Pp. 107-118 in: Proceedings of the 16th Annual Computer Security Applications Conference; December 11-15, 2000, New Orleans, Louisiana; IEEE CS Press; ISBN-0-7695-0859-6

Protection Profiles for Remailer Mixes
Do the New Evaluation Criteria Help?

Kai Rannenberg
Microsoft Research, Cambridge, UK
kair@microsoft.com

Giovanni Iachello,
Institut für Informatik und Gesellschaft
Freiburg University, Germany
g.iachello@iol.it

Abstract

Early IT security evaluation criteria like the TCSEC and the ITSEC suffered much criticism for their lack of coverage of privacy-related requirements. Recent evaluation criteria, like the CC and the ISO-ECITS now contain components assigned to privacy. This is a step towards enhanced privacy protection, especially for non-experts. We examined the suitability and use of these components and the criteria as a whole by specifying a number of Protection Profiles (PPs) for remailer mix networks, as mix networks aim at user anonymity and unobservable message transfer. This contribution reports on the PPs and the experiences gained. It also introduces proposals for improving the criteria that were derived from this work.

1. IT security certification and criteria

The complexity of today's information technology (IT) makes it impossible to evaluate its security by simple "examination". Moreover, for most users it is hardly possible to conduct more detailed checks, which are necessary for a qualified evaluation, as they cannot afford the expenditure this would entail. Thus, more and more users are faced with the problem of knowing very little about the technique they use for important transactions (e.g. processing sensitive patient data, signing documents, or making payments).

One way to enable confidence in IT is to evaluate and certify products and systems by neutral and competent institutions on the basis of published IT security evaluation criteria. Related certification schemes exist since the mid 80's, for example, in the USA, the UK and Germany. There are regional differences between the schemes, but typically (cf. Figure 1) a sponsor asks (and pays) for an evaluation that is conducted by an accredited (commercial or governmental) IT Security Evaluation Facility (ITSEF) and monitored and certified by a (governmental or commercial) Certification Body (CB). In most cases the sponsor of an evaluation is the vendor of a product or system (Target of Evaluation – TOE). An overview of Certification Schemes and more details can be found in [11].

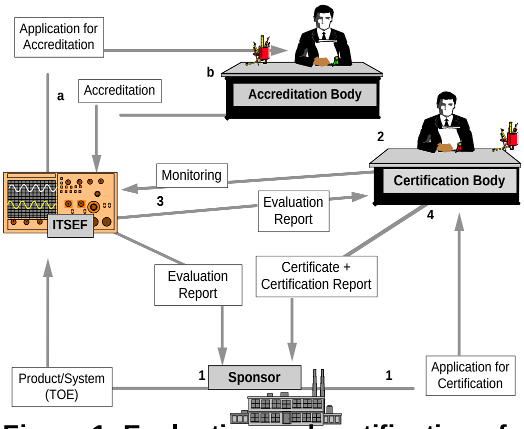

Figure 1: Evaluation and certification of a TOE (1-4) and accreditation of an ITSEF (a-b)

To enable comparisons of evaluation results, criteria catalogs have been developed, which structure the IT security requirements. Some examples are given in Table 1.

While the TCSEC [13] had a rather fixed security model aiming at the confidentiality of military information, subsequent criteria (e.g. ITSEC [3]) had a broader scope. These criteria are frameworks that IT manufacturers, vendors, or users can use to specify what security functions (*Functionality*) they wish to have evaluated and to what depth, scope, and rigor the evaluation should be performed (*Assurance*). In practice, functionality requirements refer to the behavior of the product with regard to security concerns, while assurance requirements specify the actions that the developer, the writers of documentation, and the evaluators must complete.

Independent evaluation can be very useful for privacy enhancing technologies, as those very often aim at the protection of individual users, and this is exactly the user group that usually does not have the resources to assess IT on its own. Of course evaluations and criteria then have to be comprehensive, especially regarding privacy. As the next section shows, this was not the case.

Table 1: Some IT security evaluation criteria and their editors

Publication / Project Dates	Editors	Criteria Name Current Version
1983/85	USA Department of Defense (DoD)	Trusted Computer System Evaluation Criteria (TCSEC – “Orange Book”)
1990/91	Commission of the European Communities (CEC)	Information Technology Security Evaluation Criteria (ITSEC) Version 1.2
1990 - 99	International Organization for Standardization / International Electrotechnical Commission ISO/IEC JTC1/SC27/WG3	Evaluation Criteria for IT Security (ECITS) International Standard 15408: 1999
1993 - ??	Common Criteria Project CDN / D / F / GB / NL / USA Govt. Agencies	Common Criteria (CC) Version 2.1

2. Problems regarding privacy and multilateral security

Various aspects of security certification and the underlying early criteria have been criticized, for example the balance of the criteria and the meaningfulness and use of results (cf. e.g. [7; 11]).

A main point of criticism from the application side was that the criteria were too biased towards hierarchically administered systems and the protection of system operators. The criteria seemed to ignore the fact that threats originate not only from users and outsiders, but also from operators and manufacturers of the systems. *Multilateral security*, i.e. taking into account the security requirements, not only of operators, but also of users and customers, was ignored. Especially privacy aspects of telecommunication transactions were not covered, e.g. unobservability of calls to help lines or anonymous access to patent information on the Internet. From a technical point of view, systems with a decentralized organization and administration were only insufficiently covered. Also, data-collecting functionality was overemphasized, while data-economical functionality was missing.

