The SEMPER Framework for Secure Electronic Commerce[†]

Matthias Schunter[*]
Universität Saarbrücken
Institut für Informatik
Lehrstuhl für Kryptographie und Sicherheit
D-66123 Saarbrücken
Tel.: +49 (681) 302 5608
E-Mail: schunter@acm.org

Dr. Michael Waidner
IBM Zurich Research Laboratory
Säumerstr. 4
CH-8803 Rüschlikon
Tel.: +41 (1) 724 82 20
E-Mail: wmi@zurich.ibm.com

Dale Whinnett
Albert-Ludwigs-Universität Freiburg
Institut für Informatik und Gesellschaft
Abteilung Telematik
Friedrichstr. 50
D-79098 Freiburg
Tel.: +49 (761) 203 5700
E-Mail: dalew@iig.uni-freiburg.de

[†] [ScWW98; ScWW99] are preliminary versions of the this article.

[*] Versandanschrift für Korrekturen.

Keywords

SEMPER, Electronic Commerce, Computer Security, Electronic Payment, Digital Signature, Cryptography

www-Abstract

Beschreibung des integrierten und offenen *SEMPER* Frameworks für sicheren elektronischen Handel über das Internet sowie der Ergebnisse von Feldversuchen bei diversen europäischen Handelsunternehmen.

Abstract

The goal of the ACTS Project *SEMPER* (Secure Electronic Marketplace for Europe) has been to provide the first open and comprehensive framework for secure commerce over the Internet and other public information networks.

A prototype of the *SEMPER* Framework for Secure Electronic Commerce has been implemented in the Java programming language. It supports the payment systems Chipper, ecash™, Mandate, and SET.

This article describes the basic concepts of the SEMPER Framework for Secure Electronic Commerce, as well as experiences gained in the field trials of the SEMPER software.

Kernpunkte für das Management

Beschreibung des integrierten und offenen *SEMPER* Frameworks für sicheren elektronischen Handel über das Internet sowie der Ergebnisse von Feldversuchen bei diversen europäischen Handelsunternehmen.

- *SEMPER* modelliert Handel als gesteuerten Ablauf von fairen Austauschen und sicheren Lieferungen von Geschäftsobjekten.
- Das Framework hat einen Schichtenaufbau. Es werden Schichten für Geschäftsobjekte, Austausch und die Steuerung der Abläufe beschrieben.
- Ein Prototyp wurde in diversen Umgebungen getestet und durch Nutzerbefragungen evaluiert. Beispiele sind Versandhandel (Otto Versand, Hamburg), Beratungsdienstleistungen und Veröffentlichungen (FOGRA, München), oder Verkauf von Fortbildungen über das Internet (EUROCOM, Athen).

1. Introduction

A wide range of businesses are rapidly moving to explore the huge potential of networked information systems, in particular the Internet-based World-Wide Web. 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.

The goal of the ACTS project *SEMPER* (Secure Electronic Marketplace for Europe) has been to provide the first open and comprehensive solution for secure commerce over the Internet and other public information networks. The project started in September 1995 and ended in December 1998.

The members of the SEMPER consortium are *Commerzbank* (D), *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, F), *INTRACOM* (GR), *KPN Research* (NL), *Otto-Versand* (D), *r^3 security engineering / Entrust Technologies* (CH), *CNET* (F), *SINTEF* (N), *Stichting Mathematisch Centrum / CWI* (NL), *Universities of Dortmund, Freiburg, and Saarbrücken* (D). Sponsoring partners are *Banksys* (B), *Banque Generale du Luxembourg* (LU), and *Telekurs* (CH). The IBM Zurich Research Laboratory provided the technical leadership for the project.

1.1 Roles and Services in the Marketplace

The main purpose of an electronic marketplace is the same as that of a physical marketplace, i.e. to bring potential *buyers* and *sellers* together:

- Sellers offer their goods and buyers order them; this comprises a two-party negotiation, sometimes ending with an agreement, i.e., a contract.
- Sellers deliver their goods and buyers make payments; the result is a two-party (fair) exchange.

Buyers or sellers might be dissatisfied with the exchange, which means that a number of exception handlers and dispute handlers and, possibly, an arbiter are necessary.

In all these actions, the parties have specific *security requirements,* namely integrity, confidentiality, and availability.[1] Confidentiality includes anonymity, which is often a requirement for browsing catalogues, or for low-value purchases. Some examples of typical electronic commerce scenarios are:

- Mail-order Retailing: a retailer accepts electronic orders and payments, based on digital or conventional catalogues, and delivers physical goods.

---

[1] We assume that the reader has some basic knowledge of computer security. For an introduction in security as needed for electronic commerce we refer to [FoBa97], for an overview of cryptography to [MeOV97].

- On-line Purchase of Information and Subscriptions: similar to 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") through content hosting to billing services.
- Contract Signing: two or more parties exchange signed copies of the same statement.

Naturally, an open system for electronic commerce cannot be restricted to these scenarios. It should be easily configurable and extendible to a broad range of different scenarios.

1.2 What is New in SEMPER?

SEMPER is the first project that aims at the *complete* picture of secure electronic commerce, not just specific pieces (e.g., electronic payments [AJSW97]), specific scenarios (e.g., electronic on-line purchases), or specific products and protocols. The relationship of selected electronic commerce projects[2] to SEMPER is explained in Section 5.

SEMPER provides an *open framework* for electronic commerce. This includes a legal framework [Baum99; SEMP99b], as well as a technical framework which is described in this article (and in more detail in [SEMP99a; SEMP99b]).

The technical framework enables the integration of any protocol and product which provides the necessary services (see Section 3.6). Therefore, applications are not restricted to specific proprietary technology or specific protocols.

The prototype implementation of the SEMPER Framework uses existing technology for standard services like payments, public-key certification, and cryptographic services. Of course, this was not possible for the more advanced services like fair exchanges [AsSW97; AsSW98], specific certification services [Baum99], trust management [SEMP99b], dispute handling [AsHS98], and anonymous communication where we had to design our own solutions.

