Development of a
Secure Electronic Marketplace for Europe

Michael Waidner

IBM Zurich Research Laboratory
CH 8803 Rüschlikon, Switzerland
e-mail <wmi@zurich.ibm.com>

Abstract: Backed by the European Commission, a consortium of partners from European industry, financial institutions, and academia has embarked on a research project to develop the fundamentals of secure electronic commerce. The goal of Project *SEMPER* (Secure Electronic Marketplace for Europe) is to provide the first open and comprehensive solutions for secure commerce over the Internet and other public information networks. We describe the objectives and summarise the initial architecture of *SEMPER*.

1 Introduction

Backed by the European Commission, a consortium from European industry and academia has embarked on a research project to develop the fundamentals of secure electronic commerce. The goal of the 9-million ECU project, *SEMPER* (Secure Electronic Marketplace for Europe), is to provide the first open and comprehensive solutions for secure commerce over the Internet and other public information networks.

A wide range of businesses are rapidly moving to explore the huge potential of networked information systems, especially with the Internet-based WWW (World-wide Web). The Internet, which already connects more than 3 million computers and a substantially larger number of users, is growing at a breathtaking pace with thousands of newcomers every day. Although the Internet has its roots in academia and is still dominated by free-of-charge information, dramatic changes are expected in the near future. For instance, the WWW will be used for a wide variety of electronic commerce such as on-line trade or delivery of advanced multimedia information services. The evolution of broadband networks and "information highways" will intensify this trend.

The need for secure transactions in this new business environment, which involves networks available to the general public, has triggered a number of related efforts. These initial developments are based almost exclusively in the US and most of them are limited to proprietary, or otherwise closed solutions, involving only electronic payment issues. In contrast, *SEMPER* is directed towards a comprehensive solution for secure electronic commerce, considering legal, commercial, social, and technical requirements as well as different options for an electronic marketplace.

Proceedings of ESORICS 96, Rome, September 1996;
LNCS 1146, Springer-Verlag, Heidelberg 1996, 1-14.

SEMPER started on September 1st, 1995. The first of the three project phases addresses a coherent security model and a generic, open security architecture for the electronic marketplace. This architecture is independent of specific hardware, software, or network architectures. The most fundamental electronic commerce services, such as secure offering, order, payment and information delivery, are also integrated in the first phase.

Subsequent phases will concentrate on more advanced services. These will include fair exchange of documents, credentials, advanced document processing, notary services and multimedia-specific security services, such as protection of intellectual property rights. Multi-party security and protection of users' privacy receive prime attention. *SEMPER* uses and integrates existing architectures, tools, and services where appropriate.

Trials will be provided for WWW and ATM-based broadband networks. They will demonstrate the broad applicability of *SEMPER*'s architecture and services.

The *SEMPER* project is part of the Advanced Communication Technologies and Services (ACTS) research program established by the European Commission Directorate General XIII for 1994-1998 [http://www.analysys.co.uk/acts/cec/].

The members of the *SEMPER* consortium are *Cryptomathic* (DK), *DigiCash* (NL), *EUROCOM EXPERTISE* (GR), *Europay International* (B), *FOGRA Forschungsgesellschaft Druck* (D), *GMD - German National Research Center for Information Technology* (D), *IBM* (CH, D, F), *INTRACOM* (GR), *KPN Research* (NL), *Otto-Versand* (D), *r3 security engineering* (CH), *SEPT* (F), *Stichting Mathematisch Centrum / CWI* (NL), *University of Freiburg* (D), *University of Hildesheim* (D). *Banksys* (B), *Banque Générale du Luxembourg* (L) and *Telekurs* (CH) are associated with *SEMPER*. *IBM Zurich Research Laboratory* provides the technical leadership for the project.

