Secure Network Communications and Secure Store & Forward Mechanisms within the SAP R/3 System

Bernhard Esslinger
SAP-AG
Bernhard.Esslinger@sap-ag.de

Jürgen Schneider
SAP-AG
J.Schneider@sap-ag.de

Abstract

Information security and data protection is gaining more and more importance with business software such as R/3 because:

- Business applications become "mission-critical" if companies carry out their most important business processes with them.
- Programs and data are subject to a greater danger of loss, change and espionage in client/server environments than in mainframe based systems.
- The danger increases even more as the systems become interconnected with publicly accessible LANs and WANs.

R/3 processes highly sensitive data (for example, company-internal and person-related information). Therefore a number of security mechanisms are already active in R/3 since the beginning:

- authentication of all users by means of passwords,
- R/3 authorization concept, and
- protection of the communication between front-end and application server by compression.

Now SAP enhances the security of R/3 by

- Securing online network communications (the SNC Project) and by
- Implementing secure store&forward mechanisms for electronic payment (the SSF Project).

1 Motivation and company policy

To ensure that the most technically advanced and scientifically sound security products can be used with R/3 SAP has decided not to include cryptographic modules into it's software.

SAP wants to support reasonable strong cryptographic mechanisms to protect the legitimate interests of the user. To ensure the exportability of R/3 SAP has chosen to provide hooks for third-party cryptographic modules.

To be able to support a variety of different security products a standardized interface is preferable. The use of such an interface enables the user to install a security product of his own choice, setup a security policy according to his own requirements, and to use algorithms that in his opinion are strong enough to protect his data. If some algorithms or protocols are proven to be insecure he simply can switch to another product that supports sufficiently secure algorithms.

This strategy has a number of advantages:

- The software is exportable.
- Each user can use his favorite security product which uses well analyzed protocols and algorithms.
- Algorithms and protocols can be changed without touching the application.
- SAP needs no extra department with ultimate expertise in implementing cryptographic algorithms and protocols.

To integrate security software with the specified standard interface (Generic Security Services API, GSS-API Version 2) there is the need for flexible and competent partners to develop the software parallel to the integration process. It is a good idea to look for partners in the academic community at an early stage of the development process. To be present on both European- and US-market GMD (German National Research Center for Information Technology) and MIT (Massachusetts Institute of Technology) have been selected as project partners.

2 The Secure Network Communications Project (SNC)

To satisfy the growing security requirements SAP has started the "Secure Network Communications" project. The major goal of the project is to better protect the access to R/3 via the front-end and to protect the communication between the front-end and the application server (see Figure 1). Here, it should be ensured to an even greater extent that only authorized users can log on to the system, and that the data on the WAN or LAN cannot be spied upon, falsified or deleted during communication.

Within this project SAP implemented in R/3 Release 3.1 an option permitting the integration of other vendors' network security products to provide secure

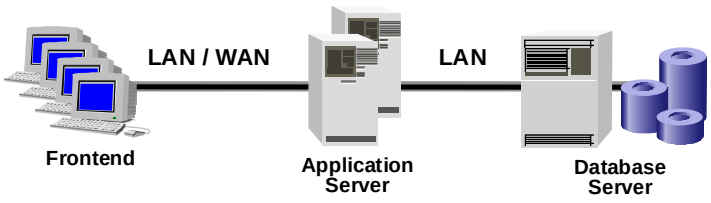


Fig. 1. 3-Level Client/Server Architecture of R/3

authentication and protected network communication. This option allows the use of products equipped with the standardized *Generic Security Services API (GSS-API) Version 2*.

The integration of R/3 in network security products has two significant advantages for the customer:

1. The security of R/3 is increased, since further security measures can be implemented using the security products supported.

   - There is the possibility to do an *end-to-end encryption* between the front-end and the application server.
   - Passwords are no longer observable on the link.
   - Some security products, for example SECUDE (see below), allow the use of *smartcards* for authentication. Smartcards are linked with the computer via a dedicated hardware device (smartcard reader) and are operational only after the user has entered a Personal Identification Number (PIN). As a design criterion a smartcard should never reveal the stored secret key and thus should be impossible to be copied.

2. Using this capability, customers can secure different applications in their client/server environment, including R/3, *with the same* security system.