Another objective which distinguishes SEMPER from other projects is the concept of "multi-party security": Ideally, SEMPER users can ensure their own security with only minimal trust in other parties. This is done, e.g., by providing evidence for all critical actions, so that these actions can be disputed before an arbiter, should a fault or attack occurs. Multi-party security is primarily a question of protocols, not of service interfaces. This means that although SEMPER provides several multi-party secure protocols and includes all the necessary interfaces, it does not prevent the integration of less secure protocols and products if a user wishes to use them.

---

[2] An overview of electronic commerce projects can be found at <http://www.semper.org/sirene/outsideworld/ecommerce.html>.

1.3 Overview

First, a description is given of the SEMPER model of electronic commerce, which is based on a perception of commerce as a workflow of atomic transfers and fair exchanges of business items, such as electronic goods. This is reflected in the SEMPER Framework for Secure Electronic Commerce which is described in the next section. The SEMPER Framework is structured in layers. The lower layers provide the business items, the transfers and the fair exchanges. The higher layers provide generic workflows for the most common commerce scenarios together with the means to configure them according to the specific requirements of a particular user. This is followed by experiences with field-trials of the SEMPER prototype. The article concludes with a summary and a comparison of SEMPER with other more recent frameworks for electronic commerce.

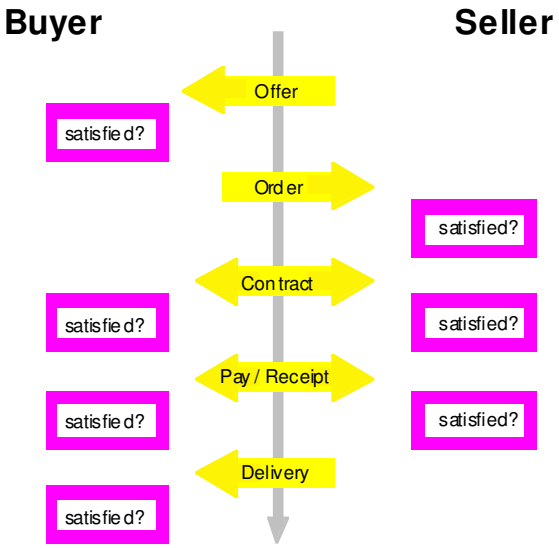

Figure 1 Electronic Commerce is Modelled as a Sequence of Transfers and Exchanges. Note that the protocol might enable other sequences as well, e.g., after "Contract" "Payment without Receipt" might also be enabled.

2. Model for Electronic Commerce

The SEMPER Framework described in this paper is based on a generic model for two-party electronic commerce. This model describes the flow of control, as well as actions and decisions related to commerce services.
The main idea of the model is to describe business scenarios in terms of sequences of *transfers* and *fair exchanges* of business items, with decisions based on the

success of these actions (see Figure 1). This model is similar to the *dialogues* of interactive EDI.

2.1 Atomic Actions: Transfers and Exchanges

The main interactive actions between two players are *transfers* and *fair exchanges*. In a *transfer*, one party sends business items to one or more other parties. The sending and receiving parties can define certain security requirements, such as confidentiality, anonymity, or non-repudiation of origin. Business items which can be transferred or exchanged include:

- valuable information, such as the result of consultancy, or program and video data,
- statements, such as signed documents or certificates, and
- money, such as credit-card, cash, or bank transfer payments.

A *fair exchange* is a simultaneous exchange of packages of business items, typically between two parties. The parties have the *assurance* that their packages are sent if, and only if, the peer entity sends its package as expected. Either both packages are exchanged or none. If no fairness guarantee is required, such an exchange can be modelled as two transfers.

As an example of applying fair exchange, imagine Bob has requested consultancy services from Alice, e.g., a translation, or legal expertise. Alice wants to deliver to Bob a file containing the report. The file represents a high value, so Alice wants a receipt if Bob receives the file. Bob, on the other hand, only wants to issue a receipt if he receives the file.

Figure 2 gives an overview of all possible exchanges of these primitive types. Transfers are included as exchanges of "something" for "nothing."

Transfer / Exchange *of* → *for* ↓	Money	Signature	Information
Nothing (i.e., Transfer)	Payment	Signature transfer	Information transfer
Money	Fair money exchange	Fair payment with receipt	Fair purchase
Signature	*Same as ...*	Fair contract signing	Certified mail
Information	*... in upper ...*	*... right half*	Fair information exchange

Figure 2 Transfers and exchanges of primitive types.

2.2 Electronic Commerce: Sequence of Exchanges

Using transfers and exchanges, a typical business scenario is modelled as a sequence of exchanges with user-interaction and local decisions between successive exchanges (see Figure 1).

In the course of an ongoing business transaction, 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 is raised which might end up at a real court if all else fails.

This local decision depends on the success of the previous exchanges, the items received, and, possibly, on user-input. After each round, a local decision is made whether to proceed and, if so, how. These sequences are similar to the workflows of ordinary business transactions.

3. The SEMPER Framework

In the following, the main services of the SEMPER Framework are described. After a short overview, the layers are described in more detail starting with the lowest layer. For a complete description of SEMPER, please refer to the final

documentation of the design [SEMP99a] and the final report [SEMP99b] which contain a more comprehensive presentation.

3.1 Overview

The SEMPER Framework (see Figure 3) is structured in layers. The lowest layer handles low-level security primitives and other *supporting services*, whereas the highest layer manages commerce issues only:

- The supporting services are the usual cryptographic services, communication, secure archiving of data (keys, non-repudiation tokens, audit trail), setting preferences, access control, and the trusted user interface "TINGUIN" (Trusted INteractive Graphical User INterface). In addition, secure communication services, such as confidential, authenticated, or anonymous communication are provided by this layer.
- The transfer layer provides services for transferring business items. This includes transfer-related security services, such as non-repudiation of origin.
- The exchange layer supports the fair exchange of business items.
- The commerce layer provides the local business sequences of the model which are executed locally by each player. Examples are sequences like "mail-order retailing," "on-line purchase of information," or "registration with service provider." The commerce layer can be configured by downloading new services or extending existing ones.

