in: Günter Müller, Kai Rannenberg (Ed.): Multilateral Security in Communications, Addison-Wesley-Longman 1999, 323-348.

Security Functions in Telecommunications – Placement & Achievable Security

Reiner Sailer, Hannes Federrath, Andreas Pfitzmann

Abstract

The placement of security functions determines which kinds of data they can protect against which kinds of attackers. Firstly, security functions are classified regarding their sphere of influence, e.g. between end points of communication associations (end-to-end) or end points of transmission lines (link-by-link). Next, we derive and illustrate restrictions and implications that limit the free choice concerning the placement of security functions. The general results are applied to interconnected telecommunication systems structured according to the OSI reference model. We investigate placement alternatives within user domains, network operators' domains, and service providers' domains and related implications on achievable security. We classify existing security solutions and describe the security that can be achieved by the respective solutions in order to promote the application of theoretical results.

1 Classification of security functions

The functionality of telecommunication networks can be classified according to the placement of co-operating functions in specific components and layers. There are two basic degrees of freedom concerning the placement of functions within telecommunication networks (see also Fig. 1):

- *horizontal degree of freedom*: choices for the placement of functions along different components
- *vertical degree of freedom*: choices for the placement of functions along different layers of single components

Security functions are functions that contribute to the security of a system, i.e. to the achievement of security goals. Voydock and Kent [8] differentiate between end-to-end (ete) and link-by-link (lbl) functionality and apply respective security functions as follows: lbl oriented security mechanisms offer security for information that is transmitted via an individual communication channel between two network nodes. The final source or destination of this information is not taken into account regarding lbl security.

Ete oriented security functions consider the network as a medium for the exchange of protocol data units (PDUs) in a secure way between source and destination nodes. Ete security functions protect PDUs during their transmission between end points of a telecommunication relationship.

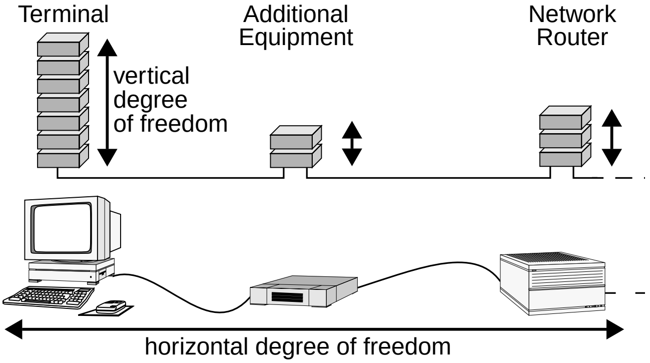

Figure 1: Degrees of freedom within user domains

In the following, we define different classes of functions. The classification proposed here is based on the relationships between entities that implement the respective distributed functionality. This classification will be applied to security functions and serves as a basis for the assessment of security functions and the related achievable security, cf. Fig. 2.

Functionality is called *link-by-link* (*lbl*), if and only if respective functions are placed within adjacent nodes of a telecommunication network, and if they relate to a single physical transmission link.

Functionality is called *point-to-point* (*ptp*), if and only if respective functions span several but not all links in the communication path between the end points. In doing so, the part of the network that connects the distributed functions is used as a transmission medium only (and does not contribute to the ptp-functionality under consideration).

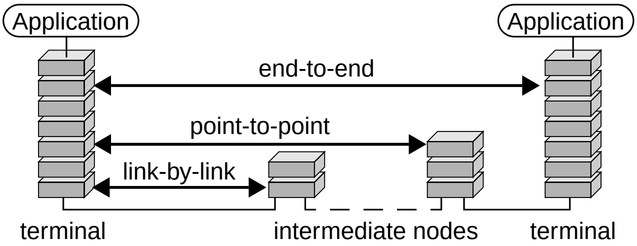

Figure 2: Classification of distributed functions

Functionality is called *end-to-end* (*ete*), if and only if related distributed functions are only located within end points of a communication. All communication nodes in between these end points serve as a transport medium only. End-to-end and link-by-link functionality, therefore, designate boundary cases. Applying these definitions to security functions, end-to-end security can be defined as follows:

End-to-end security means the implementation of security goals regarding communication events of two or more entities by end-to-end functionality. Point-to-point and link-by-link security are defined in a similar way. Fig. 2 illustrates the different kinds of relationships and respective classes of functions within telecommunication networks. It shows end-to-end, point-to-point, and link-by-link functionality distributed on different network nodes.

The transition from lbl to ptp functionality takes place when intermediate nodes are spanned by respective functionality, i.e. intermediate nodes are included in the sphere of influence of the co-operating functions under consideration. The figure also depicts ete relationships between terminals (end points). Intermediate nodes symbolize network nodes or additional equipment within the user domain (e.g. private branch exchanges, network termination, IP-routers, access routers, firewalls). The following examples illustrate the use of end-to-end, point-to-point, and link-by-link relationships with various security services.

Ete security: Examples of solutions for end-to-end security in networks are the Secure Socket Layer [10], Privacy Enhanced Mail [13], Pretty Good Privacy [14], or the Secure Shell [11]. The Secure Socket Layer (SSL), for example, offers authentication, data integrity, and data confidentiality security services implemented by co-operating security functions located at communication end points.

Ptp security can be implemented by peer network gateways as a protection against corruption of user data transmitted over the Internet between peer network gateways, e.g. by virtual private network routers [9]. Respective security functions span the whole network between the gateways, i.e. several links. The Point-to-Point protocol (PPP [15]) is another example. It provides access to the Internet via telephone lines. Included security functions, e.g. authentication or encryption, are ptp security mechanisms spanning local exchanges and other intermediate nodes between terminals and Internet service providers (ISPs).

Lbl security is implemented by the IEEE 802.10 secure data exchange (SDE) sublayer [1] that is proposed to enhance layer 2 within local area networks. Lbl security functions are often mainly proprietary because their usage is transparent for the higher layers.

2 Integrating security functions seamlessly

Emerging security requirements which have been ignored in the past will lead to changing security goals for future telecommunication services and related applications. Therefore, security cannot be implemented once and then forgotten. Security will be similar to a system state that has to be maintained during changing environmental conditions. This section introduces generic ways to include security mechanisms and related functions in telecommunication systems (network nodes, terminals, additional network infrastructure). It illustrates (i) choices for the placement of security functions, (ii) general boundary conditions for the implementation of security functions, and (iii) the impact of the placement of security functions on achievable security goals.