   - The user only needs to log on to a security system once per session, and can then use all of the client/server environment services. This *single sign-on* automatically authenticates the user to the servers/applications without a password having to be entered each time. In many security systems a single sign-on is only valid for a certain number of hours or ceases to be valid when the smartcard is removed from the reader. Frequent logging on, including management of several passwords in the system, is no longer necessary. In this way, the user should be more willing to choose longer and more complex passwords.
   - In the same way, the system administrator also only needs to operate and maintain only *one* security system. The external security system authenticates users and servers. (However, the authorization profiles and user master records must still be maintained in R/3 itself, since the R/3 authorization concept is used within R/3.)
   - The company security policy can be integrated into R/3 with no additional costs. The policy implemented in the local network security product is used transparently in R/3. If the policy is changed in the security product it implicitly changes in the application.

2.1 The Security Products that R/3 is Aiming for

SAP has been looking at the following products for company-wide network security of client/server systems:

- Kerberos 5 from MIT and commercial vendors like OpenVision,
- SECUDE 5.0 from GMD (German National Research Center for Information Technology),
- OSF DCE based products from various vendors,
- SESAME 4 based products from various vendors,
- Entrust from Nortel.

SAP is starting with support for Kerberos and SECUDE. There will be implementations on all R/3 current 3.1 application server platforms (various UNIX operating systems, IBM AS/400 and Microsoft Windows NT). On the front-end platforms Windows 95, Windows NT, Unix/Motif, OS/2 Presentation Manager and Apple Macintosh are supported. Cooperation projects with Kerberos and SECUDE went alive at pilot customer sites in the fourth quarter of 1996.

SECUDE is the only available product supported by SAP right now, because MIT does not market a product. Right now we develop easy-to-handle transfer of authentication data between the security product and R/3. Furthermore we are developing an intensive test suite for the GSS API v.2 (and give it back to the IETF) and a certification process for vendors of other security products to fit with our interface.

SAP will supply validation procedures for these security products till end of 1997. Using these procedures the products will be checked to comply with the interface specification. The strength of the algorithms and protocols can and will not be checked within the certification process. The first products scheduled for certification are Kerberos from OpenVision (MIT does not market Kerberos itself) and SECUDE. Due to US export regulations SECUDE will be the first product supported by SAP that is available outside the US.

2.2 Integration of the R/3 System into the Network Security Products

The above mentioned products offer services

- for the *non-disclosing* authentication of users/programs/resources and
- for the protection of the transferred data (as plain text with an integrity check or encrypted).

An application must be modified according to these network products in a way that

- communication can be secured,

- customer administrators have the option of maintaining authentication centrally,
- a single sign-on can be implemented.

This means SAP must adapt the communication interfaces between all basic components of a R/3 system to utilize the functionality of the network security product (see Figure 2).

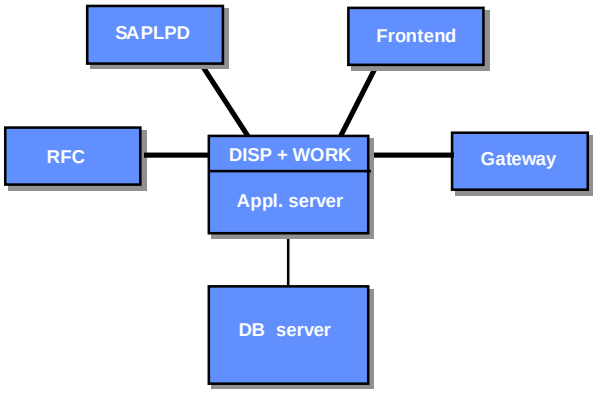


Fig. 2. Overview of the links to be secured by SAP

To ensure exportability and a maximum of flexibility for the user no security software will be supplied on the R/3 software releases. Customers wanting to use a certain network security product must obtain the product and an appropriate license directly from the respective supplier. This procedure is necessary to satisfy the strict and widely varying international legal requirements for export and utilization of cryptographic techniques.

2.3 Technical Information

There are already products on the market for establishing company-wide network security: a common feature of most of them, however, is that their installation *alone* does not influence the (in)security of existing applications. To use the security functions the entire network communication of the applications must be adapted („kerberized“). That is, for every product, extra adjustments within the application are necessary, which have varying effects on the total communication mechanism of the application, depending on the architecture of the security product.