On top of these layers are so-called Business Applications. Business Applications are neither a layer nor a part of SEMPER, but the name used to refer to any application that uses the SEMPER services. As Business Applications can be implemented outside SEMPER, they are a priori untrusted and not allowed to perform security-critical actions without user authorisation.

Note that the security guaranteed by the layers gets broader and more abstract towards the top. The transfer layer only guarantees secure transmission, whereas the commerce layer guarantees security of a whole commerce scenario.

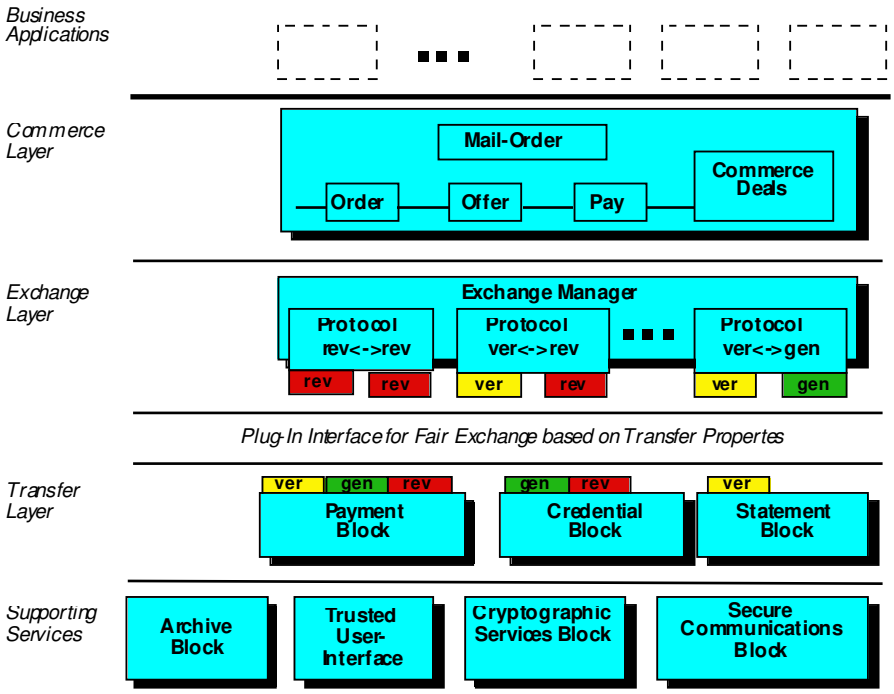

Figure 3 The SEMPER Framework for Secure Electronic Commerce.

3.2 Supporting Services

Most of the *supporting services* illustrated above are not specific to electronic commerce. Apart from the trusted user-interface "TINGUIN", none of them was a primary objective of SEMPER.

The goal of the TINGUIN is to solve a common problem of Internet-based electronic commerce. In current electronic commerce sessions over the Internet the buyer typically interacts with the seller through his WWW-browser. As the HTML-pages presented in the browser are under the control of the Web-server (i.e., the seller), this exposes the buyer to considerable risks. A message on such an HTML page, stating that an offer made by the seller has been signed correctly, cannot be trusted. A dishonest seller could simply put this message on the HTML-page without validating the signed offer (in fact the seller may not have signed the offer at all). Another problem of using the browser is that any input supplied by the user may be read by the seller. This could lead to serious problems if, for example, the buyer has to enter a PIN code, or password, in order to use an electronic purse, or make a signature.

In the SEMPER prototype, the TINGUIN is a secure and consistent user-interface in a separate window, which can be clearly distinguished from other (browser) windows. All normal service blocks of SEMPER interact with the TINGUIN block that provides this window. They use common "look-and-feel" elements

which the TINGUIN block provides. All security-critical input and output for a user should be made via the TINGUIN. Together with the other SEMPER services, this enables the user to be certain that the output of the TINGUIN has been verified locally and to be warned that any input via the TINGUIN could have commercial consequences.

Naturally, this interface has to be controlled by the user's own device and it must be difficult to simulate. The user must always be able to recognise the genuine TINGUIN. Special attention has to be given to the ergonomics of the TINGUIN because any misunderstanding on the part of the user could result in a breach of security. We refer to [Webe97] for a discussion of this kind of requirements in the context of SEMPER.

3.3 Transfer Services

The *transfer layer* provides services for packaging and trading business items. The basic items are electronic payments, credentials, and general statements which include digital signatures and data. These business items can be bundled into tree-like packages. The security attributes attached to each transfer determine the level of security which is required for the transfer or exchange of the item.

Each type of business item is managed by a separate manager which provides for unified services integrating existing implementations as described in Section 3.6.

Furthermore, the transfer services define the interfaces of the exchange-enabling properties used by the fair exchange protocols in Section 3.4.

3.4 Exchange Services

The *fair exchange services* developed by SEMPER are *optimistic* and *generic*. Optimistic means that a third party is only used if a fault occurs, in order to restore fairness [AsSW97; AsSW98]. Generic means that the fair exchange protocols can be used to exchange arbitrary business items.

The reason for developing *generic* fair exchange, instead of using fair exchanges specific for each table cell in Figure 2, is that, whereas a fair exchange protocol "payment for receipt" may work with one payment scheme, it may not work with another. The result would be that instead of having a maximum of nine different protocols, each new implementation of an item could require new fair exchange protocols. Furthermore, exchanging packages of items would require specific fair exchange protocols for any fixed combination of items to be exchanged. Thus, for a given number of n different kinds of electronic items n^2 different fair exchange protocols would be required if one only wanted to exchange one item for another. Furthermore, adding a new item to be exchanged (such as a new payment module) would mean adding another $n+1$ fair exchange protocols.

In order to achieve the desired independence of the actual items to be exchanged, a minimal set of "exchange-enabling properties" were defined. These have to be implemented by two transfers of business items in order to be used in a fair exchange protocol:

- *External Verifiability*: The third party has the ability to check whether a transfer was successful or not. This can be achieved, e.g., by sending, or re-sending, the message via the third party.
- *Revocability*: The third party is able to undo a transfer (e.g., revoking a credit-card payment).
- *Generateability*: The third party is able to remake a transfer (e.g., signing a replacement receipt).