The following example illustrates how this lack of consideration for user protection in the criteria affects evaluation results. It also shows that in this case the evaluation was focused primarily on the protection of the operators and neglected the protection of users or customers. A function for the selective logging of activities of individual users was classified as a non-critical mechanism that did not need evaluation. In the opinion of the evaluators, failure of this mechanism would not create weaknesses because if the function was not active, the activities of all users were logged [5]. From the operator point of view no real security risk existed, because no audit data would be lost – only perhaps more data than planned would be collected. However, from the users’ point of view this is a considerable risk, because excessive logging and the resulting data can lead to substantial threats for users and customers, e.g. when this data is misused.

3. The CC/ECITS and their privacy families

Since 1990 two initiatives aim at globally uniform evaluation criteria, mainly to enable the mutual recognition of evaluation results. A joint committee of ISO and IEC (JTC1/SC27/WG3) developed the “Evaluation Criteria for IT Security” (ECITS), which are now published as IS 15408 [9]. In parallel to the ISO/IEC standardization, North American and European government agencies developed the “Common Criteria” (CC). Since CC V. 2.0 there is a large convergence with the ISO ECITS, and CC V. 2.1 [1] and IS 15408 are fully aligned.

After the problems with earlier criteria had also been brought up to the attention of ISO/IEC, the new criteria contain a section aiming at privacy protection (cf. Section 3.2). At the moment there are no plans for another version of the CC, but the ECITS will undergo the usual periodic revision of ISO/IEC standards, which will probably be done by JTC1/SC27/WG3 in 2003.

3.1. Overview of the CC/ECITS

The CC/ECITS share the goals and general approach of other evaluation criteria (cf. Section 1), but offer more flexibility. They provide a catalog of *functional components*: a modular, structured library of customizable requirements, each tackling a specific functionality aspect. The CC/ECITS also provide a catalog of *assurance components* and seven *Evaluation Assurance Levels* (EALs). These are an ordered set of packages of assurance components: each EAL contains the lower EAL and further assurance requirements. One can simply choose an EAL but also generate a specific package of assurance components.

On the one hand, the additional flexibility eases the formulation of security targets, but on the other hand, it complicates the comparison of evaluation results.

Table 2: Present CC/ECITS privacy families

Family name	Description	Applications
Anonymity (FPR_ANO)	Ensures that a user may use a resource or service without disclosing his/her identity.	Make enquiries of a confidential nature to public databases, respond to electronic polls, or make anonymous payments or donations.
Pseudonymity (FPR_PSE)	Ensures that a user may use a resource or service without disclosing his/her identity, but can still be accountable for that use.	Charge callers for premium rate telephone services without disclosing their identity, or to be charged for the anonymous use of an electronic payment system.
Unlinkability (FPR_UNL)	Ensures that a user may make multiple uses of resources or services without others being able to link these uses together.	Make multiple use of a pseudonym without creating a usage pattern that might disclose the user's identity.
Unobservability (FPR_UNO)	Ensures that a user may use a resource or service without others being able to observe that the resource or service is being used.	Technology for telecommunications privacy, especially for avoiding traffic analysis to enforce constitutional rights, organizational policies, or defense requirements.

To resolve this problem and still give users the opportunity to formulate their own requirements, the CC introduced the concept of the *Protection Profile* (PP). A PP describes the functionality and assurance requirements for a certain application (e.g. health care administration) or technique (e.g. firewalls). Ideally, several products will be evaluated against one PP, so that the results can be compared. ISO is setting up a registry for PPs and the CC project is maintaining a PP list [2].

3.2. CC/ECITS privacy families

The CC/ECITS contain four *Functional Families* directly related to privacy and organized in a privacy *Class*. Some of their components were inserted late in the editing process (e.g. some of the Unobservability components were not present in CC V. 1.0). Table 2 gives a brief description of the privacy families in the CC/ECITS.

4. Experimenting by writing PPs for mixes

As the CC/ECITS aim at covering security requirements also for “non-traditional” applications not covered in earlier criteria, it seemed useful to examine the new approach by actually using the criteria to produce PPs. Although during the development of the CC a number of example PPs had been produced for testing purposes [2], no PP aiming at privacy existed. To gain more experience as to whether the CC/ECITS are complete and adequate to express requirements on privacy friendly functionality, some example PPs were written.

As application we chose the mix system (cf. Section 5), because it is a prime example of a distributed application where multilateral security concerns involving operators and users come up. The availability of an extensive literature on the subject, of real world implementations and the interest that anonymous and untraceable communication have gained recently, were also favorable reasons that made this kind of application an ideal testing ground.

5. Short introduction into mixes

A mix is a remailer system that aims at hiding the correspondence between sender and recipient of a message [4]. Refinements and other applications besides email, such as ISDN telephony and WWW access are described in e.g. [12, 10, 15]. The basic functionality allows achieving unlinkability of communicating partners, but anonymity can also be achieved if the sender does not explicitly state its identity in the message. As a further development, also pseudonymity can be implemented using a mix remailer system, using so-called “return addresses”.

A mix system achieves untraceability of messages essentially by deploying a distributed architecture, where each node is independently administered. The sender selects a path in the mix network to reach the receiver, and each node resends the message to the next one according to instructions present in the message itself. The message is encrypted in such a way that each relay node only gets to know the node from which it received the message and the node to which it forwarded the message.

There are at least two working implementations of mixes: the first is a free software called Mixmaster [6], which is now in use at various sites, but is generally administered by volunteers and thus not apt for widespread commercial use. A commercial pseudonym-based mix system is being introduced by Zero Knowledge Systems [15], which offers a client product for sending email through a set of independently administered nodes, for which ZKS also produces the remailer software.

