and CORBA security policy domain) is also one security environment domain, i.e. how the security policy is enforced within the domain, is a local matter.

- Each boundary between TINA administrative domains is assumed to be also a boundary between security technology domains. This reflects that stakeholders with various kinds of CPE, varying priorities regarding security, and under possibly different national laws cannot be assumed to have the same security technology. It requires that the security technology used for interdomain interactions (including the mechanisms used) is negotiated as part of the respective cooperation security policy. This may result in the use of the same or a compatible security technology or the use of a mutually agreed security technology gateway.

Control of access to information and functionality at the service level (authorization) prevents illegitimate use of these resources within the service usage. It is based on the identities of stakeholders and authorization information regarding subscriptions and access to management functionality. The authorization process is implemented as a part of the service logic in the respective service components, e.g., the User Agent (UA) for the access service as well as the User Service Session Manager (USM) and the Service Session Manager (SSM) for the actual telecommunications service. The authorization decisions are made inside a service component after the invocation of an operation of an operational interface of the service component. These decisions rely on the authentication of the claimed identities as well as the authenticity, integrity, and (optionally) confidentiality of the messages exchanged with the other stakeholder. The three latter properties must be provided by DPE services, whereas the first, i.e. the authentication of the other stakeholder, takes place at the service level as part of the establishment of an access session between different stakeholders. CORBA security does not provide the necessary inter-ORB inter-domain authentication service. An additional authentication service must provide facilities for the mutual authentication of stakeholders. This service is used for the establishment of access sessions. Since user mobility must be supported, the authentication technology must be designed in such a way that the consumer does not need to trust the equipment in the consumer domain. This suggests the use of smart card technology. The integration of established smart card technology can guarantee that the owner (administrator) of the CPE (administrative domain) used at the moment cannot learn the secret authenticating information of the consumer.

DPE Security:
Session keys resulting from the execution of an authentication protocol at the service level may be used subsequently at the DPE (CORBA) level to prove and verify the identity of client objects. The session keys may also be used to provide authenticity, integrity, and confidentiality for the established security association using CORBA inter-ORB security. The exact security context of the security association, i.e. which security mechanisms are used, is established at association setup using the security tokens of the inter-ORB protocol. The context is derived from the cooperation security policy for the interaction between the domains, which also includes the choice of the mechanisms and the authentication servers (or public key certifiers).

Access control at the DPE level has to prevent illegitimate invocations of operations. This access control is, in contrast to the access control at the service level (authorization), not determined by the semantics of the service, but by the question whether the originating stakeholder of an attempt to invoke an operation is allowed to invoke this operation at all. The only possible results of an access decision are access permitted or access denied. Granularities are the whole interfaces or the single operation of an interface. In order to analyze how CORBA access control can be used, it is necessary to study how TINA service components are built up using CORBA. Service components are likely to be implemented as CO groups, even though the service architecture [23] does not prescribe the mapping of service components onto COs or CO groups. The service component's interfaces are in that case provided as contracts of the respective CO group. However, CORBA lacks the concept of object groups. Additionally, in contrast to TINA COs, CORBA objects have exactly one interface. We assume that each TINA CO is implemented as a set of CORBA objects and that each CO interface is implemented as a dedicated CORBA object, as proposed in [9]. The service component's interfaces (the contracts of the group) are, thus, provided as the CORBA interfaces of those CORBA objects that implement the CO interfaces that serve as contracts.

Potentially, the operations of all CORBA interfaces are accessible to everyone who has access to the kernel transport network that connects the single ORB systems. However, only few of these interfaces shall be accessible by objects acting for other stakeholders, i.e. shall be accessible for another identity than the one of the administrative domain (the owner) the object is allocated to. These interfaces are implementations of those interfaces of service components that are part of an interdomain reference point. Access to all other CORBA interfaces must be restricted to those objects that act under the same identity. This is easily achieved by ORB-local access control.

Access control for the objects that are indeed part of an interdomain reference point is more complex. However, our observation is that the functionality offered across domain boundaries is always structured as interfaces so that each instance of an operational interface is dedicated to exactly one other stakeholder, as suggested in [9]. As long as this observation holds, identity based access control can be applied at the granularity of service component interfaces, i.e. at the CORBA level to the whole object implementing the interface of the respective service component and can be realized by simply checking authenticity and integrity of the message conveying the invocation request. If the observation above does not hold, two cases with different granularities can be distinguished. If an interface of a service component is accessed by various stakeholders, but with the same rights, the unit of access control can also be the whole interface. On the other hand, if an interface is accessed by various stakeholders with different rights, the unit is the single operation of the interface. In both cases, access control should be supported by a list bound to each unit of access control that contains the identities of the stakeholders that are authorized to access the unit. This mechanism can be expected to be supported by a wide range of CORBA products.