The following sections will apply these results to various practical examples. This helps readers to benefit from the outcome of our work and to achieve practical solutions that fit their security needs taking into account their individual environmental conditions.

2.1 General boundary conditions

Several boundary conditions have to be considered when integrating security functions into telecommunication systems. They result from requirements like:

- the transparency of security functions regarding the nodes between the enhanced nodes (saving investment in existing communications infrastructure),
- the security environments needed to protect the implementation of the respective security functions,
- the security goals that are to be achieved by respective security functions, and
- synchronization, operation, and management of security functions.

At first, security is not used as an end in itself but aims at adding new quality to applications. Therefore, the existing architectures on which these applications are based, need to be taken into account when deciding where to place security functions. This leads to some generic possibilities of where to place security functions. Secondly, to work correctly security functions need secure run-time environments. Such environments can be security modules [6], carefully administrated terminals, smart cards, or whatever is trusted by the users who rely on their respective security functions.

Thirdly, the enhanced system should satisfy the principles of layering according to the OSI reference model [4]. The most important principles are related to the abstractions which structure open interconnected telecommunication systems into layers according to their tasks regarding the exchange of application data. This leads to an ordered set of subsystems represented in a vertical sequence (layers 1 to 7, counted from the bottom). Entities in the same layer of different telecommunication systems are termed peer-entities. Actual communication only occurs between entity instances dedicated to the same layer. Peer entities use the services of lower layers to exchange data (see Fig. 3). Layer (N) entities use services offered by layer (N-1) and offer services to layer (N+1). Layer 7 directly offers services to applications. Retaining these principles unchanged (as a basis for open systems interconnection) implies that security functions should influence existing architectures as little as possible.

This practice saves huge investments in existing network infrastructure and results in broadly applicable security solutions. Furthermore, the existing comprehension of network protocols and their interaction is maintained. Boundary conditions for the inclusion of security functions result from transparency requirements regarding vertical and horizontal communications (cf. Fig. 3):

- The service access points (vertical interface) between existing layers should be maintained by introducing security sublayers. Newly implemented security sublayers must be kept transparent by maintaining existing primitives, parameters, and their interpretation at these interfaces. Thereby, additional primitives may be offered and new optional parameters may be introduced. Including security functions in single telecommunication systems is investigated in Section 2.2.
- Included security functions must not adversely affect the exchange of messages between peer entities, i.e. peer entity communications. Resulting boundary conditions are thoroughly explored in Section 2.3.

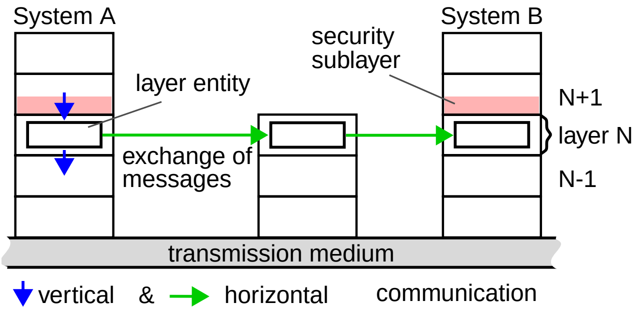

Figure 3: Vertical & horizontal transparency

Finally, applying security functions will not be free. At least they may adversely affect the achievable quality of service (e.g. delay, jitter). Attacker models are introduced in Section 2.4 to describe assumptions about attackers regarding specified security goals and to illustrate the achievable security goals considering the placement of security functions. Attacker models are a preliminary for tailor-made security and serve as a basis for the evaluation of existing security solutions in Section 5.

2.2 Transparency at service access points

Basically, security functionality can be added to existing network architectures and implementations by inserting transparent sublayers into existing protocol stacks of communication systems.

Enhancing network protocols means, that all changes regarding security functions are transparent to applications. Furthermore, different applications use the same security functions (economies of scale) if they use the same network protocols. The security functions will be dependent on the network protocols and sensitive data is not secured between the application and the security functions (including primary and probably secondary memory). Generally, a protocol data unit that is to be transported to a peer entity of layer (N) is delivered as a service data unit (SDU) to layer (N–1) using the respective services of the service access point.

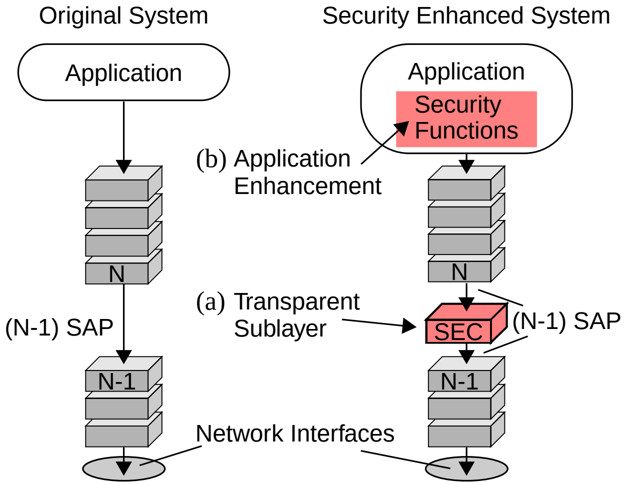

Figure 4: Generic system enhancements

Concerning security functions that are applied to service data units, there is a need for transparency regarding lower layers that reside within the sender communication system and regarding the higher layers that reside in receiver (or intermediate) communication systems.

The service interfaces between adjacent layers (N) and (N–1) must be maintained concerning both layers (Fig. 4a). A simplified example of a sublayer enhancement implementing security functions is represented by the Transport Layer Security Protocol (TLSP, [2]). According to Fig. 5, transport layer data is transformed by security functions within the TLSP sublayer. This procedure needs to be transparent for the underlying network layer of the sender stack. Therefore, security functions are applied to service data units only because only they are exchanged transparently by the underlying layers.