Only recently have people started discussing a standardized Security API with a standard communication model to abstract from the individual products and their characteristics. The standardization proposals of a generic security interface (Generic Security services API / GSS-API) have been defined by the CAT (Common

Authentication Technologies) work group of the IETF (Internet Engineering Task Force).

This work group comprises representatives from companies/organizations such as Bull, Cybersafe, DEC, HP, IBM, ICL, MIT, OpenVision, OSF and SUN. A large part of the discussions were based on the security mechanisms of Kerberos 5 because it is contained as a (possible) security technology in the products of all the named companies.

The programming interface *GSS-API Version 1* was released in September 1993 as Internet RFCs-1508 & 1509 and is, for example, implemented in:

- Kerberos 5 e.g. from MIT, Cybersafe, OpenVision, ICL,
- SECUDE 5.0 GMD,
- OSF DCE 1.1 available from vendors such as DEC, HP, IBM,
- SESAME v. 4 available in products from Bull, ICL, Siemens.

When GSS-API version 1 was developed, the main consideration was the simplest form of client server communication, and compatibility was created on source level. On UNIX platforms the functional specification, with certain restrictions, also allows compatibility on the object level and for shared libraries. On the other hand, more precise function specifications are required for other platforms. The development of a first Microsoft Windows DLL interface was published in February '95, a DLL interface for the Apple Macintosh has not yet been published.

Due to the support of parallel-processing and computers with multiple processors it is imperative for the API to support the migration of security contexts across process boundaries. Therefore GSS-API Version 1 is not adequate for the needs of R/3. Among other things, R/3 uses from GSS-API Version 2 the functions gss_export_sec_context() and gss_import_sec_context() to transfer the endpoint of a secured connection (a security context) across process boundaries. GSS-API Version 2 will probably become a proposed standard within the first half of 1997 and then make version 1 obsolete (the high level specifications for GSS-API v.2 within RFC 2078 were already published in January 1997, but there will be an update soon). The GSS-API v.2 extensions needed to run R/3 are available with Kerberos 5 from MIT and SECUDE 5 from SAP's development partner GMD. Some other vendors of OSF DCE and SESAME based products have also announced these extensions.

For the R/3 System to be able to use a network security product, the product has to be available as a shared library or Dynamic Link Library (DLL) on all platforms and offer the functionality of GSS-API v.2. The R/3 System will continue to be shipped and installed without any additional secure network communication; the dynamic loading of the shared library/DLL will be controlled via the configuration (profile file, environment variables) at runtime.

The network communication security can be configured in the following way:

- Unsecured communication

- External authentication
- External authentication + integrity check
- External authentication + integrity check + encryption.

Undoubtedly you won't get security for "free" - it will definitely show up on the performance bill. Fortunately the above mentioned performance costs arise at a point in the R/3 architecture, which can easily be scaled, namely the application servers.

Encryption generally impacts performance regardless of whether it is carried out within the R/3 System or by network security products. Network security products may be better optimized than proprietary solutions, so that R/3's approach of using the services of the security products via the GSS-API is certainly the best alternative. (Each vendor does the business that he knows best.).

The use of the GSS-API offers SAP's customers the choice between several security products. Moreover, that guarantees that the customer can always get technologically up to date implementations.

3 The Secure Store & Forward Project (SSF)

For today's business application software it is increasingly important to support electronic financial transactions over publicly accessible data communication networks. In the course of such electronic transactions business data, such as electronic payments, order and account information is leaving the secured realm of an R/3 system to be transmitted over insecure networks. The most prominent example of a publicly accessible but insecure networking infrastructure is the Internet. In order to participate in the growing business of electronic commerce on the Internet and to use it for electronic financial transactions the data being exported and imported from R/3 have to be secured.

Some security requirements of electronic financial transactions are inherently different from the requirements for securing online communication between distributed components of one system. The communication „endpoints" of financial transactions cannot be computer systems or software processes, but have to be persons or other subjects with a legal meaning. Thus, the notion of end-to-end security to be achieved is different. Here, authentication mechanisms are used to provide evidence for the identity of a person and non-repudiation is becoming an additional requirement.