6. The Protection Profiles written

Initially, a choice was made to write two PPs following an “architectural” subdivision suggested also by the criteria, i.e. writing a PP for a single mix node, and then one for the whole mix network (cf. Figure 2). This represents the traditional way of subdividing security problems into manageable pieces, and derives from the "secure each part to secure the system" paradigm. In the case of the mix network, however, this path resulted in a dead-end, because the second PP, which stated requirements for the whole network, actually tried to make a compromise between the security requirements of the network, and those of the user of the network. For example, a standard option for protecting the mix network from some kinds of flooding attacks is that of keeping audit logs, which clearly endangers the potentially anonymous users, because someone gaining access to the logs or a corrupt administrator could use the information contained therein to trace back the messages to their original senders.

Our next attempt was to divide the PPs based on the "multiple interests" paradigm, i.e. by writing one document for each of the involved parties (which in the mix system are the administrators and the users) and by considering their security needs and concerns separately. This approach again led to two PPs; the first one (see 6.1) was largely rewritten from the previous PP dealing with a single mix node, and states the security requirements of a single node, which overlap largely with the requirements as felt by the administrator of such a node (e.g. resistance to attacks, secure operating environment, physical protection...). The second (see 1.1) addresses the needs of the user with respect to the whole network, and includes requirements like anonymity and unlinkability of communicating parties. Eventually, it was found that the main challenges for the expressive power of the criteria were posed by this second document, because some of the security requirements related to fundamental privacy-enhancing properties were not to be found in the stock CC/ECITS components.

The process of writing PPs is supposed to be top-down. The author identifies a set of threats, devises a set of security objectives that should counter all the threats, and finally expresses these objectives through a set of CC/ECITS components. The development of the PPs started with an initial survey of the mix literature and implementations to gain confidence with the underlying concepts and technology. This resulted in the production of a set of threats that were then clearly stated and used as the basis of the PP.

The threat list must be obviously *complete* and *relevant*, and this is the reason why each PP was written many times over before reaching a stable state. Also the threats must be stated in a manner to ease the formal demonstration of correspondence with the security objectives, to the degree mandated by the choice of the EAL.

The following sections give an overview over the PPs with the user-oriented PP being described in more detail. The complete text of the PPs is freely available [8] and also contains extensive justifications for the selection of threats and countermeasures.

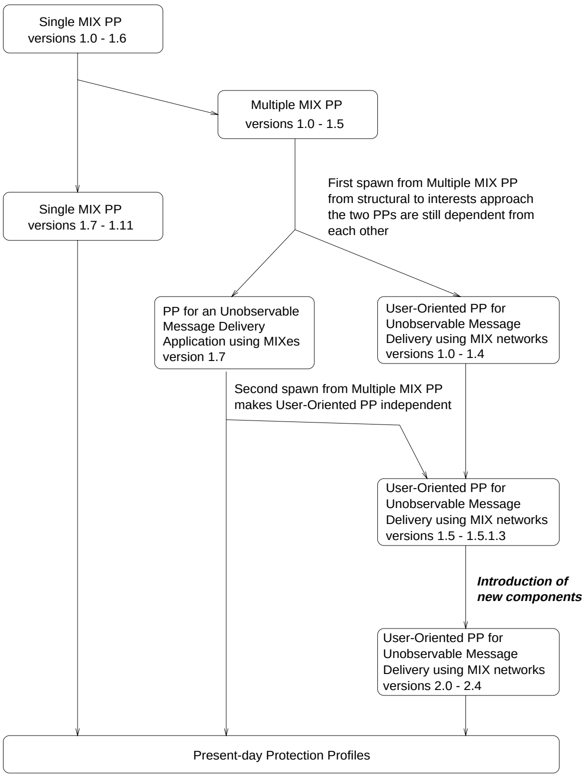

Figure 2: PP development history

6.1. Single Mix Protection Profile

The Single Mix PP was written to address the security problems of a single mix system, without considering the requirements of the user (who wants to send anonymous mail) and also ignoring all the security threats that may derive from the connection of the system with other mixes. Thus the threat list of this PP (Table 3) includes items like *flooding attacks, logical access to the TOE, replay attacks, traffic size analysis*. The last two threats (marked with "TE") are intended to be countered not by the mix itself but by the environment (operating system, etc.).

6.2. Multiple Mix Protection Profile

The "PP for an unobservable message delivery application using mixes" (Multiple Mix PP) was initially written to complement the previous, and to take into account both the entire network of mixes, and the requirements set by the user, which sees the mix network as one homogeneous and opaque entity. Thus, the threats this PP addresses are *message interception* and *denial of service*, as shown in

Table 4. The PP tries to reconcile the needs of the mix operators with those of the users. This leads to a conflict, which is difficult to solve using the standard CC components.

Table 3: Threats in the Single Mix Protection Profile

Threat Label	**Description**
T.DenialOfService	An attacker may try flooding the mix with a great amount of messages, thus causing an overload on the mix and possibly leading to Denial of Service by the mix.
T.FloodingAttack	An attacker may try flooding the mix with a great amount of messages, to single out only one unknown message and discover its destination.
T.ForgeOrigin	An attacker may send to the mix messages with a forged origin field.
T.InterceptMessages	An attacker may intercept and read the content of the messages (including the message origin and final destination, arrival and departure order and time) exchanged by users and the mix or between mixes.
T.LogicalAccess	An attacker (this includes unauthorized users of the host system) may gain access to the mix.
T.ReplayAttack	An attacker may intercept an incoming message and feed it to the mix many times, and, by checking the outgoing messages, discover where it is directed.
T.SizeAnalysis	An attacker may track the sizes of incoming and outgoing messages, thus linking origin and destination.
T.WrapAndImpede	An attacker may completely wrap the mix, thus selectively or totally impeding exchange of messages with the users or other mixes.
TE.UntrustworthyAdministrator	The mix administrator may abuse his trust and compromise completely the operation of the mix.
TE.ImproperAdministration	The mix may be administered by the mix administrator in an insecure or careless manner.