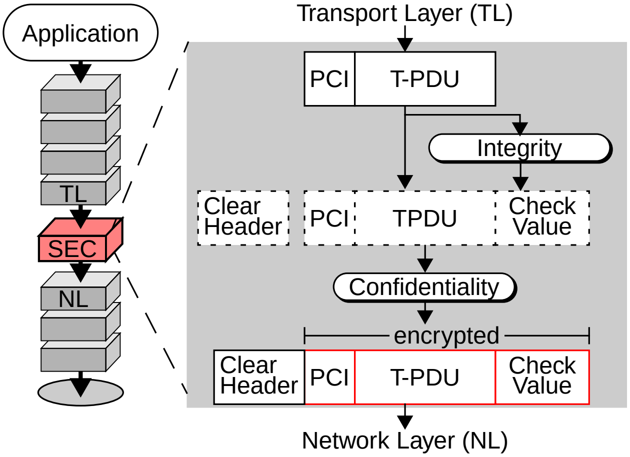

Figure 5: Exemplary security sublayer

Another approach deals with security functions included in applications, or close to applications (Fig. 4b). As a result, every application that uses sensitive data must be changed or enhanced. On one hand, this enables tailor-made security mechanisms for single applications. On the other hand, it is a huge effort to deal with every application (and future applications) separately. This approach, however, is independent from the network protocols and may be preferred in an environment of mobile applications.

Principally, included security functions transform data (i.e. encrypt, append message authentication codes) that is transparent to underlaying layers. Thereby, they protect data processed or generated by applications or by higher layers.

If existing communication systems (terminals, network nodes) are not to be changed – and, therefore, a transparent sublayer cannot be inserted into these terminals and network nodes –, a transparent sublayer can be logically included by using additional infrastructure. The additional infrastructure implements the layers below the intended sublayer and, therefore, neutralizes the use of these layers within the related communication subsystem. The security functions are applied to the resulting data (cf. Fig. 6).

Finally, the secured data is encoded in the same way as in the related communication system. From a security point of view, this approach is logically equivalent to the sublayer approach of Fig. 4a if and only if there are no successful attacks within the highlighted domain.

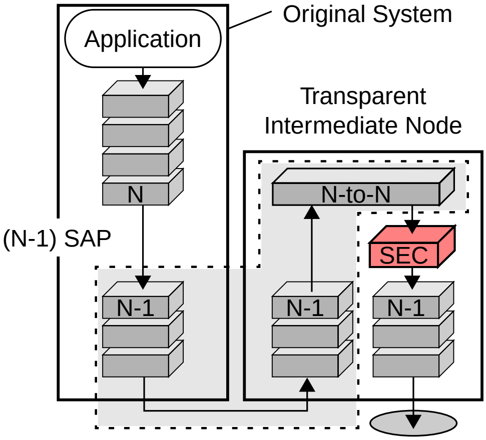

Figure 6: Additional security infrastructure

In addition to the added security functions, all the layers 1 to N-1 have to be implemented within the additional infrastructure. The N-to-N functions just translate indications (confirmations) on the incoming link into requests (responses) at the outgoing stack when forwarding the data. Quality of service (qos) parameters may be changed by introducing additional (functionally transparent) infrastructure, because the layer (N-1) to layer (1) functions are applied three times. This may affect delay or jitter or other quality of service parameters adversely. Therefore, adding new infrastructure will mainly be efficient, if the security functions are added at lower layers.

2.3 Transparency regarding the exchange of PDUs

This section discusses limits arising from horizontal transparency requirements (cf. Fig. 3) in order to save investments in existing infrastructure and to retain the layering principles of the OSI architecture. This demands for every protection function (e.g. encryption) below an entity on the sender side an inverse function (e.g. decryption) below the peer entity of the receiving side in order to comply with above transparency requirements. Otherwise, an intermediate system might have to operate on encrypted data without having access to related keys. The resulting boundary conditions ensure that peer entities above and below the integrated security functions are not adversely affected by the security enhancements. Thereby, we enhance the view from single communication systems (cf. Fig. 4) to multiple communicating systems. Fig. 2 shows a basic model for the placement of co-operating security functions within telecommunication systems.

Subsequently, boundaries for link-by-link, point-to-point, and end-to-end security functions are derived from horizontal transparency requirements concerning intermediate nodes. We will discuss security functions distributed over two communication systems (see Fig. 7).

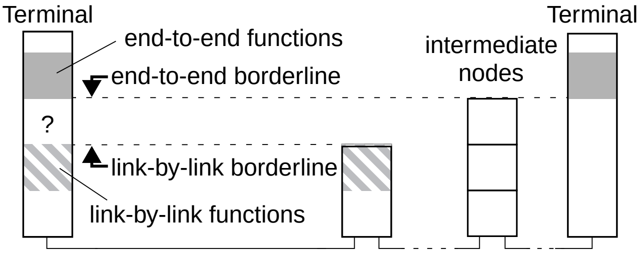

Figure 7: End-to-end and link-by-link borderlines

The related definitions and conclusions are easily adapted to security functions distributed over more than two communication systems (e.g. secure multicast service functions). The *end-to-end borderline* describes a *lower vertical limit* for the implementation of end-to-end security functions. The end-to-end borderline is defined by the lowest layer, that is not used in any intermediate system within the communication path between the end points under consideration. This ensures that no intermediate system (i) tries to process (ete) encrypted data, (ii) tries to change (ete) integrity protected data, or (iii) fails because of the unexpected addition of synchronization data exchanged between (ete) security functions.

Consequently, ete security functions residing above the ete borderline do not interfere with functions of intermediate nodes. E.g. routing information for data cannot be encrypted within terminals, because all intermediate network nodes need to evaluate related information to route the data. Concerning packet switched networks, the end-to-end borderline is located between layer 3 and 4.

The *link-by-link borderline* marks an *upper vertical limit* for the implementation of link-by-link security functions. It relates to a single physical communication link and resides on top of the highest layer that is implemented within all adjacent systems of the communication link. All adjacent communication systems must implement inverse lbl security transformations (e.g. encryption, decryption) at a single (sub)layer. Implementing lbl security functions within different layers of a single communication system would imply multiplying security management. Moreover, the security management and operation would be dependent on the routing of messages. This is not desirable due to related costs. This argumentation regarding lbl security functions is applied to the whole network domain. In conclusion, lbl security functions will almost always be implemented within a single layer throughout a network domain. This is the reason for the definition of a link-by-link borderline.

2.4 Attacker models – defendable attacks

Attacker models describe assumed attackers and their resources and facilities. They are the foundation for a security evaluation of the system under consideration. In our example, the system comprises two user domains (information source, information sink) that are connected via telecommunication networks.