Communications Contents Security:
In TINA, all service contents information is delivered by streams. (There is an ongoing discussion also to allow to provide contents information via operational interfaces without relying on streams. However, the following is also valid for such a delivery of contents.) Until now, CORBA does neither support streams nor stream protection. A DPE supporting streams must also support DPE security mechanisms applicable to streams. The establishment of a stream should include the establishment of a protection context for the stream. This context determines the security mechanisms and the key(s) used. The protection context is derived from the cooperation security policy of both domains. Such a policy, and possibly a session key, may already exist, e.g., if both parties are in a user provider relationship. If the parties have not authenticated each other directly (e.g., both parties are users of a common provider) and want to establish end-to-end security, they can use the authentication service mentioned above for direct mutual authentication and the negotiation of a session key.

5. Conclusion

We analyzed and structured the TINA security problem domain. It was demonstrated how the security services and mechanisms can be provided as part of the DPE functionality. Ongoing conceptual work in CrySTINA is dedicated to the identification and specification of the necessary single security services, a formal model of how administrative domains with different security policies agree on a cooperation security policy, and the use of a hierarchical public key infrastructure based on certificates as specified in [8]. The approach to TINA security presented is implemented extending the CORBA security features and using commercial CORBA products. Future work will cover the additional trust relationship between the current user of a CPE and the CPE itself that is needed for the support of personal mobility with full security.

References

[1] T.Eckardt, T.Magedanz, R.Popescu-Zeletin, M.Schulz, M.Stapf. Personal Communications Sup-

port in the TINA Service Architecture - A new TINA-C Auxiliary Project. *Proceedings TINA'96 Conference*, Heidelberg, Germany, September 1996, pp. 55-64.

[2] Internet RFC 1508. Generic Security Service - Applications Program Interface. September 1993.

[3] Internet RFCs 1825–1829, 1851, 1852. IPv4 and IPv6 Security. August/September 1995.

[4] Internet RFCs 1113, 1814, 1815. Privacy Enhancement for Internet Electronic Mail, Parts I-III. August 1989.

[5] ISO/IEC 7498-2: Information Technology - Open Systems Interconnections - Basic Reference Model - Part 2: Security Architecture (Also ITU-T Recommendation X.800).

[6] ISO/IEC 10164-7: Information Technology - Open Systems Interconnections - Systems Management: Security Alarm Reporting Function (Also ITU-T Recommendation X.736).

[7] ISO/IEC 10164-8: Information Technology - Open Systems Interconnections - Systems Management: Security Audit Trail Function (Also ITU-T Recommendation X.740).

[8] ISO/IEC 9594-8: Information Technology - Open Systems Interconnections - The Directory: Authentication Framework (Also ITU-T Recommendation X.509).

[9] B.Kitson. CORBA and TINA: The Architectural Relationships. *Proceedings TINA'95 Conference*, Melbourne, Australia, February 1995, pp. 371-386.

[10] Netscape. Secure Socket Layer. March 1996, http://home.netscape.com/eng/ssl3/

[11] C.Neuman, T.Ts'o. Kerberos: An Authentication Service for Computer Networks. *IEEE Communications Magazine*, September 1994, pp. 33-38.

[12] Object Management Group. The Common Object Request Broker, Architecture and Specification, Revision 2.0. July 1995, http://www.omg.org/corba/corbiiop.htm.

[13] Object Management Group. CORBA Security. December 1995, http://www.omg.org/library/corbserv.htm.

[14] Object Management Group. Common Secure Interoperability (CSI). March 1997, http://www.omg.org/library/schedule/Technology_Adoption.htm.

[15] Object Management Group. CORBAsecurity/SSL Interoperability. June 1997, http://www.omg.org/library/schedule/Technology_Adoption.htm.

[16] Open Software Foundation. *OSF DCE Application Guide*. Prentice-Hall, 1992.

[17] R.Rueppel. *Analysis and Design of Stream Ciphers*. Springer Verlag, 1986.

[18] B.Schneier. *Applied Cryptography*, 2nd edition. Wiley, 1996.

[19] S.Staamann. Overall Integrity of Service Control in TINA Networks. *Proceedings of Communications and Multimedia Security Conference '97*, Athens, Greece, September 1997.

[20] S.Staamann, U.Wilhelm. CORBA as the Core of the TINA-DPE: A View from the Security Perspective. *Proceedings of Object World Frankfurt 1997*, Special track *Distributed Object Computing in Telecommunications*, Frankfurt, Germany, October 1997.

[21] TINA-C Document: Authentication in TINA Access Session, Version 1.0 (Draft). Engineering Note, August 1996.

[22] TINA-C Document: Security Architecture, Version 2.0. Technical Report, March 1996.

[23] TINA-C Document: Service Architecture, Version 5.0. Baseline Document, June 1997.

[24] TINA-C Document: TINA Business Model and Reference Points, Version 4.0. Baseline Document, May 1997.

[25] H. Woo Sun, K. Eun Chul, J. Hee Kyung. Realization of TINA Service Architecture on the Internet. *Proceedings TINA'96 Conference*, Heidelberg, Germany, September 1996, pp. 235-243.

[26] P.R.Zimmermann. *PGP Source Code and Internals*. MIT Press, 1995.