Table 4: Threats in the Multiple Mix Protection Profile

Threat Label	**Description**
T.DenialOfService	The TOE may be isolated from the users by blocking the network connections, and causing DoS.
T.Misuse	Users may misuse the TOE and produce traceable messages, while thinking the message was correctly sent and delivered. The administrators may inadvertently mismanage or badly configure parts of the TOE as to loose the security properties of that part of the TOE.
T.MixPeek	A subverted mix may be able to gain knowledge of the origin and destination of a message by reading its content while processing it.
T.OneStepPath	A mix may gain information linking origin and destination if the path from the origin user to the destination user contains only one mix.
T.TOESubstitution	An attacker may block messages sent by some user and act as the TOE, or a part thereof. Inadvertent users may send messages to the attacker instead of to the TOE, and the attacker may then read origin and destination data and forward the message to the destination.
T.UnreliableNetwork	The connecting network may not be reliable on correctly delivering messages between parts of the TOE. Specifically, messages may be lost, altered or truncated accidentally.
TE.MixConspiracy	Subverted mixes may share input/output information to link origin and destination of a message.

6.3. User-Oriented Protection Profile

The "User-Oriented PP for unobservable message delivery using mix networks" was developed to consider only the needs of the user of the mix network. The tables in this section allow tracing each PP component to a specific threat or policy.

The PP addresses threats like *untrusted mix, misuse, or key forgery* (Table 5). Two "Organisational Policies" (marked with an "O."-label) are also included. They state requirements on the TOE that do not derive directly from any threat. The policies are however treated like threats in the following steps in the sense that "Security Objectives" must be specified to satisfy the policies.

The second step of writing a PP is to specify a set of security objectives that the TOE has to reach to counter all the threats. Table 6 shows the security objectives stated for this PP. Like the threats, the security objectives are divided in two categories, namely, objectives which are to be achieved solely by the TOE, and objectives for which the surrounding environment (operating system, administration, etc.) are partly or wholly responsible.

Table 5: Threats and Organisational Policies in the User-oriented Protection Profile

Label	Description
O.Anonymity	The TOE shall provide for an anonymous message delivery service; that is, the recipient of a message shall not be able to know the origin of the message, unless the author expressly inserts this information in the message body.
O.Untraceability	The TOE shall provide for an untraceable message delivery service; this means that, taken any message transiting through the system at any time, it shall not be possible to obtain enough information to link its origin and destination users.
T.ContentDisclosure	An attacker might intercept transiting messages between parts of the TOE and read their content, thus disclosing it, together with any related information. *Note: This is a threat not only to the operation of the TOE, but also to the user, whose communications might be traced. In particular, this threat relates to messages transiting from the user client to a node on the network and refers to both the original message content (written by the user), and also to the routing information and other auxiliary information in the message.*
T.EndPointTrafficAnalysis	An attacker might intercept transiting messages between parts of the TOE (user client and mix node), and use the related information to perform traffic analysis on a user. *Note: This threat relates to the concepts of sender anonymity and receiver anonymity. As viewed traditionally, the main goal of the mix network is to hide the relation between message sender and receiver (this property also known as sender/receiver anonymity). However, once a possible communication between two users is suspected, one may be able to monitor the end points of message chains for a statistical correlation between transmission and reception times, especially if the traffic on the network is low, the users few, and the per-user traffic low.*
T.KeyForgery	An attacker might generate forged keys, simulating the activity of a given mix, distribute them, and make the user employ them to encrypt message in the belief that such messages are only readable by the replaced mix. *Note: This is a threat to the originating user, who will send messages readable to an attacker, and might not be warned about it. A trust scheme (implemented for example by a certification authority) is required to counter this threat.*
T.Misuse	The user might install, configure or use the TOE interaction functions in an insecure manner, hence compromising the expected security properties offered by the TOE. *Note: This threat is particularly relevant when considering the “human” element when this is the user, because the user is not expected to have as deep a knowledge about the TOE functions and about the security concerns as, for example, a system administrator, who represents the human element in the case of an administered mix node.*
T.OneStepPath	A mix may gain information linking origin and destination if the path from the origin user to the destination user contains only one mix.
T.UntrustworthyMix	Some mix(es) in the network may be compromised and hold, process and/or disclose information useful to trace, and/or reveal the content of, communications.
TE.MixConspiracy	Some mixes in the network may be compromised and share information useful to trace, and/or reveal the content of, communications. *Note: This threat represents an extension to the T.UntrustworthyMix threat, in that it introduces the concept of information sharing between parts of the TOE.*
TE.PartialNetworkBlock	An attacker might block the connection between parts of the TOE and the user. *Note: This is a typical DoS attack, where part or the entire TOE is rendered unusable.*
TE.Redirection	An attacker might redirect the connections between parts of the TOE and act as to replace that part seamlessly, thus effectively acting as a compromised mix subset.

A mapping between objectives and threats must be demonstrated (the rigor of this analysis depends on the selected EAL), but this is omitted here due to space constrains. However, Table 7 shows the mapping between threats and objectives.

After specifying the objectives, the PP author must select functional components from the criteria, which are fit to reach the objectives and give a mapping demonstration. In this phase the most problems arose caused by the expressive deficiencies of the CC/ECITS components. This eventually led to proposing new components (Section 7 describes the alternatives that were tried before resorting to this option).