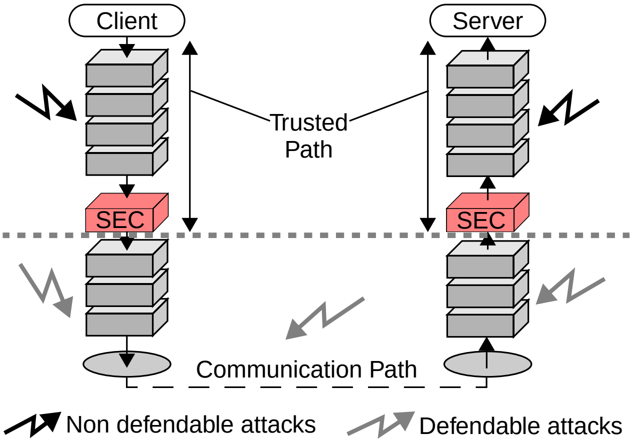

Figure 8: Attacker model – achievable security

Fig. 8 illustrates various points of attack. User data, produced by applications within the source, serve as the sensitive objects. Protocol data (control data) is added to application data during protocol processing by layer entities. Moreover, sensitive information can be deduced by observing the use of communication services or the mere existence of communication events through traffic analysis at transmission links.

Assume an attacker that has manipulative access to points between the source of sensitive data and related security functions. These attacks cannot be countered because the security functions have not yet been applied to related sensitive data. Therefore, a trusted path must exist between the source of sensitive data and related security functions. A trusted path means a path in which the security of sensitive data is inherently guaranteed. The same holds for the path between security functions within the receiving user domain and the destination of the sensitive data. If an attacker has manipulative access to security functions, these security functions are useless against this attacker. The reason is, that attackers would be able to manipulate respective security functions unnoticed by receivers (see [5] for examples of so called trojan horses).

Data and related information that are produced or processed above the security functions are protected against attacks between the security functions and the network interface (e.g. eavesdropping if encryption is applied as security function). Attacks within the switching network (subscriber lines, transmission lines, exchanges) and at the receiver side between the network interface and the security functionality are equally protected. Protocol control information that is added by layer entities below the security functions (e.g. addresses, service indicators) is not protected against attackers and can be used to deduce further information like communication behaviour etc. of users.

A *trusted domain* comprises systems or parts of systems (e.g. mobile terminal, security module) that are trusted by the respective users. Trusted domains implicitly place trust in the expected behaviour and correct implementation of software and hardware components. No successful attackers (incl. no trojan horses [5]) are assumed within a trusted domain. A trusted domain is always related to a single user or group of users. These users have to decide, whether remaining threats – and the related costs for attackers to exploit them – are tolerable or whether additional security measures have to be taken.

In order to be able to evaluate security measures, it is helpful to understand related attacks that exploit threats. Therefore, Fig. 9. illustrates the general practice of an attacker to enforce access to data encapsulated in transmitted messages. In this case, the transmitted data is protected by the Secure Socket Layer functions. At first, an attacker has to gain access to a transmission medium (if there is no access within the end systems themselves). This access must enable an attacker to receive messages that are sent by an end system. Secondly, the eavesdropped signals have to be decoded (channel decoding). Thereby, an attacker gets access to layer 2 frames (see the lower header H and trailer T in Fig. 9). Thirdly, decoding the layer 2 frames, an attacker can read the layer 3 protocol control information (the middle H in Fig. 9, comprising sender and receiver network addresses, logical channel identifiers, etc.) and the layer 3 protocol data units. Fourthly, this practice is continued with the TCP data encapsulated in IP packets.

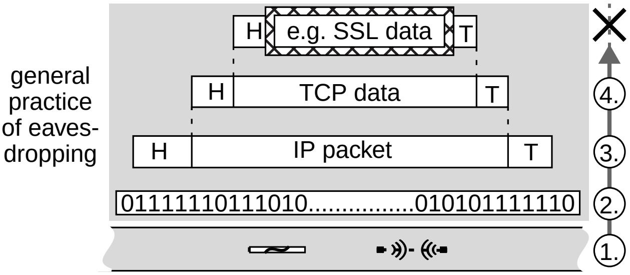

Figure 9: Eavesdropping at transmission links

In our SSL example, an attacker will not be able to decode the application data protected by the SSL functions if (s)he does not know the appropriate keys or how to break the security functions used within the SSL implementation. Therefore, all data that is produced above the SSL is protected against eavesdropping at lower layers or transmission links. Radio links are particularly vulnerable to eavesdropping and should, therefore, be protected at least by link-by-link encryption. Eavesdropping is not restricted to transmission lines. Senders and receivers produce electromagnetic emissions during the processing of data that is to be transmitted. The emission of transceivers can be used for eavesdropping far away from the emission source.

In a similar way, encapsulated data may be manipulated and written back to the transmission media. Effective security measures break this chain and – within the limits stated in the attacker model – prevent potential attackers from getting access to sensitive data (e.g. hindering decoding or access to transmission media) or from changing or inserting data carrying sensitive information without this being noticed. Depending on the communication protocols, changing application data may be achieved solely by inserting messages into the communication path.

Another attacker model may be used, if information leakage needs to be considered and it is assumed that there are trojan horses within the end points above end-to-end encryption functions [22].

2.5 Security gaps – sources of non defendable attacks

Security gaps are areas or a set of layers where no protection is provided by ete, ptp or lbl functions. In Fig. 10, protocol information exchanged by instances of terminal A and B that reside below the ete and above the lbl borderline are not protected against unauthorized access by instances within intermediate systems C and D. Basically, there cannot be any lbl or ete functions between the ete and lbl borderlines (cf. definitions).

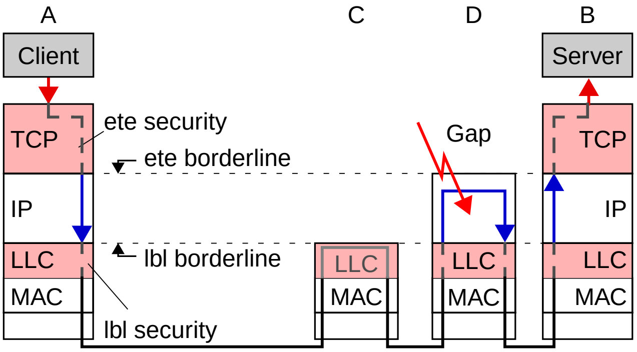

Figure 10: Security gap – in between ete & lbl functions

It is reasonable to define distinctive ete and lbl borderlines respectively for encryption (data not processible), integrity (data not unnoticedly changeable), and availability mechanisms because the impact of intermediate nodes on ete and lbl borderlines may differ. Therefore, the routing of messages or connections affects the borderlines for ete and lbl security. Additional routing information concerning intermediate nodes (apart from destination addresses) enables avoiding intermediate nodes that implement high protocol stacks and would, therefore, restrict ete security by moving the ete borderline upwards.