2 Electronic Commerce

Like on a physical marketplace, the main purpose of an electronic marketplace is to bring potential *sellers* and *buyers* together:

- Sellers *offer* their goods and buyers *order* these goods; together this is a two-party *negotiation*, sometimes ending with a *contract.*
- Both seller and buyer might need certain *certificates* for such a contract. For instance, a buyer might only want to buy from sellers that are accredited with a well-known payment system provider, so that they can use a certain payment instrument, or they may only trust them if a consumer organisation has declared them trustworthy, or a seller might be allowed to deliver certain goods only to residents of the European Union.
- Sellers *deliver* their goods and buyers make *payments*; together this is a two-party *(fair) exchange.*

- Instead of goods, the buyer might receive a specific certificate that subsequently enables *conditional access* to certain services, e.g., like a subscription to a journal.
- Buyers or sellers might be dissatisfied with what has happened so far, i.e., several *exception handlers* and *dispute handlers* are necessary.
- Some services require third parties to co-operate, e.g., *notaries* and *financial institutions.*
- Many services require that buyer and seller have some relations already established, e.g., to banks or government agencies. This requires *registration* and *certification*, and in most cases also *directory* authorities.

In all these actions, the parties have specific *security requirements,* namely integrity, confidentiality, and availability. Confidentiality includes anonymity which is often a requirement for browsing catalogues or purchases for small amounts.

Several typical scenarios of electronic commerce are to be covered by *SEMPER*:

- *Mail-order Retailing:* A retailer accepts electronic orders and payments, based on digital or conventional catalogues, and delivers physical goods.
- *On-line Purchase of Information:* Like mail-order retailing, but with digital, maybe copyright-protected goods that are delivered on-line.
- *Electronic Mall:* An organisation offers services for several service providers, ranging from directory services (“index”) over content hosting to billing services.
- *Subscriptions:* An organisation offers services on a subscription basis, e.g., subscription to news services, database services, or journals. The subscription might be valid only for some time, and it might be transferable or not.
- *Statements:* Transfer of electronic documents, supporting all kinds of security requirements, such as confidentiality and non-repudiation of delivery. A statement might be based on a pre-defined statement template certified by a third party.
- *Contract Signing:* Two or more parties exchange signed copies of the same *statement*.
- *Insurance:* Subscription to an insurance, payment of fees, regulation of damages.
- *Auctioning:* Users participate in an auction, maybe anonymously, and with the usual fairness requirements.
- *Ticketing:* A user buys a ticket that can be used to access a certain service for some time or exactly once, etc., and for that user or for the user’s family, etc.

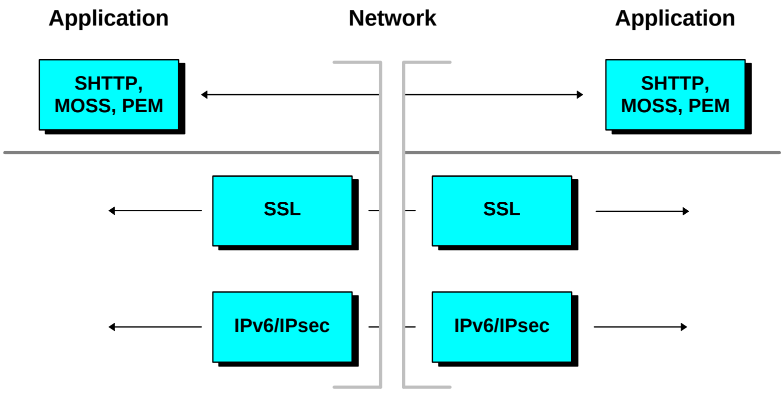

Fig. 1. Proposed Internet security enhancements

3 Existing Technology

The development of electronic commerce on the Internet has come about in a very fast but highly disorganised manner. Currently, there is only a limited understanding of the functionality and security properties of services that are required by merchants and their potential customers. Coherent strategies for marketing, advertising, accounting, and payment are missing. Neither a comprehensive model of an electronic marketplace nor a generic functional and security architecture exists.