Table 6: Security Objectives in the User-oriented Protection Profile

Security Objective Label	Description
SO.AdequateDocumentation	The TOE shall provide the user with adequate, readable documentation on the correct use of the security functions.
SO.Anonymity	The TOE shall accept and process messages without requiring that the processed data may be in any way linked to the origin user.
SO.ConcealMessageContent	The TOE shall enforce that the content of all messages transiting on the network be inaccessible to all third parties, in whatever point of the network the messages are intercepted.
SO.CounterTrafficAnalysis	The TOE shall be constructed as to counter traffic analysis techniques specifically aimed at analyzing the communications between user client software and the mix network.
SO.DivideSecurityInformation	The TOE shall be constructed as to provide the user the ability, and enforce the correct use of such ability, of determining the allocation of unlinkability-relevant data among different parts of the TOE.
SO.DivideSecurityProcessing	The TOE shall provide to the user the ability, and enforce the correct use of such ability, of freely choosing a combination of mix nodes among which to allocate the processing activities achieving unlinkability.
SO.EnforceProperUse	The TOE (and especially the user interface part of the TOE) shall enforce the proper and secure use of the security functions of the TOE. *Note: Require e.g. secure pass phrases, encryption, and minimum message chain length.*
SO.EnforceTrustDistribution	The TOE shall enforce the user's choice of information and processing distribution.
SO.Identity	The TOE shall uniquely identify the single mix nodes and users and provide means to transmit data to a specific mix while preserving the confidentiality of such data.
SO.KeyTrustAssurance	The TOE shall provide the user the ability, and enforce the correct use of such ability, of validating any public key used for encryption purposes against some trusted mechanism, to gain confidence that the communicating partner is actually who he claims to be.
SO.MinimizeSecurityInformation	The TOE shall be constructed as to minimize the use, distribution and availability time frame of information impacting unlinkability.
SO.Untraceability	The TOE shall also ensure that no subject (user, administrator, threat agent) has the possibility to gain sufficient information as to track back the origin of a message.
SOE.AntagonisticManagement	The TOE shall be independently and antagonistically managed. *Note: The main problem with this security objective to be fulfilled by the environment is that it is nearly impossible to enforce it without some form of post-deployment assurance evaluation control and maintenance.*
SOE.DistributedNetwork	The TOE shall rely on a topologically distributed network. *Note: this is required to maximize the resources an attacker must deploy in the attempt to “cut off” a part of the network from the rest. Apart from requiring specific design choices, this requirement can only be met by implementing a sound collective administration policy, and by providing means to assure the users of the effects of such a policy.*

After developing the new components, a second version of the PP was written, which keeps the threats and objectives of the first version, and uses the new components to express its requirements, accordingly to the recommended top-down practice for PP development. The new version of this PP is decidedly simpler, more effective, and more precise in the requirements definition. Table 8 shows the functional components used by the PP that employs the new proposed components (marked in the third column) and also shows the relations between components and objectives. Objectives not “covered” by any component must be addressed by assurance requirements or by additional requirements on the environment, which are however not relevant at this point, and are here omitted.

For this PP EAL 5 was selected to assure that the TOE is developed, delivered, and evaluated following rigorous commercial practices. A formal model of the TOE security policies must be provided and evaluated, and the system must be independently tested.

7. Experiences gained

The top-down methodology has many advantages, the main one being that the development process of the PP is clean, and the formal demonstration of correspondence between the various threats, objectives and requirements is relatively simple. The problems arise when the PP author needs to express requirements for security objectives not covered by CC/ECITS components.

Table 7: Mapping Security Objectives to Threats and Organizational Policies

	O.Anonymity	O.Untraceability	T.ContentDisclosure	T.EndPointTrafficAnalysis	T.KeyForgery	T.Misuse	T.OneStepPath	T.UntrustworthyMIX	TE.MIXConspiracy	TE.PartialNetworkBlock	TE.Redirection
SO.AdequateDocumentation						*					
SO.Anonymity	*										
SO.ConcealMessageContent			*								
SO.CounterTrafficAnalysis				*							
SO.DivideSecurityInformation							*	*			
SO.DivideSecurityProcessing							*	*			
SO.EnforceProperUse						*					
SO.EnforceTrustDistribution						*	*	*	*		
SO.Identity								*			*
SO.KeyTrustAssurance					*						
SO.MinimizeSecurityInformation								*			
SO.Untraceability		*									
SOE.AntagonisticManagement									*		
SOE.DistributedNetwork										*	*

During the development of the user-oriented PP, three such issues were identified:

1. Requirements on the distribution of the TOE: although it may be viewed as a purely architectural requirement, it is worthwhile to note that many secure systems are based explicitly on a distributed architecture to perform the security relevant tasks. Mixes are an example, but also digital payment systems, etc. show such patterns.
2. Requirements on the policies requiring the minimization of knowledge: clearly information that has been disposed of cannot be disclosed. Deleting information as soon as it is not anymore essential to the operation of the system is thus always a safe practice.
3. Requirements on unlinkability properties to be enforced by the TOE: the statement of unlinkability of operations is possible through the stock CC/ECITS components, but not so for unlinkability of users, which is precisely what the mix network provides.

To solve the expressive deficiencies of the CC/ECITS a number of options may be considered, and the following three are worthwhile to mention:

1. Restate the security objective differently, (i.e. “fit” the objective to the requirements),
2. Force the criteria components to cover the objective (i.e. “fit” the components to the objective),
3. Develop new functional components.