Gateways at network boundaries may, for example, shift the ete borderline upwards, particularly if these gateways translate between different codes (e.g. some e-mail gateways). In conclusion, it may be advantageous to enable users to control the routing of calls (routing control). The following example depicts a security problem arising from this gap and its solution by combining ptp security.

Gap example: anonymous service access

Specialized information services are becoming more and more convenient in public switched networks and in the Internet. The mere access of such services creates information about personal interests. This information can be combined with the network address of the originating communication system to derive the interests of particular users. Therefore, the demand for anonymous service access is likely to rise with the increased use of such services. On one hand, network addresses have to be known at the end points of a communication association to exchange data between services and users; they cannot be omitted with current service implementations.

On the other hand, network addresses cannot be encrypted when passing the network because they are used by intermediate nodes to route user data between users and services. As a consequence, network addresses of layer 3 cannot be protected end-to-end. In an environment where user identities can be derived from network addresses, anonymity (at least towards network operators) is not possible by using end-to-end security only. Using dedicated proxies to access such services represents a feasible solution (cf. Fig. 11).

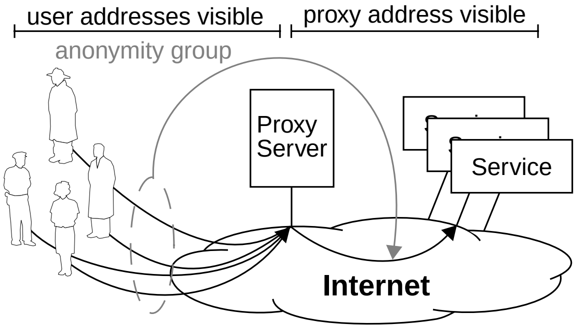

Figure 11: Bridging of gaps by combined ptp security

In doing so, many users form an anonymity group. The visible association is located between the proxy and the service providers. Only the proxy can deduce the originator within the anonymity group regarding a single service access. Protection against attackers that observe the proxy and try to relate ingoing proxy accesses and outgoing service accesses is addressed by special mechanisms employed between users and proxies (cf. [17]). Here, the gap concerning ete security in the network layer is eliminated by employing (i) ptp security between users and proxies to establish anonymity groups and (ii) ptp security between proxies and Internet information services for access control, billing, or protection of the service contents.

If anonymity towards the proxy is needed, chains of proxies (Mixes, see [20]) can be employed instead. In this case, anonymity is not guaranteed if either all proxies work together (against the user) or the anonymity group is reduced to one user by selective availability attacks at the user side of the first proxy. Anonymity is preserved in higher layers by the use of additional security functions applied to higher layer addressing and data exchanged between the user and the service node. [21] describes an implementation for anonymous internet service access.

3 End-to-end versus link-by-link security

User requirements on communication services are not restricted to quality of service aspects but also include confidentiality and integrity of user data and availability of services. Confidentiality and integrity requirements related to user data can be implemented by end-to-end, point-to-point, or link-by-link functions. Availability can not be guaranteed by end-to-end mechanisms alone. The following paragraphs discuss advantages and disadvantages of ete, ptp, and lbl approaches regarding user requirements on security.

3.1 Arguments for end-to-end security

Saltzer, Reed, and Clark [7] recommend placing as many of the functions as possible at the end points of a communication association (e.g. within terminals). They argue for end-to-end functions as close to the user (application) as possible. They discourage application supporting functions in lower layers, because these functions only have an effect up to the layer they are implemented in. Using lower layer functions will imply trust on intermediate systems concerning the correct implementation of layer functions. As their example (cf. Fig. 12) shows, this may cause severe security leaks [7].

At the Massachusetts Institute of Technology, a network system, involving several local networks connected by gateways, used packet checksums on each hop to protect against corruption of bits during transmission. The implicit assumption of programmers – that data corruption will occur only at transmission lines – has led to the conclusion that this offers a reliable transfer of data. One gateway, however, did not stand up to the assumption. It developed transient errors while copying data from an input to an output buffer. After many source files of an operating system had been transferred through the defective gateway, some of these source files had been manipulated and had to be corrected by comparison with and correction from old listings.

Translated to the security area, the checksum would correspond to Message Authentication Codes (MACs), used to protect packets from manipulation during transmission. This is shown in Fig. 12. Even if the MAC functions work correctly these packets are not protected above these MAC functions. Therefore, the trusted path in Fig. 12 comprises all functions below the source and above the MAC functions within all communication systems in the data path. A gap may, therefore, be defined as the part of a (distributed) trusted path that is located outside the trusted environment, e.g. within intermediate nodes. This gap is eliminated (respective corruption is compensated) by using ete integrity checks within source and destination, e.g. at the TCP layer as depicted in Fig. 10.

Moreover, the implementation of ete functions reduces the complexity of the lower layers. This raises the performance of the overall communication systems, because only necessary functions are applied and because they are applied only once (with the exception of lbl and ptp transmission error detection and correction). Security specific advantages of the ete approach to security enhancements are: (i) security functions can be implemented, configured, managed, and controlled by the users themselves, (ii) ete security is independent of intermediate systems (except availability), and (iii) no changes of intermediate systems are required.

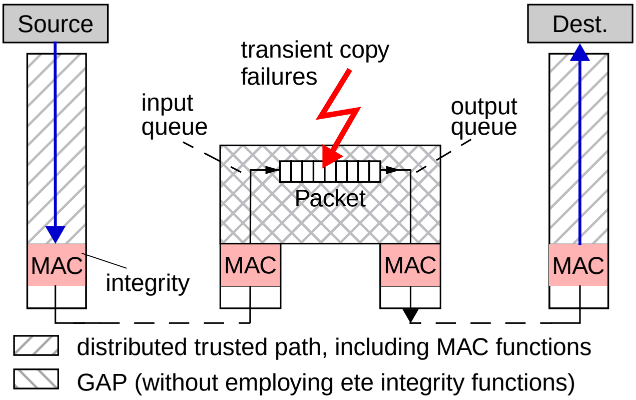
Source
Dest.
transient copy failures
input queue
output queue
Packet
MAC
integrity
MAC
MAC
MAC
distributed trusted path, including MAC functions
GAP (without employing ete integrity functions)

Figure 12: End-to-end arguments – an example

Of course, application specific (or at least terminal specific) security functions push up the cost for enhancements. They might reduce compatibility and interoperability of applications or terminals. Furthermore, user controlled security functions may be restricted for political reasons (e.g. the use of strong cryptography within Russia or export from the United States).