Most proposals for electronic commerce originated from one of the following three classes:

Communication security protocols: Most proposals for secure electronic commerce are based on techniques for classical end-to-end security. The currently best known protocols are the following:

- SHTTP [ReSc 95], PEM [Linn93], and MOSS [CFGM95] are extensions to HTTP, electronic mail, and MIME, respectively. In order to use them one has to modify the applications, e.g., one needs security enhanced browsers and servers in order to use SHTTP. They work on individual application layer messages, which is an advantage for electronic commerce because digital messages are used like paper documents: for disputes one needs individually signed messages. *CommerceNet* [http://www.commerce.net] has developed some examples of how SHTTP can be used to simulate paper forms that must be filled-in and signed (e.g., cheques).
- SSL [HiEl 95] and IPv6/IPsec [Atki95] offer secure communication *below* the application layer. Therefore they can be used almost transparently. Their main problem with respect to electronic commerce is that they do not work on documents, i.e., the user does not receive something like a signature that can be stored and used in case of disputes.

All mentioned protocols use the same set of security mechanisms and cryptographic algorithms, primarily digital signatures based on RSA, encryption based on DES and RSA, and MD5 as hash function (for an explanation of all these techniques, see [Schn 96]). All of them were developed in the US, and since they provide an open interface to strong cryptography they are subject to US crypto export regulations. They all require a public-key infrastructure. Both SSL and SHTTP are integrated in commercial products, and most vendors of web browsers and servers announced to support them in their products.

Merchant servers with support for secure transactions: The best known example of such a commercial server comes from *OpenMarket* [http://www.openmarket. com]. From a security point of view, the heart of their architecture is a "payment switch" [GSPT 95]. The payment switch supports different types of customer identification (e.g., password, some secure tokens), collects payments (supporting different payment systems; *OpenMarket* announced to support *CyberCash* [http://www.cybercash. com]), and grants access to information (i.e., specific URL's of short life time) after successful payment. The server supports *SSL* and *SHTTP*. Obviously the architecture is highly centralised and considers the server side only.

Electronic on-line payment systems: Most of the existing work on electronic commerce services concentrates on the development of electronic payment systems. The spectrum of systems includes (see [JaWa 96] for more details):

- systems that do not use any strong protection methods and require prior registration of user accounts, and may be considered as insufficiently secure;
- systems that implement a credit card model, processing customer authentication and payment information by specific security protocols, e.g., *iKP* [BGHH 95] and the proposed Mastercard/VISA standard *SET* [SET 96]);
- one system (*ecash*, from DigiCash, see also [Chau 89]) that implements an anonymous electronic cash model.

Outside the Internet, some interesting, smartcard-based off-line payment systems were developed, which could be used on-line as well. The spectrum ranges from classical electronic wallets and purses to systems that provide strong multi-party security and anonymity (e.g., the system developed by the ESPRIT Project CAFE [BBCM 94]). The leading payment system companies, Europay/Mastercard/VISA intend to support transactions based on smartcards (they published joint specifications), and the US Financial Services Technology Consortium (FSTC) initiated a project that will use a PCMCIA card as "Electronic Checkbook," also via Internet. All these approaches share the problem that the customer's stations need an interface to smartcards or PCMCIA cards, which is not the case in general, yet. Probably this will change in the near future.

None of the different existing or proposed on-line payment systems are interoperable. Most of them do not provide strong multi-party security or user privacy.

Public-key infrastructure: There are mechanisms and standards for key certification, e.g., CCITT X509. Up to now, there is no sufficient certification infrastructure for public keys, but several projects aim at this. Examples are the TEDIS Project FAST, and activities within RARE and TERENA, based on the results of the EU VALUE Project PASSWORD. Several national post offices (e.g., the USPO) plan to provide such services.

Miscellaneous: In addition there are several initiatives that primarily aim at co-ordination and consensus forming, like *CommerceNet*. Similar initiatives exist or are proposed in Japan and Europe.

Beyond these systems, few other services are available for electronic commerce. The experimental *NetBill* [SiTy 95] system supports accounting and billing based on central billing servers. Several companies offer technology for secure metering or copyright protection based on superdistribution (or variant thereof) [MoKa 90]. Some companies offer tools for using EDI messages in electronic commerce over the Internet.