Also, data protection mechanisms have to take into account that many electronic transactions are achieved via store and forward communication, where not all participating parties are belonging to the same organization and not all of them are present online all the time. To accommodate batch processing and to cross boundaries of security domains (for example crossing firewall systems) the data is often stored on intermediary systems where it needs to be protected even outside the actual online communication.

The requirements described above motivated the Secure Store & Forward (SSF) project at SAP to provide adequate data security in the context of electronic transactions with R/3. For a solution to be adequate, the full spectrum of the business application modules provided with R/3, covering financial applications, as well as sales, production planning, logistics, human resources and others should be supported. In addition, either the source or destination of the data transmission might be a different software system. This lead to the following overall requirements:

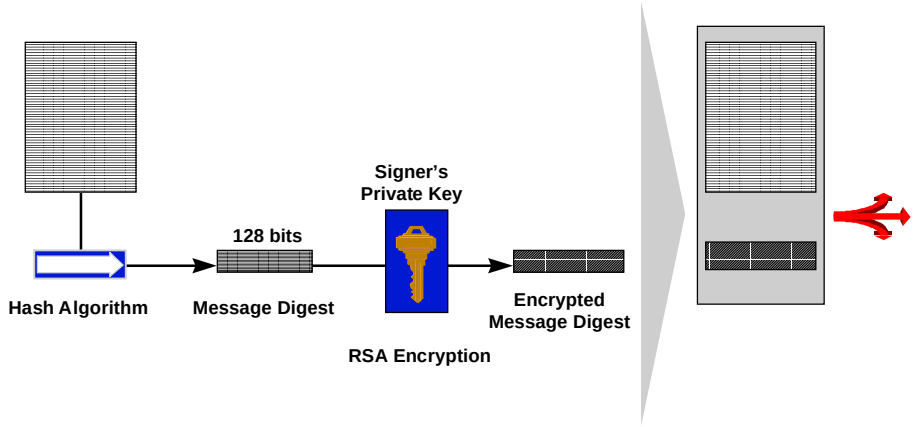
- The solution should provide data integrity, privacy, authentication and non-repudiation in the context of electronic transactions.
- The solution should be independent of the contents of the data to be protected and should be applicable, for example, to financial transactions, as well as personal data and others.
- The creation of the protected data must be separated and independent from the transport mechanism, protocol and medium.
- A standard and platform-independent security format must be used to facilitate processing of the secured data with various security toolkits and also by non R/3 systems.

The first R/3 application currently being equipped with SSF support are electronic payments.

3.1 Secure Electronic Payments with R/3

Electronic payments with R/3 can be done both for the incoming and outgoing directions. A broad range of formats for international payments such as SWIFT MT100, MT940 and many country-specific formats are supported today. The exchange of the data is realized via physical transportation of files on storage media such as diskettes, via dial-up connections or leased lines. The use of Electronic Data Interchange (EDI) is an option. With SSF support, use of the Internet for transporting the payment data in a secure way will be possible. This comprises the following:

1. To protect the payment data from being altered and to achieve authentication and non-repudiation electronic signatures are used (see Figure 3). One or more electronic signatures are possible to support different business policies of customers. The process of signing the payment data is embedded into the business workflow of the customer. Only when all electronic signatures of the empowered individuals have been performed, the payment data is ready for encryption and transmission.



To construct a digital signature for some given data a hash function is applied to the data first, which delivers a so-called „message digest". The „message digest" represents an unambiguous fingerprint for the message but is usually much shorter. If an ideal hash function is used, it will be impossible to compute input data which will produce the same digest. Then, the message digest is encrypted using the signer's private key. Anybody who has access to the corresponding public key of the signer can decrypt the message digest and verify the authenticity of the signature and the integrity of the data by applying the hash function to the data and comparing the result with the decrypted message digest.

Fig. 3. Digital Signature

2. To achieve confidentiality/privacy of the payment data the signed data is put into a digital envelope (see Figure 4). This means the signed data is encrypted for the intended recipient (bank, for example).

Both electronic signatures and digital envelopes require that there is some sort of a *public key infrastructure*. Archiving the received data will not effect the validity of the digital envelope and its contents. The properties of the digital envelope and its contents can be verified long after the original transmission.