Based on transfers with these exchange-enabling properties for the two items to be exchanged, the so-called exchange manager negotiates with its peer which generic fair exchange protocol will be used.

An example of such a generic fair exchange protocol, as it is used in SEMPER, is depicted in Figure 4. The protocol can be used to exchange any generateable item for any externally verifiable item. It is similar to the protocol described in [AsSW97]: the basic idea is that the participants first agree on the exchange. If they agree, i.e., the descriptions of the item expected are matched by the item offered, the responder transfers its item. If the item matches the expectation of the originator, the originator then sends its item as well. If the originator acts incorrectly and does not send its item, the responder complains to the third party which then produces an equivalent replacement for the item (this can be done since the second item was generateable) if and only if the first transfer was successful (this can be done using the external verifiability provided by the first item). A more detailed description can be found in [SEMP99b].

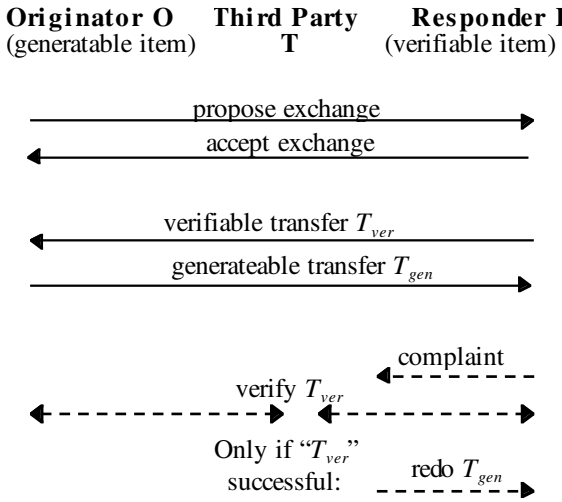

Figure 4: Fair Exchange of Generateable and Externally Verifiable Transfers.

Following the same pattern, other protocols which guarantee fairness can be built as long as one of the items provides generateability and the other external verifiability, or if both offer revocability.

3.5 Commerce Services

The *commerce layer* implements the flow of control in the SEMPER model using the transfer and exchange service for interactions with the business partner, and the supporting services for user interaction and persistent storage. It also performs the trust management and access control necessary for downloading certified commerce services.

In order to provide overall security, the commerce layer sets up security contexts, called "deals", which provide secure communication and signal certain commerce security attributes to all ongoing protocols. An example of a security service signalled by a context is "anonymity" which has to be guaranteed by all blocks participating in the deal.

Inside these deals, the commerce layer then runs those sequences of the model which directly implement the protocols of business scenarios, e.g., how specific merchants, or types of merchants, handle customer registration and the offering, ordering, payment, and delivery of goods. It implements the local flow of control, i.e., the enabled sequences of exchanges, of the electronic commerce model for each player separately. A set of client and server commerce services check each other. They resemble an automatically verified electronic equivalent of the "terms of business" of the players. The commerce layer not only offers entire commerce

protocols, but also building blocks that may be of more general use, in particular services to manage and fill out standardised order forms.

Note that the commerce layer services are usually structured in an hierarchy. An offer/order or a payment/delivery building block can be used by a generic mail-order service, as well as a service for selling database access. Thus, in principle, the commerce layer should provide transfer- and exchange-based workflows implementing all common commerce scenarios.

Since one cannot fix the set of required services in advance, the commerce layer includes services for the secure downloading of new services. This allows customers to participate in business scenarios they never encountered before. Since arbitrary terms of business may be implemented in a new commerce service, a new commerce service may not guarantee a sufficient degree of security. Therefore, downloading must be supplemented by evaluation and certification of downloadable services, as well as proper access control.

3.6 Openness with Service Managers and Service Modules

So far, the description has been limited to the generic services of each building block without looking into internal details of them. In the following, the abstract concept which guarantees openness of the implementation will be described (see [AASW98] for a detailed design based on this abstract concept).

Generic services are provided by so-called *service managers* and several *service modules* (see Figure 5). The union of a manager and its modules is called a *service block*.

The *service block* provides the generic, unified service, e.g., service = “payment”, which includes services for managing modules.

The generic service is based on a model of the service, which should cover a broad range of protocols implementing this service, i.e., there are generic interfaces for a whole class of services.

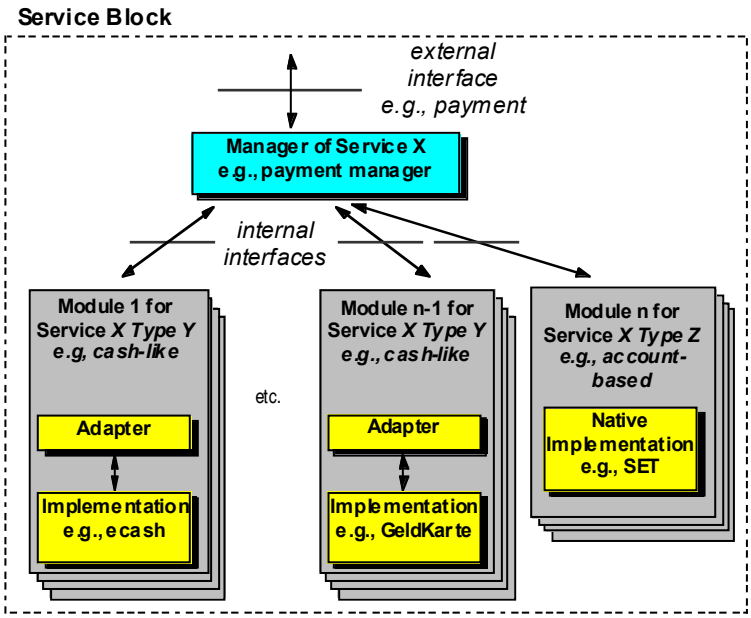

Figure 5: Open Services with Managers, Modules and Adapters[3].