What is missing? Some aspects of secure electronic commerce are not covered by any of the mentioned projects, or at least not in a sufficient form:

- All listed technical projects deal with partial aspects of secure electronic commerce only. No project aims at the complete picture, i.e., at defining a complete model and architecture for secure electronic commerce.
- Although some systems are supposed to become standards, only few standardised API's exist. Defining generic API's and gateways between protocols is absolutely required for an open marketplace.
- Most electronic commerce systems are closed: They use proprietary technology, or support only a specific set of protocols and mechanisms. Often they are based on one central server that acts as a trusted third party for all participants, per marketplace. Often they require specific browsers and servers to be used.
- Although most proposals use public-key cryptography, only little attention is paid to multi-party security. No decision procedures for disputes are defined, which would be necessary for non-repudiation of origin. Usually no security requirements are explicitly formulated, and often no systematic security evaluation is performed.
- The aspects of customer anonymity and privacy are not sufficiently considered yet. Neither are the requirements completely clear, nor are the technologies completely available. Several payment systems, with *ecash* as the most advanced, provide some sort of anonymity, but anonymous payments without anonymous communication does not make much sense. No project deals with the more general problems like anonymous credentials.
- Most systems assume a master-slave relation between seller's server and buyer's browser. The resulting asymmetry limits the complexity of protocols that can be performed in this model, and does not allow protocols between users (i.e., between two slaves without master).
- Most systems are limited to 2 parties. For instance, SSL supports a secure session between browser and server only. Integrating a secure connection to a third party like a "bank" in a payment system would be difficult.

- All projects that aim at prototype or product developments consider just on-line purchases, i.e., offer, order, payment and delivery. Multi-party problems (like auctioning) and fairness aspects (like contract signing, certified mail) are not considered yet.
- Most projects are US based. This means that their results are subject to US export control, i.e., they are not necessarily available outside the US. For instance, an SSL or SHTTP enabled browser developed in the US must not be exported unless the cryptographic algorithms are replaced by weak, i.e., breakable "export versions." Additionally, the law of some countries (e.g., France) does not allow to use products that support strong encryption of arbitrary data.

4 Objectives of *SEMPER*

The list of scenarios, actions, and security requirements in Section 2 already describe the working area of *SEMPER*. Within this area, the main objective of *SEMPER* is

> to develop, implement, trial and evaluate an open architecture for secure electronic commerce, especially taking into account multi-party security and privacy requirements.

Open Architecture for Electronic Commerce: *SEMPER* defines an *open* and *system independent* architecture for electronic commerce:

- The architecture is *independent* of specific hardware, operating systems, or networks.
- The architecture supports "plug-in" of new components, i.e., it is *independent* of specific service implementations, e.g., independent of the specific payment systems used in the trials; most payment systems can be "plugged-in."
- The architecture is *independent* of specific business applications. It supports any business application of electronic commerce that can be expressed in our model, i.e., as sequence of exchanges.
- The design process is *open* for public review. The *SEMPER* consortium has committed to publish all specifications, and appreciates security evaluations by third parties. The results of *SEMPER* will be used as input for standardisation.

Security: As in the physical marketplace, all participants have specific security requirements:

- Buyers often require to reliably authenticate the sellers they are dealing with. Note that it is easy to set up a WWW server and attach the name of a well known seller to it; even names that are already in use can be assigned; the highly fault-tolerant Internet tolerates such inconsistencies;
- Buyers might wish to browse anonymously through the catalogues of sellers, and if money and goods are exchanged fairly, identification of the buyer is not necessary at all.

- A seller does not want to deliver on-line goods without some guarantee of payment.
- In some scenarios, a seller might require specific credentials from a buyer.
- Buyer and seller might wish means for secure on-line payments, but certainly all parties — payer, payee and the financial services providers — do not want an increased risk compared to the physical marketplace.