3.2 Arguments for link-by-link and point-to-point security

Exclusively using ete security functions does not contribute to a general basic protection against outsider attacks. Handling errors or terminal errors would severely affect the overall security achieved. This is not acceptable to large companies. General basic protection is best implemented by terminal and application independent security mechanisms. Handling errors, application updates, or the use of mobile equipment do not affect these basic protection mechanisms. Hence, lbl and ptp security represent a permanent value in an environment of daily changing applications and users. Because lbl or ptp functions offer services to all layers above, a single cryptographic enhancement may supply different applications with encryption facilities. Lower layer security functions can be used to protect protocol data of higher layers. E.g. address information can be concealed at transmission links if it is encrypted by layer 2 security functions, cf. [1]. Placing, for example, integrity functions solely at application level would force corrupted packets to be retransmitted over the hole communication path. Detecting corruption within lower layers enables more efficient retransmission over one link. Moreover, the corruption probability may rise considerably when moving from lbl to ete error detection. Eventually, lbl and ptp security mechanisms are more fault-tolerant regarding local failures. Rerouting or alternative path mechanisms easily bridge the failure of a single link without affecting the superior communication association. However, lbl mechanisms require trust in intermediate nodes that are not supervised or controlled by the user.

3.3 Combinations

The discussion above motivates a combined lbl, ptp, and ete approach to security solutions. Reasonable approaches have to balance requirements concerning cost, security goals and related assumptions (attacker models), and technically feasible security enhancements:

- Lbl security functions are useful for general basic protection of user and service data in lower layers and may protect against traffic analysis attacks. Lbl security mainly addresses security threats arising from attackers at transmission lines, e.g. eavesdropping at radio links.
- Ete security functions are preferable for application data transmitted transparently over insecure networks. Ete functions are controlled directly by the users and are independent of network or service providers.
- Ptp security functions (e.g. certificate retrieval services, shared authentication services, or virtual private network services) may be applied (i) to reach the economies of scale needed to make such services profitable and secure, (ii) to ensure high availability and centralized management, or (iii) to take advantage of the compatibility offered by network services.

In the following section some placement strategies for benefiting from the remaining horizontal degree of freedom based on the kind of security functions used and whether implemented as lbl, ptp, or ete functions, are presented.

4 Placement strategies

This section suggests specific placement strategies for security functions tailored for efficiency or effectiveness of respective security solutions. These strategies are derived from the above discussion about ete, ptp, or lbl security functions.

Boundary conditions according to horizontal transparency mainly apply to cooperating security functions.

Single allocated security functions enforce specific measures to guarantee the horizontal transparency requirements (and resulting borderlines). Examples are filtering functions that may reside within terminals in any layer, as long as the filtering information is still accessible. Filtering of TCP or IP traffic is simple, because TCP and IP protocols know how to manage missing answers (e.g. conditioned by filtering initial requests at the receiving side). This does not necessarily apply to any protocols that might be filtered. On the application layer, where reliable transfer is generally assumed, discarding service requests by filtering may imply the need for the insertion of service reject information messages to lead the initiating entity into a reasonable state.

Throughout this section, placement strategies are illustrated in an exemplary environment for IP based networks. Our exemplary environment consists of a user domain including the following components: user terminals, a firewall system (shared by all terminals within the domain), and a network router connecting the user domain and the Internet. Applying the principles of Section 2 leads to the following boundary conditions regarding the horizontal placement of security functions:

- The *ete borderline* resides at the upper limit of layer 3, because IP-routing functions are applied within the network routers.
- The *lbl borderline* usually resides at the upper limit of layer 2, if bridges etc. are used to build up switching groups within subnetworks in the user domain.
- *Ptp functions* are mainly applied within applications or within layer 3, e.g. for certificate retrieval or for bridging insecure Internet domains via VPN routers.

Applying a balanced mixture of both strategies promotes solutions that are effective and efficient and that can be evaluated against specified security goals.

4.1 Effectiveness: focus on end points

A security solution is effective, if it supports security services that are (i) *tailor-made*, i.e. fit individual needs, (ii) *multilaterally secure*, i.e. balance the security goals of different parties in a communication relationship, and (iii) *trustworthy* for the parties relying on them. In order to achieve these goals, we propose to shift security functions as close to the end points – and within the end points as close to the application or the user – as possible. This approach is illustrated in Fig. 13. The set of security functions to be placed are shown on the left. This set is derived from the security goals, the type of function (ete, ptp, lbl) is derived from the attacker model and from boundary conditions stated in Section 3. The figure depicts the placement of ete security functions within terminals.

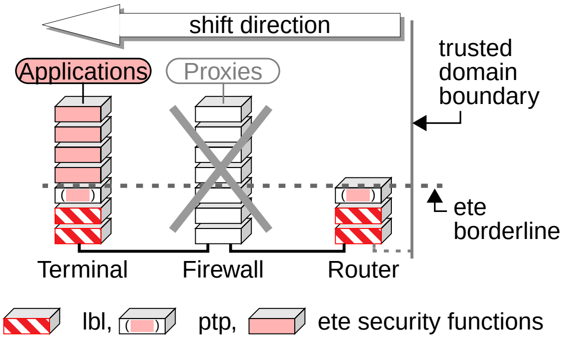

Figure 13: Effectiveness – shift towards end points

Security functions are shifted in the direction of the terminal that denotes an end point of communication relationships. Lbl functions are shifted to the terminal. Consequently, the surrounding communication infrastructure (routers, etc.) needs to implement inverse lbl security functions. Ptp functions are applied within the terminals, within the router, and within other components that implement functions within the security gap (between ete and lbl borderlines). In our case, the security gap is represented by the IP layer. Bridges or hubs in between the terminal and the router are not affected by ptp functions. Because of this end point focused approach, no firewall or additional infrastructure is needed. This approach guarantees a minimal trusted path because security functions are close to the generation or processing of sensitive data.

As security functions are implemented close to the application, they may serve specific application needs. This promotes tailor-made security. Placing security functions close to the user ensures maximum control of security functions by the user. Users can, therefore, be included in resolving the conflicting security requirements of different participants of a service. Furthermore, security functions can be included in security modules owned and, therefore, trusted by their respective users. It is assumed that there are secure environments within the respective components to implement security functions in an effective way.