For instance, the external interface of the payment manager is based on a *generic payment service* that covers all kinds of *payment protocols* (or at least a small number of generic payment services, one for each payment model, such as *account-based* or *cash-like*). Note that this does *not* mean that only a few specific payment protocols can be supported, but that the interface definition is so general that *any* reasonable payment protocol can be accessed via that interface. If, e.g., company *XYZ* comes up with a new payment system *abc*, all they have to do in order to link *abc* into the architecture is to map the service interface of *abc* to the internal payment interface. This guarantees the desired openness of the marketplace.

A *service manager* provides a common interface to several modules together with methods for negotiation and selection of an appropriate module.

A *service module* corresponds to a protocol implementing the service, i.e., it more or less implements one entity of such a protocol. Its interface to the service manager is called an *internal interface*[4] of this service.

Having several modules per service allows different protocol implementations by different manufacturers. Service modules are said to be of the same type if their behaviour at the internal interface is the same. Examples of types of internal payment interfaces are “cash-like”, and “account-based”. Modules could be “SET”,

---

[3] The last module is an example of a module written specifically for SEMPER and needs no adapter.

[4] This is also often called a Service Provider Interface (SPI).

and “e-cash”, where “SET” implements the account-based model and “e-cash” the cash-like model.

As *SEMPER* aims to build on existing products as far as possible, it cannot be assumed that they all originally fit the same internal interfaces. Therefore, the interface of an existing implementation is enhanced by a *service adapter* so that the resulting module supports the required service.

The *SEMPER* architecture describes a fixed set of service managers. The set of service modules is not fixed, i.e., service modules can be dynamically attached to managers.

4. The SEMPER Trials

The SEMPER software prototype was put to the test by conducting a series of end-user trials which began in July 1997 and continued until the conclusion of the project in December 1998. The implementation used in the trials included a basic set of security services for electronic commerce, e.g., secure identification of business partner, secure offers, orders, payment and delivery. The trials were conducted in four phases using seven trial sites representing various business contexts which included mail order, tele-training, a literature service, database access and image distribution (see Figure 6). The players involved in the tests included buyers (the trial participants), sellers (the business applications implemented for the trials, e.g. a merchant server using the SEMPER software), a registration and certification authority (provided by the SEMPER partner GMD) and financial institutions (a test bank at the GMD, a SET payment gateway at the IBM Zurich Research Laboratory and at the Commerzbank in Frankfurt).

Trial Characteristics	*EUR*	*FOG*	*FRE Basic*	*Otto*	*Acril*	*Acti*	*OPL*	*FRE SME*
Trial Phase	I		II	III				IV
secure identification of business partner	√	√	√	√	√	√	√	√
digitally signed offer	√	√	√	√	√	√	√	√
digitally signed order	√	√	√	√	√	√	√	√
generic purse (test payment system)	√	√	√	√	√	√	√	√
digital goods delivered on-line	√	√	√		√		√	√
real goods delivered off-line				√			√	√
web pages encrypted/sent via SEMPER					√	√	√	√
credit card data transmitted via SSL					√	√		√
SET payment protocol				√		√	√	√
encrypted credit card data via SEMPER							√	√
stored value - chipcard and user device							√	√
real credit card payment				√			√	√

Figure 6: Essential characteristics implemented in trials

The addition of new payment options as the trials progressed served to illustrate the extendibility of the SEMPER architecture. In the early trials a generic test purse was implemented. This was followed by the addition of a credit card purse and stored value payments using a chip card and, finally, by integration of the SET payment protocol.

During the course of the trials more than 70 persons were able to experiment with the SEMPER prototype. Their comments and criticisms were collected by means of questionnaires and personal interviews.

4.1 Trial Services

The security attributes implemented for the Basic and SME trials were message authentication, message integrity and message confidentiality. The following secure services were available:

- *Registration/certification:* The participant obtained a password (registration key) from the trial organiser. The GMD, acting as Registration and Certifica-

tion Authority (RA/CA) received a list of trial participants with the assigned registration keys. A copy of the registration was stored locally at the RA/CA, i.e. the GMD, and the certification procedure was then activated automatically. The issued certificate was stored locally at the GMD and a copy sent to the trial participant. The participant checked the incoming certificate and stored it in his/her archive.

- *Offer:* the offer service provided the necessary mechanisms for access of product catalogues stored either on flat files or various relational databases. In addition, specific software was provided for the access/link of legacy systems (e.g., Otto Versand's BTX system).
- *Order*: the order service provided pre-defined forms which trial participants could use to make a specific order/purchase of an item, or items. These forms were designed according to the needs of the individual service providers.
- *Payment*: in the Basic Trial two payment protocols were abstracted and integrated in the common payment framework (i.e. the payment manager). The SET and e-cash protocols. In the SME trial three forms of payment were offered: credit card payment using SSL (in France) and the SEMPER credit card purse (in Holland); stored value payments using a smart card and the Telechipper®; the SET purse, with a test acquirer (IBM Zurich) and real acquirer (Commerzbank, Frankfurt).

4.2 Basic Trials

During the Basic Trials the *SEMPER* prototype was tested using trial sites provided by two members of the consortium, Eurocom and Fogra. The Eurocom site, based in Athens (GR), offers distance learning services. Eurocom intends to use *SEMPER* to enable students to browse their offering of courses, register and pay on-line and, subsequently, gain on-line access to the selected course presentation, notes, and examinations. Fogra, a research institute for the printing industry, located in Munich (D), offers its customers on-line ordering and delivery of documents and software. It also sees an opportunity for on-line consultancy.

Eurocom conducted its trials as part of a seminar, "Conducting Business Over the Internet", offered to SME employees. Fogra conducted trials at the IMPRINTA trade fair and with five of their customers. Although the functionality and flexibility of the *SEMPER* architecture was appreciated by the Eurocom and Fogra trial participants, the state of the user interface was considered to be insufficiently developed for the ease of use to which non-specialists are accustomed. As a result, a new round of supervised trials, with participants selected on the basis of their networking experience, was conducted. Fogra's trial site (server and business application) and an improved user interface were used for these trials which were conducted at the Institute for Computer Science and Social Studies at the University of Freiburg (D). The trial participants, who had all used the Internet for 3 years or more and had a good awareness of security issues, subjected the prototype to particularly thorough testing, checking, for example, the software's response to incorrect input (seed too short, incorrect password entry, attempting to obtain a second certificate from the CA, attempting to continue without inputting the re-

quested information, rejecting offers, etc.) and provided valuable input for further refinements of the prototype.