The first two options are not viable in the long run. The first one breaks the top-down paradigm, and distorts the PP to state what is expressible by the criteria, necessarily avoiding all security issues which are not simply stateable by the CC/ECITS. The second option “overloads” the CC/ECITS components to express requirements for which they were not thought. This has many drawbacks; for one thing, it simply may not be always possible. Moreover, the requirements tend to become unclear, and ineffective, and the PP evaluation becomes more complicated because of the convoluted use of the components.

The third option has undoubtedly many formal and theoretical advantages, and some drawbacks. On the one hand, the requirements may be stated in a simple fashion, and the top-down structure is preserved. On the other hand, while the CC/ECITS allow for expansion of the base requirements sets, one of their main advantages (easier comparability of PPs) is not guaranteed for PPs that use such novel components.

The full discussion of the various problems encountered, and of how it was decided to write new components is too lengthy to be included here, but it can be said that each of the previous issues arose when trying to express specific objectives through the criteria, and an effort was made to approach the problem by using all three strategies [8]. In each case, the conclusion was that the technically best way to proceed was to develop new components. The decision might have been different in the situation of a concrete evaluation. In such a case, resource constraints (getting an evaluation through without spending too much time discussing novel approaches) and the need of easing the evaluation process (therefore staying with the standard set of components) might have got priority. However, with respect to improving the CC/ECITS, two new families and one largely revised family are proposed in the next section.

Table 8: Mapping Functional components to Security Objectives

		New Component	SO.AdequateDocumentation	SO.Anonymity	SO.ConcealMessageContent	SO.CounterTrafficAnalysis	SO.DivideSecurityInformation	SO.DivideSecurityProcessing	SO.EnforceProperUse	SO.EnforceTrustDistribution	SO.Identity	SO.KeyTrustAssurance	SO.MinimizeSecurityInformation	SO.Untraceability	SOE.AntagonisticManagement	SOE.DistributedNetwork
FCS_CKM.1	Cryptographic key generation										*					
FCS_CKM.2	Cryptographic key distribution											*				
FCS_CKM.4	Cryptographic key destruction											*				
FCS_COP.1	Cryptographic operation				*											
FDP_ACC.2	Complete access control (MUDAC)									*	*					
FDP_ACF.1	Security attribute based access control (MUDAC)									*	*					
FDP_IFC.1	Subset information flow control (CCE)					*										
FDP_IFF.4	Partial elimination of illicit information flows					*										
FDP_IRC.2	Full information retention control	*											*			
FDP_ITT.1	Basic internal transfer protection				*											
FDP_RIP.2	Full residual information protection												*			
FIA_ATD.1	User attribute definition										*					
FIA_UID.1	Timing of identification						*	*								
FMT_MSA.1	Management of security attributes						*	*								
FMT_MSA.2	Secure security attributes						*	*	*							
FMT_MSA.3	Static attribute initialisation						*	*	*							
FMT_SMR.1	Security roles						*	*								
FPR_ANO.2	Anonymity without soliciting information			*												
FPR_TRD.2	Allocation of information assets	*					*			*					*	
FPR_TRD.3	Allocation of processing activities	*						*		*					*	
FPR_UNL.2	Unlinkability of users	*												*		

8. New and revised functional families

Three new functional families are proposed to be included in the CC/ECITS set. They are aiming at a more precise handling of privacy-related information and are summarized in Table 9. Each family is discussed in a separate section below. The formal family description that follows the typographical, layout and content standards of the CC/ECITS, can be found in [8].

When writing new components for the CC/ECITS, one of the most complex decisions is to identify an “abstraction level”. The decision is complex because the level of abstraction of the stock CC components varies greatly. Some of the requirements are very low-level (e.g. stating that the password input procedure should conceal the typed password on the output terminal), others are at a very high level (for example requirements which state access policies). The choice of the abstraction level obviously influences the formulation of the requirements and their expressive power. Low-level requirements are easily verifiable on the TOE implementation, while higher-level requirements may require as much as a separate analysis and a formal security model (provision for security models is provided in the assurance requirements section of the CC/ECITS). In writing the new components an effort was made to maintain a sufficiently general approach (which allows also to reuse the component) while aiming at precise, comparatively easily assessable security properties.

Table 9: Proposed new and revised functional families

Label	Name	Purpose
FDP_IRC	Information retention control	Limit the accumulation of non-essential information.
FPR_UNL	Unlinkability	Extend the current unlinkability requirements.
FPR_TRD	Distribution of trust	Allow users to allocate information and processing activities in a way protecting their privacy.

8.1. Information retention control (FDP_IRC)

The "Information retention control" family addresses a basic need in secure information processing and storage applications, which however appears not to be covered by the CC/ECITS. Namely, this is the need for secure management of data no more needed by the TOE to perform its operation, but still stored in the TOE. Examples of collected information may include the following:

- Connecting IP numbers on anonymizing proxy;
- One-time cryptographic keys, which if eventually disclosed could allow the decryption of intercepted and stored communications or files;
- TOE usage histories, as interactive command line shell histories, or information presentation tools cache files (i.e. WWW browser caches and histories), which, while useful to the user and TOE during specific activities, could be used to track user or TOE actions, if preserved across sessions.

The traditional view of IT systems as data storage systems induced naturally into thinking that once entered, data would be seldom deleted from the system, and if so, mainly because of space exhaustion problems. But in a multilateral or high security environment it is important to minimize the distribution, and temporal time frame in which information is contained in the system. Also, users might want their IT products to avoid retaining data that they consider exploitable by third parties or threatening their privacy. In this case, such a requirement can help users to gain confidence that the product is secure, as far as it deletes every copy of the data when not needed anymore. An effort was made during the development of the PP, to state this requirement using the standard components available in the CC/ECITS, particularly, using the access control requirements, and stating some very complex access control policies using such components. However, the conclusion was reached that using such components for an end for which they were not intended (namely, traditional access control policies in a TOE) caused many harmful consequences, apart from breaking the top down approach as mentioned in Section 7:

- The PP becomes more complex, and thus more difficult to read and to evaluate.
- The correspondence demonstration is also more complex.
- The resulting requirements, however well formulated, are simply not tight enough, and the PP reader is induced into a false sense of security, mistakenly thinking that the PP actually protects the user, even when this is not the case.