Multi-party security means that the security requirements of all parties are considered individually, and that all security requirements of a party are guaranteed without forcing this party to trust other parties unreasonably. In particular, mutual trust between parties with conflicting interests like payer and payee in a payment is not assumed. Ideally, a party only has to trust itself and the jurisdiction and even the decision of a court may be verified.

In order to support the necessary degree of security, several cryptographic mechanisms must be applied. The architecture of *SEMPER* has to support

- for authentication: certification; credentials; non-repudiation of origin, submission, delivery; contract signing; fair exchange;
- electronic payment systems following different payment models, e.g., pre-paid cash like, credit card like, cheque like, money transfers;
- anonymous communication;
- copyright protection.

The Internet poses the strongest security challenges: It is completely open, without a central network security management, without any provisions for communication integrity, authenticity, or confidentiality. Even worse, the structure and openness of Internet makes life for attackers as easy as possible. For instance, it is a trivial task to check the traffic routed through a node controlled by an attacker for *telnet* or *ftp* passwords, or to send electronic mail under an arbitrary sender address via *smtp*. See, e.g., [ChBe 94] for a description of the most common security problems of the Internet. Thus, showing the feasibility of a secure and advanced electronic marketplace on the Internet proves feasibility for almost all other types of networks.

Trials: The architecture and services developed by the *SEMPER* consortium will be evaluated by means of trials. The first trial will be based on the Internet only, while later trials will use ATM-broadband networks.

The initial trials will be based on the minimum set of services that are necessary to secure the existing services of the 3 trial partners in *SEMPER*, namely

- EUROCOM (Athens), offering on-line multi-media training courses;
- FOGRA (Munich), offering several publications and on-line consulting;
- Otto-Versand (Hamburg), offering a small part of their mail-order catalogue.

Transfer / Exchange *of* → *for* ↓	Money	Signed document	Information
nothing (i.e., Transfer)	Payment	Certificate transfer etc.	Information transfer
Money	Fair money exchange	Fair payment with receipt	Fair purchase
Signed document	*Same as …*	Fair Contract Signing	Fair conditional access
Information	*… in upper …*	*… right half*	Fair information exchange

Fig. 2. Transfers and exchanges of primitive types

5 Initial Architecture

Model: The model of *SEMPER* describes business sessions in terms of sequences of *transfers* and *exchanges* similar to the *dialogues* of interactive EDI.

A *container* is the general data structure for what can be transferred and exchanged. It contains several *primitive types* together with their security attributes in a tree-like structure, namely

- signed documents, such as certificates, receipts, and signed statements;
- information, such as digital goods, information necessary to access a service (e.g., an address and password or a cryptographic key that protects a video stream), and information necessary to access physical goods;
- money.

A container can be structured according to a *template* which also defines the semantics of its contents, and which might be certified by a third party (e.g., like today’s standard contracts for apartment rentals with fields to fill in). The concept of templates is similar to the concept of messages in EDI. Each template clearly defines the meaning of the data contained in the fields of the template.

In a *transfer*, one party sends a container to one or more other parties. The sending party can define certain security requirements, such as confidentiality, anonymity, non-repudiation of origin. The sender receives an acknowledgement for each transfer, but this acknowledgement does not necessarily prove successful submission.

A *fair exchange* is an exchange of containers where two or more parties have the *assurance* that if they transfer something specific to the others, they will also receive something specific. Note that we require a *guarantee* of fairness. If no such guarantee is required, we can model such a conversation by several transfers.