4.2 Efficiency: focus on domain boundaries

A security solution is efficient, if it promotes security services that are (i) *scalable*, i.e. adaptable to qos requirements and a changing number of users, (ii) *shared* by different users or applications, i.e. generating economies of scale (multiplex-gain), (iii) *easy to integrate*, and (iv) *compatible* with existing and future solutions. The efficiency placement strategy approximates these requirements by shifting security functions as close to domain boundaries as possible. The user domain shown in Fig. 14 includes a network router at the domain boundary. A firewall serves many terminals and, therefore, is closer to the domain boundary than the terminals themselves. First, the network router is enhanced by lbl and ptp security functions. The counterparts of lbl security functions are installed within the network; the peer ptp security functions are placed within any component of peer user domains. Ete and ptp security functions that can not be included in the router are included in a firewall. If there are ete security functions enhancing applications, this leads to application level firewalls.

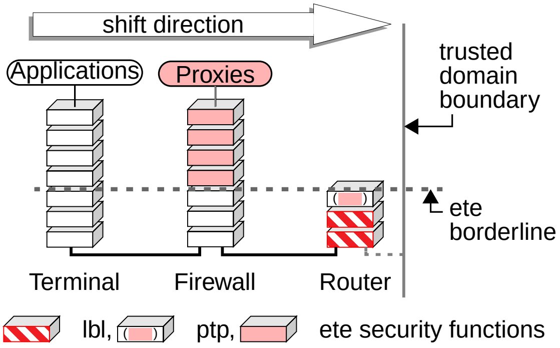

Figure 14: Efficiency – shift towards domain boundary

Ete security functions that can not be included in a firewall (e.g. because there are no proxies of the respective application available) are included in the terminals. This efficiency approach guarantees a high multiplex-gain by enhancing shared components that may serve multiple terminals. Resulting solutions are scalable by including additional firewalls or routers. By keeping the terminals unchanged, this approach promotes easy to integrate security functions (black-box enhancement), allows centralized maintenance of security functions, and will show high compatibility.

4.3 Balancing effectiveness and efficiency

The placement strategies presented represent the boundary cases of conceivable strategies for solving genuine security tasks. The following table shows the advantages and drawbacks of the strategies above:

close to users	*close to domain boundary*
+ network independent + tailor-made security + security options user controlled	+ multiplex gain + end systems unchanged + central administration
- no multiplex gain - change of end systems	- network dependent - higher layer security costly - long trusted path - single point of trust

Table 1: Comparison of the competing approaches

A balanced approach will take into account the requirements given in a local application area. We give some hints about when to prefer which strategy:

- *Common security functions* are placed close to domain boundaries. They reach a high multiplex-gain, offering the possibility for centralized maintenance.
- *Specialized or optional security functions* are implemented close to the user. This promotes tailor-made security and the inclusion of users during the negotiation of security services. Thus, specialized security functions do not affect (e.g. concerning the achievable qos) other applications or other users.

Therefore, solving a placement task implies first deciding which security goals are to be implemented and whether efficiency or effectiveness is preferred. Afterwards, the security functions implementing the security goals are placed according to the respective strategy.

4.4 Further boundary conditions

Further requirements need to be considered when deciding about the placement of security functions. Some important conditions, affecting the practical use and feasibility of placement strategies include:

- existing *trusted domains* (e.g. smart cards [3] or security modules [6]) in multi user environments,
- financial, organizational, and operational *expense* for implementation and maintenance (e.g. adaptation to longer keys, new security algorithms),
- *transparency* or *control requirements* regarding applications or users,
- *security management* (e.g. key generation, distribution, and change),
- *negotiation* of optional security services,
- *synchronization* of co-operating security functions, and
- *interoperability* and *compatibility* requirements.

The effect of these additional criteria on the efficiency and effectiveness of security functions depends on the respective application case. Technology dependent boundary conditions (e.g. achievable encryption rates) and resulting dependencies on products or companies make general statements about the optimal placement impossible. Rather, the examples above aim at enabling the readers to find a placement for security functions that satisfies the security goals and the boundary conditions which apply to their specific environment.

5 Exemplary evaluation of security solutions

This section discusses some general types of security solutions considering the tools presented in the preceding sections. The section is structured according to the different players in telecommunication: users, service providers, and network operators. Manufacturers would have to be included, if the development process were discussed. Choosing different security solutions, the maximum achievable security is derived, assuming correct implementation and management. This means that we do a best case evaluation! Trusted paths (assumed attacker-free zone) and the efficiency and effectiveness of respective solutions are explicitly discussed. The readers will not find comparisons of products. This would require a closer look at specific application cases which is out of the scope of this contribution.

5.1 Security solutions for user domains

User domains are the source of primary sensitive data, whereas network domains are secondary domains, serving for the exchange of data between peer user domains or user domains and service provider domains. Therefore, most ete security will reside within user domains. This is reflected in the vast variety of available ete security solutions. We will discuss only a few to illustrate the general practise of evaluation.

Pretty Good Privacy (PGP [14]), Privacy Enhanced Mail (PEM [13]), Secure MIME (S/MIME [12]), and Secure Shell (SSH [11]) include security functions in applications and, therefore, minimize the trusted path. On the other hand, respective security functions are mainly restricted and tailored to specific applications (E-Mail, Remsh, Rlogin, etc.), with the exception of S/MIME that can be used by any transport mechanism that transports MIME data, like the Hypertext Transfer Protocol (HTTP). The Secure Socket Layer and the Transport Layer Security Protocol (cf. Fig. 5) include ete security functions into the transport layer functions of end systems. In doing so, the trusted path is restricted to end systems which are under the control of the communicating parties.

Firewalls [16] include access control functions in the shared components at the user domain boundary. They serve for access control regarding services within the user domain. Virtual Private Network (VPN) routers implementing IPsec [9] security functions are used to bridge insecure (sub)networks. The path between the applications and the VPN-Routers needs to be trusted or secured by additional security means. Firewalls and VPN security functions are often integrated into a single component. Line Encryptors are used to bridge insecure areas within the user domains, e.g. between terminals and shared VPN security functions. They improve security at dedicated transmission links.

In user domains, a combination of application tailored ete and shared ptp security functions is advised (cf. discussion in Section 3).

5.2 Security solutions for network operator domains

Security enhancements for network operator domains mainly address access control regarding users and service providers, network integrity, and fraud control (misuse of services).

Example: IP-based networks. It is not efficient to screen and filter all data entering or leaving inband signalling networks (e.g. IP-based networks) because all data crossing network boundaries would have to be inspected. There is no separation of control and user data, hence the huge mass of data would imply filtering at very high speeds. Here, distributed security functions for access control should be included within the network. Basic filtering may be done at the boundaries, whereas more sophisticated and time consuming access control functions (including authentication, encryption) should be shifted towards respective network servers (end points).