The Trustworthy INteractive Graphical User INterface, *TINGUIN*, where all security relevant communications take place, is the visible and vital link between the user and the *SEMPER* software, as a result, it was subject to particular scrutiny. The credibility of the test for the participants was enhanced by the fact that it was possible for them to check the DOS (Java) window at all times during the test (and many did so). This ensured them that the test was actually *live*, i.e. that they were really exchanging certificates and protected information with the CA and the bank and website servers.

During the initialisation of the SEMPER software special attention was paid to the participants' understanding of the actions they were taking, as well as those factors which influenced their ability to successfully complete the process. The trial was conducted at various times of the day and included the weekend, as well as working days. This ensured that the on-line registration and the use of SEMPER in a purchase situation were subject to the variety of conditions currently present in the Internet, e.g., varying connection speeds, loss of connectivity, etc. The business context also allowed participants to experience the flexibility which electronic commerce offers in respect of being able to conduct business at any time of day, from the office, or home.

4.3 SME Trials

The SME trials were based on four locations, primarily using *semperised* websites of companies which were not members of the consortium (see Figure 7). The only exception to this was the Otto Versand (mail order company) trial which was included in the SME trials because of its late start and also because it used the same improved version of the trial code.

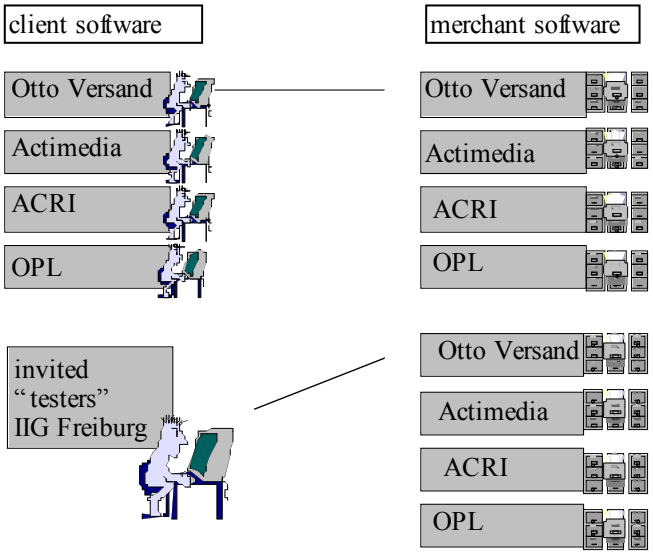

Figure 7: SME Trial Sites and Participants

Two SME trial sites were operated with companies in Sophia-Antipolis (F), Actimedia and Acri, and supported by the SEMPER partner IBM France. Actimedia sells French language CD ROMs and sees potential in the WWW for reaching a customer base throughout the world. For them the Web represents a niche for selling products to the French speaking population (and persons interested in French culture and language). This site consists of a virtual shop where CD ROM titles can be selected, placed in a shopping cart, ordered and paid for. It is aimed at private consumers living outside France. In contrast, the second French site, Acri, has a customer base of large organisations. For Acri, the potential for offering their customers tailor-made solutions and distributing highly confidential information in a secure environment was the reason for participating in the SEMPER trials. The company offers on-line simulation using fluid mechanics and the capability to mark-up aerial photographs. By allowing customers to select (and pay for) a segment of a larger representation, their services become more cost-effective and, therefore, accessible to a broader customer base.

The Oilfield Publications Limited (OPL) trial site operated from the Netherlands was supported by the SEMPER partner KPN Research. The OPL shop consists of a catalogue of books, maps and CD-ROM’s for the offshore oil & gas industry. The catalogue can be searched by users and interesting items can be put into a shopping cart. In addition to the shop, an on-line database was also offered for direct searching by users. The database contains circa 300 records, or fact sheets, related to mobile production units for gas & oil at sea. The information was delivered on-line via a secure SEMPER channel straight from the OPL database at the server. A unique aspect of the OPL implementation of SEMPER was the use of the Telechipper®, a user device with a smart card reader, keyboard and display, which was attached to the trial participant’s PC. This enabled the stored value

payments to be made securely via the SEMPER software. Trial participants accessing the OPL trial from Germany were also able to test this form of payment. In the OPL trial it was possible to make simulated or real purchases.

For its trial Otto Versand, the largest mail order company in Germany, developed an Internet presentation based on a database containing the data of 13.000 articles, including pictures and search criteria. From this database the HTML pages for the customers are created dynamically. The customer can either search through the presentation where goods are presented in shops or use a search engine by inputting specific criteria, e.g., size, article, colour, price range, etc. The set of matching articles can be viewed, browsed and ordered on the screen. Before an order is placed it is possible to obtain information regarding availability and delivery times for the goods in the virtual shopping cart. In the Otto trial there was no potential for tests using simulated purchases. This meant that a real payment system had to be used. The selected form of payment for the trials was credit cards using the SET payment protocol. This involved obtaining and using real SET merchant and customer certificates and having real payments cleared by a German bank which required lengthy and complicated negotiations. As a result, at the project's conclusion, it had only been possible to conduct a small number of internal tests. The first real SET transactions via the Otto website were finally made in January of 1999.

The final phase of trials, conducted at the University of Freiburg, offered participants the opportunity to use SEMPER from one computer to access all of the trial sites and services described above.