The flow of events for electronic payments with R/3 and SSF support is sketched in Figure 5. We decided to use the standard format PKCS#7 (Public Key Cryptography Standards No. 7) for the signed data and the enveloped data. The cryptographic functions are taken as services from security products, such as SECUDE, Entrust or others. The security products are accessed from within R/3 via the SSF API defined by SAP. The R/3 application modules just use the SSF API which provides transparent access to the security product installed.

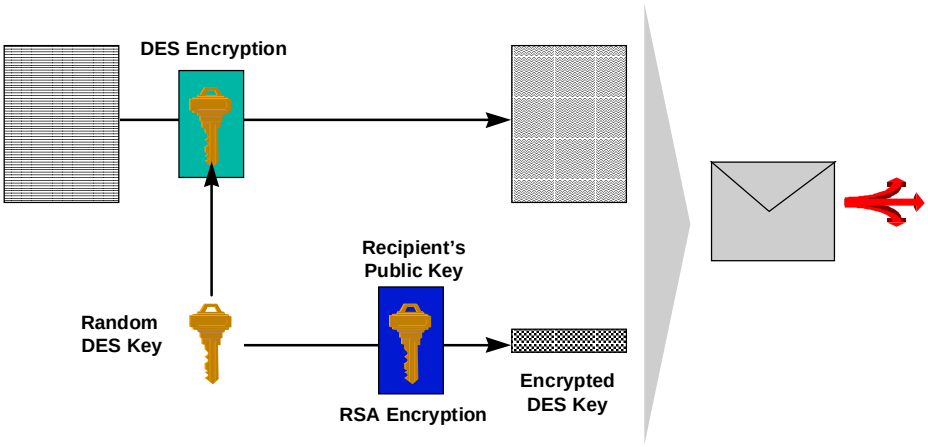
The PKCS#7 enveloped data can be transmitted using any convenient file or document transfer service. For transmission via the Internet, R/3 is acting as a HTTP

Client („HyperText Transfer Protocol“) accessing the HTTP (World-Wide Web) - Server of the bank.

A first prototype implementation was made together with Citibank, New York and Nortel’s Entrust as the security toolkit and was demonstrated last year. Currently, a first operational pilot is being developed with Deutsche Bank AG, Frankfurt using PKCS#7 and SECUDE from GMD.

3.2 Outlook

The SSF project within SAP is a response to urgent customer requests. These customers want to benefit from electronic financial transactions and from the Internet as a global and easily accessible infrastructure for electronic commerce. The SSF capabilities should quickly enable R/3 customers to do financial transactions and conduct part of their business over the Internet. The desirable progress of standards to achieve this in a homogenous fashion is too slow to respond to urgent customer needs.



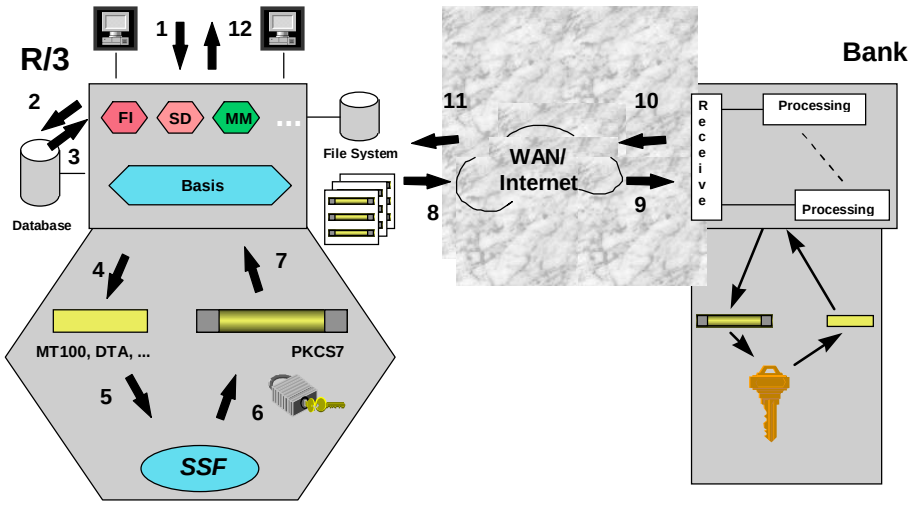
Putting a digital envelope around data to be protected means encrypting the data so that only the intended recipient(s) are able to decrypt the data. Typically, the data is DES encrypted using a newly generated DES key (message encryption key). Then, the message encryption key is encrypted using the recipient's public key. Only the owner of the corresponding private key is able to decrypt the message encryption key and then to decrypt the data contained in the digital envelope.