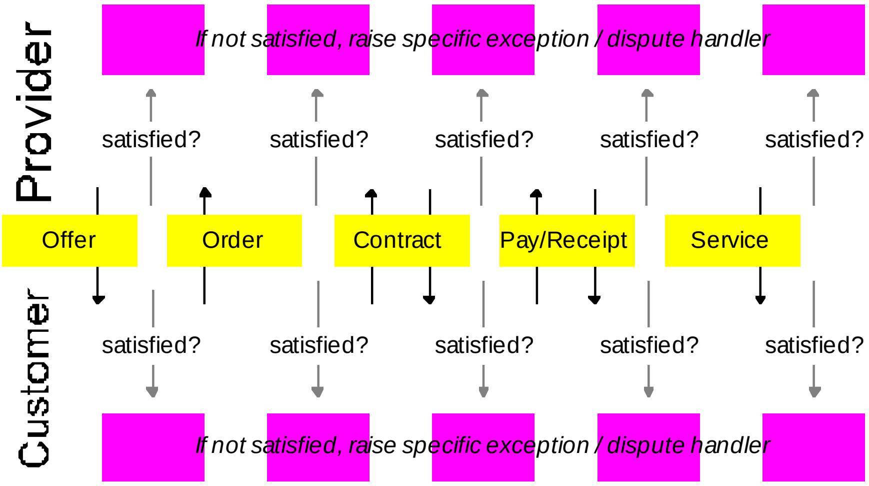
Provider
Customer
If not satisfied, raise specific exception / dispute handler
satisfied?
Offer
Order
Contract
Pay/Receipt
Service
satisfied?
If not satisfied, raise specific exception / dispute handler

Fig. 3. Example of a sequence of exchanges and transfers The protocol might enable other sequences as well, e.g., after "Contract" "Payment without Receipt" might also be enabled.

The actual sequence of transfers and exchanges in a business session can either be determined directly by the users, or it can be described by a protocol for such business sessions. Of course, a protocol may branch, i.e., allow more than one sequence.

Fig. 2 gives an overview of the possible exchanges of primitive types. Transfers are included as exchanges of “something” for “nothing.”

Obviously, the matrix of Fig. 2 is complete with respect to pairs, but there may be different security requirements in detail. The initial architecture of *SEMPER* is two-party centred. The same considerations can be applied to the multi-party case. For instance, more than two parties might wish to sign a joint contract, or one sender might want to send a certified mail to several recipients.

In the course of an ongoing business session, after each transfer or exchange, the parties are either

- *satisfied* and thus willing to proceed with a certain number of other transfers or exchanges or
- *dissatisfied*, in which case an *exception* or *dispute* handler is raised which might end up at a real court if all else fails.

Layers of *SEMPER*: The main activity of *SEMPER* during its first 6 months was the definition of an initial model and architecture, and the specification of a basic set of services.

The security architecture of *SEMPER* describes a layered structure in which the business applications are on the upper layer and services for secure commerce on the lower layers (see Fig. 4).

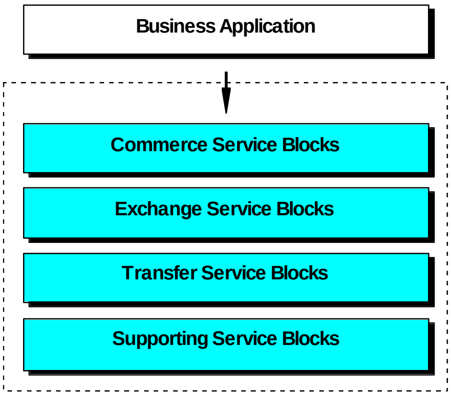

Fig. 4. Architecture of *SEMPER* — Overview

- The commerce layer offers high-level services for *business sessions* like “on-line purchase of information” or “registration with service provider”, and template management.
- The exchange layer supports *fair exchange* services.
- The transfer layer provides the *transfer* services for sending information.
- The supporting services are the usual cryptographic services, communication, archiving of data (keys, non-repudiation tokens, templates, audit), setting preferences and handling access-control, a trusted user interface which the user can enter or be shown sensitive information (TINGUIN: Trusted INteractive Graphical User INterface).

The architecture supports, but does not prescribe, the use of trusted hardware, like smartcards or electronic wallets. Commerce services, i.e., new scripts for business sessions, can be downloaded and added dynamically.