For these reasons the Information retention control family was developed. The family ensures that information no longer necessary for the operation of the TOE is deleted by the TOE. Components of this family require the PP author to identify TOE activities and objects required for those activities, and not to be kept in the TOE, and the TOE to keep track of such stored objects, and to delete on-line and off-line copies of unnecessary information objects.

8.2. Unlinkability (FPR_UNL)

In most cases the CC/ECITS model the properties and behavior of a TOE by specifying a set of relevant entities and imposing constraints on the relationships between such entities. For this purpose, the following four entities are defined:

- Subject: "An entity within the TSC[1] that causes operations to be performed"; this can be, for example, a UNIX process;
- Object: "An entity within the TSC that contains or receives information and upon which subjects perform operations"; for example a file, a storage medium, a server system, a hardware component;
- Operation: a process initiated by a subject or user, which employs a subject to interact with one or more objects or subjects; this term is not directly defined in the criteria's glossary.
- User: "*Any entity (human user or external IT entity) outside the TOE that interacts with the TOE*". It is necessary to clearly distinguish between subject and user. "User" is the physical user with all its attributes (name, role...), or an external IT entity (i.e. another system interacting with the TOE), which initiates op-

[1] TSC = TSF Scope of Control: The complete set of interactions that are under the control of the TOE to satisfy its security requirements and to implement its security features.
TSF = TOE Security Functions: The complete set of functionalities required by the TOE to satisfy its security requirements and to implement its security features.
Both definitions are slightly shortened versions of those in CC/ECITS to ease reading.

erations on the TOE, which are carried out, on its behalf, by subjects operating in the TOE.

For example an unlinkability of operations requirement would impose a constraint on the relationship between operations in the TSC relating them to a particular user. However, the full expressive potential of this model is not described by the standard CC/ECITS components. Figure 3 shows the current situation.

With regard to unlinkability, the CC/ECITS contain the FPR_UNL.1 component that provides unlinkability of operations. Although useful, this component does not cover at least one case, which is of primary importance for mixes: the unlinkability of users, in relation to a specific data object (the mail message). This kind of property is also hard to express through the other families: one could try using the unobservability (FPR_UNO) family, which is however not adequate because the action itself of transmitting a message is not hidden by the mix system. The mix hides only the relation between users, and between email and user.

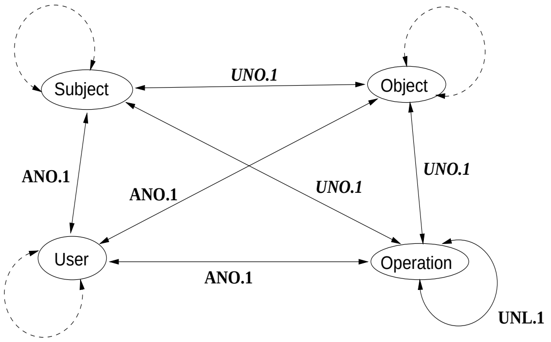

Figure 3: Unlinkability properties covered (solid arrows) and not covered (dashed arrows) by existing components

8.3. Distribution of trust (FPR_TRD)

Among the current families in the privacy class of the CC/ECITS no provision is made to address privacy requirements related to the distribution of trust among parts of the TOE, except in the FPR_UNO.2 component. Trust may be defined, not only in an IT setting, as "*Assured resting of the mind on the integrity, veracity, justice, friendship, or other sound principle, of another person; confidence; reliance.*" [14]. In a more restrictive definition, one may define it as "*confidence on the integrity of another person or organization in the managing of an asset given to him, her or it*". In this context, trust division may be described as the process of allocating assets among different trustees with the aim of minimizing the damage, which one might suffer if one of the trustees betrays the trust given.

As in IT the main asset is *information*, its accidental or intentional loss or mismanagement may result in great damages. Data may be either supplied directly to an information system, as inputted files, documents, personal information, or may be derived from interaction with the system, such as data regarding on-line time and login times of a user, requests and destination of email deliveries and WWW accesses, or called telephone numbers; often the collection of this kind of information is not clearly stated in the (contract) terms which bind user and operator of a system. Figure 4 shows the hidden information processed and possibly stored in a system, which provides textual data transmission capabilities to users. Another related observation is that the processing itself produces information, whose existence or content may not even be known to the user that requested the processing activity to be initiated (e.g. large WWW sites with distributed redundant servers redirect requests to one of the servers in a pool; this mechanism is not visible to the end user, neither is the server choice known.)

Also for this requirement an effort to state it through the standard CC/ECITS components was made, involving the use of a component (FPR_UNO.2) that allows stating requirements on the allocation of information causing observability. However, many problems arouse with this attempt. First, for modeling the requirements on mixes we would have needed a component stating requirements on the allocation of information causing *linkability*, especially linkability of communication partners. Also, similar to the problems described in Section 8.1 a "fake" access control policy was needed to state the requirement, thus lengthening the statement.