Fig. 4. Digital Envelope

We will continue to closely watch the progress of electronic payment standards, such as EDI and Secure Electronic Transactions (SET), currently being promoted for credit card payments. The protocol messages that we have designed using the PKCS#7 secure formats to support electronic payments with R/3 can evolve to conform to upcoming or future standards as soon as these standards satisfy our customer requirements.

Another important issue is the availability of a global infrastructure for public key certification and distribution. This is in fact one of the major hurdles hindering the roll-out of secure financial transactions solutions based on public-key cryptography. In principle, private companies and public organizations are willing to act as certification authorities (CA) and setup and operate public servers, if only the legal rules and constraints would be clear.

However, the whole subject is difficult, so that there will probably be no world-wide consensus on every detail and CAs in different countries will operate under slightly different laws. Therefore, cross-certification must be possible.



1	Order entry within R/3	7	Store in temporary file
2	Store order in R/3 database	8	Send to bank
3	Read orders from R/3 database	9	Receive at bank
4	Create payment format	10	Send receipt
5	Send payment format to SSF	11	Receive receipt
6	Perform electronic signature(s) and secure envelope	12	Show within R/3

Fig. 5. Flow chart: Secure Electronic Payment

Securing electronic payments from R/3 is just the beginning. For a number of the other R/3 application modules SSF provides important security enhancements to perform electronic financial transactions, such as treasury, sales and distribution.

4 Some related Links

Concerning the *GSS-API* (Generic Security Services), defined by the CAT working group of the IETF, see:

http://www.ietf.org/

ftp://ftp.internic.net/internet-drafts/draft-ietf-cat-gssv2-08.txt

This GSS interface allows an application to integrate into network security systems like Kerberos 5 and SECUDE 5.

About *Kerberos 5*:

http://web.mit.edu/

http://web.mit.edu/kerberos/www/krb5-1.0/announce.html

http://web.mit.edu/tytso/www/resume.html

http://web.mit.edu/aellwood/www/thesis/areaexam.html

About *SECUDE 5.0*:

http://www.darmstadt.gmd.de/secude/

http://www.darmstadt.gmd.de/TKT/security/commercial/

email: schneiw@darmstadt.gmd.de

Summary

Policy

Security in the sense of data protection is gaining more and more importance with SAP R/3 customers. There are two main reasons for this:

- R/3 becomes a "mission-critical" application if companies carry out their most important business processes with R/3.
- Programs and data are subject to a greater danger of loss, change and espionage in client/server environments than in mainframe based systems.

To satisfy this demands R/3 uses standard interfaces (GSS-API version 2, PKCS #7) and wide-spread security products.

The SNC Project

The *Secure Network Communications Project* preserves the confidentiality and integrity of the data transferred between the R/3 components of the network. Successful synchronization of the work between

- application vendor SAP AG,
- the builders of the network security products Kerberos (MIT, Boston) and SECUDE (GMD, Bonn/Darmstadt) and
- the active participants in the IETF working group CAT

helped to „kerberize“ the SAP R/3 system and to achieve a state-of-the-art *application-level-security*. This is done via the GSS-API version 2 which let’s the application programs at all ends call the security services offered by network security products. R/3 integrating security products is already productive. SAP welcomes the work of international standardizing bodies and reliable, well accepted cryptographic algorithms and protocols developed by the international research community.

The SSF Project

The Internet is increasingly used as a worldwide data communications infrastructure for financial transactions. *Electronic payment* between a customer and his bank is one example.

Enabling secure financial transactions over insecure networks means to build in mechanisms to achieve data integrity, authentication, privacy and non-repudiation at the level of persons and legal subjects. This is done by using electronic signatures and digital envelopes based on standards for secure data formats such as PKCS#7. The secure format puts an envelope (security wrapper) around the authenticated data before it is stored or transmitted.

The *Secure Store & Forward Project* targets this point. We plan to make the results of the SSF project productive in 1997. In between we will appreciate progress made in establishing a reliable public key infrastructure and international standards for financial transactions.

Both projects extensively use the security services of existing security products.