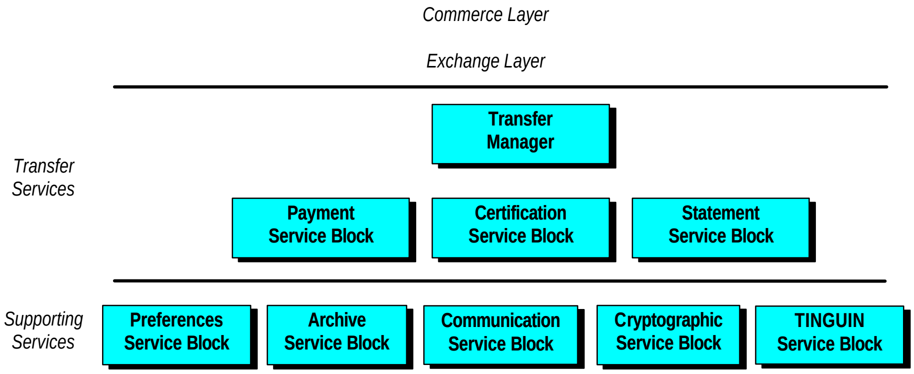

Fig. 5. Initial design of *SEMPER*

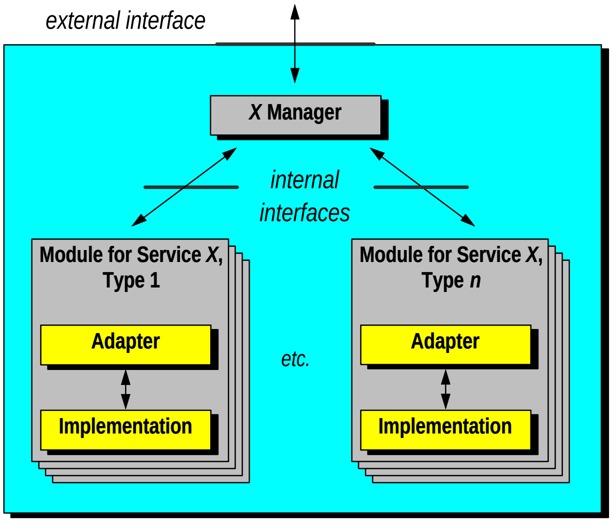

Fig. 6. Integration of different service modules in *SEMPER.* A payment manager, for example, may manage different payment systems like SET/iKP and e-cash.

The first actual design based on this architecture is summarised in Fig. 5. It supports transfer services and a fixed set of commerce services only. The functionality of the transfer layer is divided into the 3 fundamental blocks electronic payments, certification, and general statements which includes digital signatures.

Each service block in Fig. 5 provides a generic interface and allows to integrate different service modules that actually provide the service (see Fig. 6). For instance, the payment service block provides a generic "external" payment API that is independent of specific payment systems [APAW 96]. A concrete payment system can be integrated by providing an "adapter" mapping the concrete system's API to an "internal" API of the payment service block. Currently one internal API for account based systems (like *SET* or *iKP*) and one for cash-like payment systems (like *ecash)* are being designed.

The basic trials will use IBM's *iKP* and DigiCash's *ecash* for electronic payments, GMD's *SecuDE* toolkit for X.509 certificates, and crypto toolkits developed by *Cryptomathic* and *GMD* for statements. The system will be implemented in software only. Later versions might use *SET* instead of *iKP* (both implementing the same payment model).

Trust in Components: Naturally, without correctly working components, no security can be achieved:

- Software components may not behave as specified and, e.g., sign fake statements.
- The user-interface may display wrong amounts to pay, or questions to decide, so called masquerade attacks.

- Confidential user-input, such as credit-card numbers or PINs, may be stored and distributed over the network.
- Secret keys may be retrieved and misused.

Therefore, a user of *SEMPER* has to *believe* that their components and user-interface behave correctly and protect their security. We call this *trust* in components. Since *SEMPER* provides an open architecture, we cannot assume that all parties trust every component. However, trust of the parties involved can be increased by several measures:

- public design, implementation, and evaluation.
- an open architecture which allows to choose between different manufacturers;
- dedicated security modules.