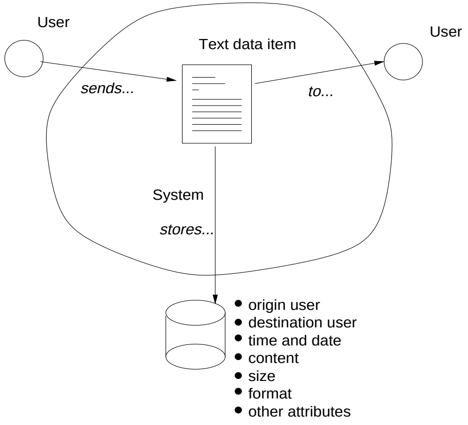

Figure 4: Hidden activities and involved in sending a data object through a system

The "*Distribution of trust*" family addresses both aspects of the trust issue, i.e. the distribution of information, and the distribution of processing activities, which may produce privacy-relevant information. The family describes specific functions that can be used to allocate information and processing activities on the TOE with the objective of protecting the privacy of users of the system. To allow such allocation, the concept of "*Administrative*

Domain" (AD) is introduced to indicate a part of the TOE whose security functions are accessible and usable to access data by a single subject (system user, administrator...) without requesting any additional authorization or performing additional authentication procedures.[2]

The mix is a prime example of a system that aims at isolating into separate ADs all the information generated by the transit of a message over the network, like routing information, reply blocks, and destination addresses. However, the concept of AD can also be useful to model general privacy-related security requirements in various environments (e.g. operating systems like standard UNIX systems provide only one AD: the system administrator can access all data and security functions in the TOE).

9. Summary and conclusion

The experience gained while writing the PPs and the new functional components includes the following major issues:

1. In general the CC/ECITS provide much more flexibility than their predecessors. They also contain much better instruments to describe privacy friendly functionality. However, the CC/ECITS do not offer all the components to formulate privacy-related objectives and properties.
2. The greatest challenges to the expressive capacity of the privacy-related functional components appear in situations where a point of multilateral security is raised (security of the TOE vs. security of the user), namely in this case the Multiple Mix PP and in the User-Oriented Mix PP.
3. For some applications, architectural choices and objectives (i.e. distributed vs. centralized system) influence the security properties of the system. This applies to mixes, but holds also for other "secure" applications, as digital money, information handling and storage, etc.
4. Simply trying to force the application's requirements or the functional components to "fit" is not a sustainable solution, as it results in an unclear and ineffective requirements definition. This issue will become more relevant with more "novel" applications being evaluated against the CC/ECITS.
5. The proposed components aim at enhancing future versions of the CC/ECITS, even when the respective part of the criteria becomes slightly longer. Privacy oriented functionality covers only a small part (ca. 10%) of the criteria, so there should be space for the improvements.
6. Especially in the area of communication the evaluation of service security becomes important for users. While the CC/ECITS provide some help for this further work is needed to further enhance the criteria to support environments where competing security interests exist.

10. References

[1] Common Criteria Implementation Board, *Common Criteria for IT Security Evaluation*, V. 2.1, August 1999, http://csrc.nist.gov/cc

[2] Common Criteria Project, List of Protection Profiles, http://csrc.nist.gov/cc/pp/pplist.htm

[3] European Commission, *IT Security Evaluation Criteria*, V. 1.2, 1991-06-28, Office for Official Publications of the EC, also http://www.itsec.gov.uk/docs/pdfs/formal/ITSEC.PDF

[4] D. Chaum, "Untraceable electronic mail, return addresses, and digital pseudonyms", *CACM*, Vol. 24, No. 2, pp. 84-88.

[5] C. Corbett, "ITSEC in Operation – an Evaluation Experience", *Proc. 4th Annual Canadian Computer Security Conference*, May 1992, Ottawa, Canada, pp. 439-460.

[6] L. Cottrell, *Mixmaster & Remailer Attacks*, http://www.obscura.com/~loki/remailer/remailer-essay.html

[7] Privacy Protection and Data Security Task Force of the German Society for Informatics, *Statement of Observations concerning the ITSEC*, V1.2, 1992-02-24, edited in Data Security Letter, No 32, April 1992.

[8] G. Iachello, *IT Security Evaluation Criteria, and Advanced Technologies for Multilateral Security – The MIX example*, June 1999, http://www.iig.uni-freiburg.de/~giac

[9] ISO/IEC, *Evaluation Criteria for IT Security (ECITS)*, Parts 1-3, International Standard 15408, 1999-12-16.

[10] A. Jerichow, J. Müller, A. Pfitzmann, B. Pfitzmann, M. Waidner, "Real-Time Mixes: A Bandwidth-Efficient Anonymity Protocol", *IEEE Journal on Selected Areas in Communications*, 16/4, pp.495-509.

[11] K. Rannenberg, "What can IT Security Certification do for Multilateral Security?" in G. Müller, K. Rannenberg: *Multilateral Security in Communications – Technology, Infrastructure, Economy*, Addison-Wesley-Longman, 1999, ISBN 3-8273-1360-0, pp. 515-530.

[12] P. F. Syverson, D. M. Goldschlag, M. G. Reed, "Anonymous connections and onion routing", *Proceedings of the 1997 IEEE Symposium on Security and Privacy*, IEEE Press, Piscataway NJ

[13] USA Department of Defense, *Trusted Computer System Evaluation Criteria*, DOD 5200.28-STD, December 1985, http://www.radium.ncsc.mil/tpep/library/rainbow/5200.28-STD.html

[14] *Webster's Revised Unabridged Dictionary*, 1913, ftp://ftp.dict.org/pub/dict/

[15] *The Freedom Network Architecture*, Version 1.0, Zero-Knowledge Systems, Inc. http://www.freedom.net/info/freedompapers/

[2] The AD is a formalization of the concept of the more intuitive "part of the TOE", which is also used in the statement of the CC/ECITS Unobservability component.