4.4 Trial Participants' Reactions to SEMPER

It is possible to identify two distinct groups among the persons who experimented with SEMPER, what can be referred to as "basic users" and "experienced users". The extent to which the user should be involved in configuring the software or activating/deactivating security options was clearly the most distinguishing point in the reactions of these two types of users. So-called "power users" want a high level of information and control, but they are currently in the minority. "Basis users" want a patent security solution which they can trust on the basis of outside expert opinion (e.g., certification by a recognised organisation, or standards body) and which requires as little personal intervention as possible. The "basic user" doesn't have enough background in security technologies to make informed choices. They suggested a help function, or links with supporting information, in particular, regarding the types of encryption keys offered, the security importance of key length, recommendations for selecting a good password, what the seed was needed for and tips about how to enter it, etc. "Experienced users", for example, found the fact that their encryption keys were generated locally and only the public key sent over the Internet to be "a very valuable asset of the software", whereas the "basic user" was indifferent to this, or unaware that this was an important security feature. In addition, the more experienced users wanted more information about key storage and the ability to refer to key related information.

Essential in e-commerce tool	*%*	*SEMPER*
secure payment	93	yes
ease of installation and maintenance	85	no
data privacy	81	yes
ease of use	80	no
signed offers/orders	76	yes
encrypted data transfer	73	yes
choice of payment options	60	yes
record-keeping	57	yes

Figure 8: Questionnaire Results: Eight characteristics rated as essential for an e-commerce tool

Users stressed that they want the same, if not more, flexibility in electronic commerce as they have in traditional markets and this includes selecting a form of payment which suits the particular purchase situation. During the SME trials the participants consciously applied different criteria to different purchase situations, using the stored value card and Telechipper®, for example, for low value database queries which were delivered on-line and credit card or other account-based payment for higher value goods which would be delivered by traditional means. In this respect, the flexibility of the SEMPER architecture to incorporate new payment systems, as they develop or are required, was viewed very positively.

The trial participants also viewed archiving and transaction browsing as essential components of an electronic commerce tool. The transaction archive was compared to an electronic form of bank or credit card statement, i.e. useful for maintaining an overview, but also necessary for collecting evidence of transactions. All users required legally binding receipts and the fact that receipts were not provided in the trials was viewed as a major deficit in the prototype implementation.

The concept of having a single unified user interface where all security relevant actions take place, which was realised with the TINGUIN, was viewed as an excellent approach. Users felt that it is essential to have a clearly defined area of interaction with the underlying systems and services in a security tool. Nearly all trial participants would prefer to keep this interface separate from the browser and a number were in favour of emphasising the personal link to the software more strongly by having the user's name clearly visible in the TINGUIN at all times. They felt this would increase user awareness of the significance of the actions supported by the software, e.g. digitally signing documents, or activating electronic payment, and also encourage them to protect themselves by protecting access to "their" SEMPER software.

The most heavily criticised "missing feature" in the TINGUIN was the lack of a status bar, or some indication that connectivity had not been interrupted, that the system was actively doing something. The content of messages displayed in the TINGUIN and the format in which information is presented was also criticised, in particular by more experienced users who wanted more precise information regarding each step. They emphasised that the content of the messages displayed must

make it absolutely clear to the user what actions are being carried out, which actions have been successfully completed and which steps remain.

As SEMPER aims to provide multi-party security for the entire communication process, of which payment is only one aspect, participants were asked to comment on perceived additional benefits of this approach. The secure identification of the communication partner and the encrypted exchange of information were cited as advantages whenever it was necessary to transmit personal data of any type. In addition, users wanted to be informed that this was taking place. Although there might be no risk of financial loss, they perceived the risk of a loss of privacy and possible annoyance, for example, unsolicited advertising. More important advantages were seen in the ability to obtain and send digitally signed documents. Trial participants saw potential in SEMPER for combining the advantages the Internet offers to react quickly to a broad range of offers with the traditional requirement for obtaining legally binding proof of these transactions.

One of the most valuable aspects of SEMPER architecture was seen to be the fact that it provides options, i.e. is not a static solution. This was seen to reflect the situation in traditional commerce where requirements and procedures vary from one business context to another. Factors such as the value of a purchase, whether the business partners have an established relationship or are doing business for the first time, all influence the way the business partners choose to conduct a transaction. In the trials this flexibility was mainly demonstrated by a range of payment options, but at least some of the trial participants could visualise this flexibility being transferred to other options, such as being able to accept different types of certificates, or apply different strengths of encryption depending on the business context.

The main user requirements for an electronic commerce tool reported by trial participants are illustrated in Figure 8. Most are provided by SEMPER, but not yet in a user-friendly form.

Users who had already used public key technologies for on-line banking, or the exchange of secure email, saw the advantage of SEMPER in the fact that the existing technologies are combined in one convenient tool which could be applied to the whole business process and would also incorporate the archiving of transactions. They also felt that a software package which combines various existing technologies, which are difficult for the non-technical user to implement himself, would result in increased dissemination and use of security technology, thus offering more opportunities for use.

> "The advantage of SEMPER is that these tools are incorporated into one tool that I can use for doing business. Its possible to assign meaningful roles to the various tools. I can say here is a databank, with goods and offers in it, and I can abstract an offer from it, digitally sign it and send it over as a container and that's more than just PGP and RSA and emails."

5. Summary

In 1995 the SEMPER consortium started with the objective to develop the first open and comprehensive framework for secure commerce over the Internet. We believe that SEMPER achieved this goal, and actually it is still the only project that aimed at securing electronic commerce as a whole.

The SEMPER framework turned out to be a valuable tool for understanding and implementing electronic commerce. An indication for the quality of the concepts is that only minor changes to the framework were necessary during the course of the project.

The design of some blocks, such as the payment block [AASW98], is very detailed and extended the state of the art. For some layers, e.g., the commerce layer, and specific topics, e.g., visualisation of security (TINGUIN), SEMPER has very promising approaches and results, but clearly more work is required [SEMP99b; Waid98]. For some blocks, e.g., the cryptographic services block [SEMP99a], one would probably replace the SEMPER designs by now existing standards.

The prototype uses existing technology as much as possible. But for some services no suitable solutions were available. This resulted in the development of new and innovative protocols, in particular for fair exchange services as described earlier [AsSW97; AsSW98], for specific certification services [Baum99], and for the support of dispute handling in electronic payment systems [AsHS98].

SEMPER also produced a legal framework and model contract for electronic commerce [SEMP99b].

Since 1995 several other, related electronic commerce framework projects started.