In addition, each user will be able to decide whom and what to trust. If some components, such as specific payment systems, are not trusted, these components will be moderated by trusted components. For trusted user-interaction, *SEMPER* provides a local “Trusted Interactive Graphical User Interface” (TINGUIN; see Fig. 5) which is unambiguously distinguishable from the user-interface of the business application, and should be ideally implemented on a separate security module, e.g., a secure electronic wallet.

6 Summary

SEMPER is the first open architecture for multi-party secure electronic commerce. We described our view of electronic commerce, the existing technologies, the objectives of *SEMPER*, and the initial architecture. For more information see <http://www.semper.org>.

This work was partially supported by the ACTS Project AC026, *SEMPER*. However, it represents the view of the author. *SEMPER* is part of the Advanced Communication Technologies and Services (ACTS) research program established by the European Commission, DG XIII. This description is based on joint work of all partners of the consortium. It is a pleasure to thank all of them for their co-operation.

7 References

APAW 96 *J. Abad Peiro, N. Asokan, M. Waidner:* **Payment Manager;** SEMPER Activity Paper 212ZR054 <http://www.semper.org/info/>.

Atki 95 *R. Atkinson:* **Security Architecture for the Internet Protocol;** Internet RFC 1825, August 1995.

BBCM 94 *J.-P. Boly, A. Bosselaers, R. Cramer, R. Michelsen, S. Mjølsnes, F. Muller, T. Pedersen, B. Pfitzmann, P. de Rooij, B. Schoenmakers, M. Schunter, L. Vallée, M. Waidner:* **The ESPRIT Project CAFE - High Security Digital Payment Systems;** ESORICS '94, LNCS 875, Springer-Verlag, Berlin 1994, 217-230.

BGHH 95 *M. Bellare, J. A. Garay, R. Hauser, A. Herzberg, H. Krawczyk, M. Steiner, G. Tsudik, M. Waidner:* **iKP — A Family of Secure Electronic Payment Protocols; First** Usenix Workshop on Electronic Commerce, New York 1995.

CFGM95 *S. Crocker, N. Freed, J. Galvin, S. Murphy:* **MIME Object Security Services;** Internet RFC 1848, October 1995.

Chau 89 *D. Chaum:* **Privacy Protected Payments;** SMART CARD 2000, North-Holland, Amsterdam 1989, 69-93.

ChBe 94 *W. R. Cheswick, S. M. Bellovin:* **Firewalls and Internet Security — Repelling the Wily Hacker;** Addison-Wesley, Reading 1994.

GSPT 95 *D. K. Gifford, L. C. Stewart, A. C. Payne, G. W. Treese:* **Payment Switches for Open Networks;** IEEE COMPCON, March 1995.

HiEl 95 *K. Hickman, T. ElGamal:* **The SSL Protocol;** Internet Draft, June 1995.

JaWa 96 *P. Janson, M. Waidner:* **Electronic Payment Systems;** SI Informatik /Informatique 3/ (1995) 10-15; extended version accepted for: Datenschutz und Datensicherung DuD (1996).

Linn93 *J. Linn:* **Privacy Enhancement for Internet Electronic Mail;** Internet RFC 1421-24; February 1993.

MoKa 90 *R. Mori, M. Kawahara:* **Superdistribution — The Concept and the Architecture;** The Transactions of the IEICE; 73/7 (l990).

ReSc 95 *E. Rescorla, A. Schiffman:* **The Secure HyperText Transfer Protocol;** Internet Draft, July 1995.

Schn 96 *B. Schneier:* **Applied Cryptography;** John Wiley & Sons, 1994, 1995.

SET 96 *Mastercard, VISA:* **Secure Electronic Transactions;** Draft, June 26, 1996. (available from <http://www.mastercard.com/set/set.htm>)

SiTy 95 *M. Sirbu, J. D. Tygar:* **NetBill: An Internet Commerce System;** IEEE COMPCON, March 1995.