Conclusion: From a user's point of view, the network should *employ* link-by-link security functions. It should *support* point-to-point and end-to-end security functions by allowing the exchange of security control data. Point-to-point availability is to be provided by the network operator, e.g. by enabling user controlled routing (IP-based networks).

5.3 Security solutions for service provider domains

Service providers offer services to many users, aiming at large economies of scale. They use the services of network operators to enable users to access their services from any place. Well-known shared services are certificate retrieval services (Public Key Infrastructure [19]) or ticket retrieval services (c.f. Kerberos [18]). Related services are key generation or distribution services. The economies of scale thereby enable the use of leading edge infrastructure offering maximum security.

MIX services that conceal the routing of a call are ptp services (c.f. [17]), because they always span only part of the communication relationship and must be placed in between the end points. Key generation and distribution services can be independent of other communication events and define their own communication event. Consequently they may represent ete or ptp services.

From a user's perspective, service providers should *offer* end-to-end and point-to-point services to enable security functions independent of network operators. In the Internet domain, this is usual; in public switched networks, the security offered by service providers often depends on security mechanisms of network operators. An example is the authentication via multi-frequency signals that may be intercepted by network operators, at transmission lines, or even within the user terminals.

Using ptp or ete security functions excludes the network operator's domain from the trusted path. The service providers are almost always within the trusted path. Therefore, a service provider offering security services or services that process sensitive data has to be trusted by the users.

6 Conclusion and outlook

The proposed classification of co-operating security functions in telecommunications in terms of end-to-end, point-to-point, and link-by-link functions emphasizes the sphere of influence of the respective security functions.

Generally, security functions are not transparent for applications or network functions that process protected data. The end-to-end and link-by-link borderlines which have been identified, therefore, depict areas, outside of which the transparency of security functions cannot be ensured. Borderlines refer not only to layer abstractions, but also to the data to be protected. Keeping to the limits set by respective borderlines, intermediate components are not adversely affected by added security functions. Consequently, no adaptations are necessary within intermediate systems adhering to the OSI layering principles.

The other approach to protecting sensitive data by added security functions is to avoid creating sensitive data at all. This demands removing or changing existing functions which process the respective sensitive data. This is obviously the more secure approach, because there is no need for trust in security functions and trusted paths. The enhancement approach is suitable for sensitive data (e.g. identification information) that is essential and cannot be removed from telecommunication systems, e.g. because it is needed for access control.

Often, a balanced approach is possible. On one hand, existing sensitive data is minimized (e.g. substituting a pseudonym for a real name). On the other hand, new security functions (e.g. for communicating with third parties taking the responsibility for these pseudonyms) are needed to retain the functionality of telecommunication services when minimizing sensitive data (e.g. to implement access control functions without disclosing the real identity of users). Redundant security functions, e.g. ete and lbl security functions which aim at the same overall security goals, offer complementary protection in case of failures in end systems (e.g. application or handling errors which make ete security functions useless) or network components.

7 Acknowledgements

The authors thank Anja Jerichow and Dogan Kesdogan for their assistance and participation in discussions regarding a former German version of this contribution.

8 References

[1] IEEE 802.10: Interoperable LAN / MAN Security. IEEE Standards for Local and Metropolitan Area Networks, 1992.

[2] ISO / IEC 10736: Transport Layer Security Protocol. 1995.

[3] M. Hendry: Smart Card Security and Applications. Artech House, 1997.

[4] ISO 7498 International Standardization Organisation: Basic Reference Model for Open Systems Interconnection. 1989.

[5] Y. Lapid, N. Ahituv, S. Neumann: Approaches to Handling „Trojan Horse“ Threats. Computers & Security, 5, North-Holland, 1986, pp. 251-256.

[6] A. Pfitzmann, B. Pfitzmann, M. Schunter, M. Waidner: Trusting Mobile User Devices and Security Modules, IEEE Computer, Vol. 30, No. 2, February 1997, pp. 61-68.

[7] J. Saltzer, D. Reed, D. Clark: End-To-End Arguments in System Design. ACM Transactions on Computer Systems, Vol. 2, No. 4, November 1984, pp. 277 - 288.

[8] V. L. Voydock, S. T. Kent: Security Mechanisms in High-Level Network Protocols. Computing Surveys, Vol. 15, No. 2, June 1983, pp. 135-171.

[9] E. Roberts: X-Tranet Routers – The security and savings are out there. Data Communications, September 1998, pp. 75 - 83.

[10] The SSL Protocol – Version 3.0. Netscape Communications Corporation, 1996. http: / / home.netscape.com / eng / ssl3.

[11] Secure Shell: by http: / / www.ssh.fi / sshprotocols2 / index.html.

[12] S. Dusse et. al.: RFC 2311,2312: S / MIME Version 2 Message Specification, 1998.

[13] J. Linn, S. Kent, D. Balenson, B. Kaliski: RFCs 1421-1424 Privacy Enhancement for Internet Electronic Mail: Parts I-IV, 1993.

[14] D. Atkins, W. Stallings, P. Zimmermann: RFC 1991 PGP Message Exchange Formats, 1996.

[15] RFC 1661: The Point-to-Point Protocol (PPP). W. Simpson, Editor. July 1994.

[16] W. Cheswick, S. Bellovin: Firewalls and Internet Security, Addison-Wesley Longman, 1994.

[17] A. Jerichow, J. Müller, A. Pfitzmann, B. Pfitzmann, M. Waidner: Real-Time Mixes: A Bandwidth-Efficient Anonymity Protocol. IEEE JSAC, Vol. 16, No. 4, May 1998.

[18] B. Neuman, T. Ts‘o: Kerberos – An Authentication Service for Computer Networks. IEEE Communications Magazine, Vol. 32, No. 9, 1994.

[19] W. Burr, D. Dodson, N. Nazario, W. Polk: Minimum Interoperability Specification for PKI – Components, Version 1. NIST Special Publication 800-15, 1998.

[20] D. Chaum: Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms. Communications of the ACM 24/2 (1981).

[21] H. Federrath, K. Martius: Multilaterally secure WEB-Access. http://www.inf.tu-dresden.de/~hf2/cebit98/index_e.html.

[22] P. A. Karger: Non-Discretionary Access Control for Decentralized Computing Systems. Massachusetts Institute of Technology, LCS, TR-179, 1977.