- Some projects developed *specific service frameworks* that correspond to the blocks on the lower layers of SEMPER, e.g., crypto [Inte97; PKCS97] and payment [DBGK98]. Experience has shown that for a commercial version of SEMPER one would use such specific frameworks for the supporting services, while one would use the SEMPER designs for the upper blocks, such as payments and fair exchanges.
- Some projects produced *implementation architectures*, e.g., the *Java Commerce Client (JCC)*[5]. SEMPER focused on the *service architecture*, and thus these approaches are rather complementary to than competing with SEMPER.
- Several projects investigated specific business scenarios, such as the *Open Trading Protocol (OTP)*[6] and the *Open Buying on the Internet (OBI)*[7] protocol, or developed business process frameworks, e.g., *XML/EDI*[8]. The results of these projects could naturally extend the commerce layer of SEMPER.

---

5 http://java.sun.com/commerce

6 http://www.otp.org

7 http://www.openbuy.org

8 http://www. xmledi .net

- Recently some projects started that aim at more general frameworks for electronic commerce. For instance, the *eCo Framework Project*[9] of CommerceNet plans to develop a framework for interoperability among XML-based e-commerce applications. None of these projects targets the specific security aspects of electronic commerce, i.e., we feel they would gain a lot by keeping the security-oriented SEMPER Framework in mind.

Since a couple of years electronic commerce has been a "hot topic" in economics and computer science. Neverthless, several security problems in electronic commerce are not sufficiently investigated yet [Waid98; SEMP99b].

The most urgent open problem is that of the security of the user's computer: At least one end of most electronic commerce transactions is handled by a personal computer with a standard operating system. Past experience has shown that these systems are notoriously insecure: they have severe security holes and are vulnerable to Trojan horse attacks. So far this has been no serious impediment for electronic commerce as criminals had probably simpler ways to make money. But the more commerce transactions are performed electronically, the more attractive becomes this fraud channel. We are convinced that more R&D work is required on the development of operating systems sufficiently secure for commercial transactions, on tamper-resistant components, and on provably secure security protocols.

6. Acknowledgements

This work was partially supported by the ACTS Project AC026, *SEMPER*. However, it represents the view of the authors only. *SEMPER* is part of the Advanced Communication Technologies and Services (ACTS) research program established by the European Commission, DG XIII. The SEMPER deliverables can be obtained at the SEMPER homepage <www.semper.org>.

We would like to thank the anonymous reviewers, Birgit Pfitzmann, Michael Steiner, Max Schmidt, and Lothar Fritsch for valuable comments which helped us to improve the paper.

References

[AASW98]

Abad-Peiro, L.; Asokan, N.; Steiner, M.; Waidner, M.: Designing a Generic Payment Service. In: IBM System Journal 37/1 (1998) 72-88.

[AJSW97]

Asokan, N.; Janson, P.; Steiner, M.; Waidner, M.: The State of the Art in Electronic Payment Systems. In: IEEE Computer 30/9 (1997) 28-35.

[AsSW97]

Asokan, N.; Schunter, M.; Waidner, M.: Optimistic Protocols for Fair

[9] http://www.commerce.net/projects/currentprojects/eco/

Exchange. In: 4th ACM Conference on Computer and Communications Security, Zürich, April 1997, 6-17.

[AsSW98]

Asokan, N.; Shoup, V.; Waidner, M.: Asynchronous Protocols for Optimistic Fair Exchange. In: 1998 IEEE Symposium on Research in Security and Privacy, IEEE Computer Society Press, Los Alamitos 1998, 86-99.

[AsHS98]

Asokan, N.; Herreweghen, E. van; Steiner, M.: Towards a framework for handling disputes in payment systems. In: 3rd USENIX Workshop on Electronic Commerce, 1998.

[Baum99]

Baum-Waidner, B.: Haftungsbeschränkung der digitalen Signatur durch einen Commitment Service. In: Workshop "Sicherheitsinfrastrukturen", Technische Universität Hamburg-Harburg, 9. - 10. März 1999; Vieweg-Verlag 1999.

[DBGK98]

Daswani, N.; Boneh, D.; Garcia-Molina, H.; Ketchpel, S.; Paepcke, A.: A Generalized Digital Wallet Architecture. Stanford University, Computer Science Department, 1998.

[FoBa97]

Ford, W.; Baum, M.: Secure Electronic Commerce; Prentice Hall, Upper Saddle River 1997.

[Inte97]

Intel: Common Data Security Architecture (CDSA); Version 27.10.1997.

[MeOV97]

Menezes, A.; Oorschot, P. van; Vanstone, S.: Handbook of Applied Cryptography; CRC Press, Boca Raton 1997.

[PKCS97]

RSA Data Security, Inc.: PKCS #11: Cryptographic Token Interface Standard, Version 2.0; 100 Marine Parkway, Suite 500, Redwood City, CA 94065, USA, April 15, 1997.

[ScWW98]

Schunter, M.; Waidner, M.; Whinnett, D.: A status report on the SEMPER framework for secure electronic commerce. In: Computer Networks and ISDN Systems 30/ (1998) 1501-1510, 1998 TERENA Networking Conference, Dresden, Germany, October 5-8.

[ScWW99]

Schunter, M.; Waidner, M.; Whinnett, D.: The SEMPER Framework for Secure Electronic Commerce. In: Proc. Wirtschaftsinformatik ‘99, Physica-Verlag, Heidelberg, 1999.

[SEMP99a]

SEMPER Consortium: Architecture, Services and Protocols; *SEMPER* Deliverable D10; La Gaude, to be published in 1999.

[SEMP99b]

SEMPER Consortium: Final Public Report; *SEMPER* Deliverable D13; La Gaude, to be published by Springer-Verlag in 1999.

[Waid98]

Waidner, M.: Open Issues in Secure Electronic Commerce; IBM Research Report RZ 3070 26/10/1998, IBM Research Division, Zürich, Oct. 1998.

[Webe97]

Weber, A.: Zur Notwendigkeit sicherer Implementation digitaler Signaturen in offenen Systemen. In: Mehrseitige Sicherheit in der Kommunikationstechnik, Addison-Wesley-Longman 1997, 465-478.