



# Secure Development

## in ABAP

THE BEST-RUN BUSINESSES RUN SAP 

- Introduction
- General Rules
- Extract from Top Ten Rules
- Authentication
- Authorization
- Auditing and Logging
- Sensitive Information
- Cryptography
- Deployment
- Summary

CZ, März '97

Ohne Firewall-System ist keine R/3-Installation völlig sicher

## SAP offenbart Sicherheitslücken

Leinfelden (mo) – Das R/3-System ist für Hacker häufig offen wie ein Scheunentor. Das ist die Konsequenz aus den Empfehlungen, die SAP in einem Sicherheitsleitfaden gibt. Der Leitfaden macht klar, daß alte R/3-Releases nicht als sicher gelten können. Auch bei aktuellen Versionen sind zum Teil umfangreiche Sicherungsmaßnahmen nötig. So ist eine Firewall praktisch unumgänglich, um den Datenbankserver vor Client-PCs zu schützen. So gestehen die Walldorfer ein, daß ohne diesen Schutz jeder PC im Netz auf die Datenbank zugreifen kann, wenn Oracle zum Einsatz kommt. Einzige Voraussetzung ist das entsprechende PC-Abfragewerkzeug. Hackern bieten sich weitere Wege ins R/3-System: zum Beispiel „Remote Function Calls“ (RFC). „Im R/3-System gibt es eine Rei-

he von sicherheitskritischen Funktionsbausteinen, die ohne spezielle Berechtigung per RFC aufgerufen werden können“, erläutert der Leitfaden. Diese Probleme bewertet SAP zu gering. So wünscht sich Bernd-Christoph Bijok, Datenschutzbeauftragter der Bosch Gruppe und hier weltweit verantwortlich für Datenschutz und Datensicherheit, daß SAP Sicherheitsbodenken ernster nimmt. Bosch hat sich bei einigen wichtigen Kerngeschäftsprozessen strategisch auf R/3 festgelegt. „SAP muß die bekannten Schwachstellen so schnell wie möglich beseitigen“, fordert Bijok. Das betrifft vor allem Zugriffsberechtigungen im Produktivsystem und Sicherheitslücken im Zusammenspiel mit der Oracle-Datenbank. Beim Berechtigungskonzept bemängelt er die konzeptionelle Schwäche, daß „die Komplexität allein schon ein Sicherheitsdefi-

zit ist“. Bei der Netzwerksicherheit fühlt sich Bijok allein gelassen: „Bisher hat uns die SAP keine klare Zusicherung gegeben, die empfohlenen Sicherheitsprodukte wie Secude auch in zukünftigen Releases zu unterstützen.“ Insgesamt ist Bijok von der Informationspolitik enttäuscht: „Die SAP muß uns demnächst frühzeitig und besser über Sicherheitsprobleme informieren und vor allem Lösungen aufzeigen.“ „Wir haben den Sicherheitsleitfaden auch intern allen Kollegen zur Verfügung gestellt“, erläutert Arnold Niedermaier, Marketing Manager zum Thema Sicherheit bei SAP. Die Sicherheitslücken führt er darauf zurück, daß zu Beginn der R/3-Programmierung die Anforderungen nicht so hoch waren. Das habe sich aber geändert, so daß keine neuen Löcher entstehen sollten.

“For hackers R/3 often is wide open..“

“In R/3 there are a number of security critical function modules which may be called via RFC without the need to have special authorizations.“

- The following slides show some examples of public discussions on security leaks in newspapers, magazines, mailing lists, newsgroups, and web sites.
- This newspaper article from March 1997 reports, that an SAP R/3 system is not secure without a firewall. It lists various paths client computers can use to access the database data:
  - Using Oracles Query Tool SQL Plus.
  - Using one of the several security critical function modules which can be called without special authorizations via RFC.
- Furthermore, the article reports customer criticism on the information policy of SAP.

## Die Datenschleuder

Das wissenschaftliche Fachblatt für Datenreisende  
Ein Organ des Chaos Computer Club



### Im Fadenkreuz: SAP R/3



- ▼ *Kryptodebatte verschärft sich*
- ▼ *Im Fadenkreuz: SAP R/3*
- ▼ *Dokumentation Congress '97*

ISSN 0930-1045  
März 1998, DM 5,00  
Postvertriebsstück C11301F

#02

**“It is possible to change the ABAP source code of the workbench in such a way, that it does not display malicious code.”**

...

**“Nobody can rule out, that this already has been done before.”**

- This issue of the magazine of the German “Chaos Computer Club” takes a close look at security in SAP R/3. The “Chaos Computer Club” is well known for revealing security deficits in information systems by hacking them and publishing the results.
- The focus of the article is on the source code of the ABAP development workbench. It claims that it is possible to change the ABAP source code of the workbench in such a way, that it does not display malicious code. Furthermore, nobody can rule out, that this already has been done before.

## Newsgroup Posting: Skip S\_TCODE

The screenshot shows a Netscape Newsgroup window titled "Re: Basis: Security - sap.mail2news.mit-r3-l - Netscape Newsgroup". The window displays a list of messages in a table with columns for Name, Subject, Sender, and Date. The selected message is from Alan Ross, dated 5/11/99, with the subject "Re: Basis: Security".

Name	Subject	Sender	Date
sap.r3.misc	Re: BASIS: Crash and Burn Box	Murray Ni...	5/11/99...
sap.misc	Re: LOG Output Determinatio...	Tylczynsk...	5/11/99...
sap.announce	SD, LOG, MM : Sales Availabi...	Dan Egge...	4/29/99...
de.com...curity	Basis: Security	Rcwix	5/11/99 2...
comp....y.misc	Re: Basis: Security	Helena Kaz...	5/11/99 3...
de.alt...sap-r3	Re: Basis: Security	Alan Ross	5/11/99 3...
sap.mai...it-r3-l	Fwd: Acct: IO, System allows ...	S. Piracha	5/11/99...
comp....ss.sap	Fwd: Basis:Acct: IO,	S. Piracha	5/11/99...
de.org.ccc	EDI: Workflow and workcenters	Ron Herr...	5/11/99...
	ACCT: Field status compatibility	Vinul Sar...	5/11/99...

The message content is as follows:

**Subject:** Re: Basis: Security  
**Date:** Mon, 10 May 1999 18:35:51 PDT  
**From:** [sap\\_security@hotmail.com](mailto:sap_security@hotmail.com) (Alan Ross)  
**Organization:** MIT-R3-L@MITVMA.MIT.EDU Mailing List  
**Newsgroups:** [sap.mail2news.mit-r3-l](mailto:sap.mail2news.mit-r3-l)

When call transaction is used it skips the s tcode auth. check for the transaction that is being called but it does check all other auth objects for the transaction.

In your example you should add ZEXP to the appropriate profiles or activity

Total messages: 3121 Unread messages: 2901

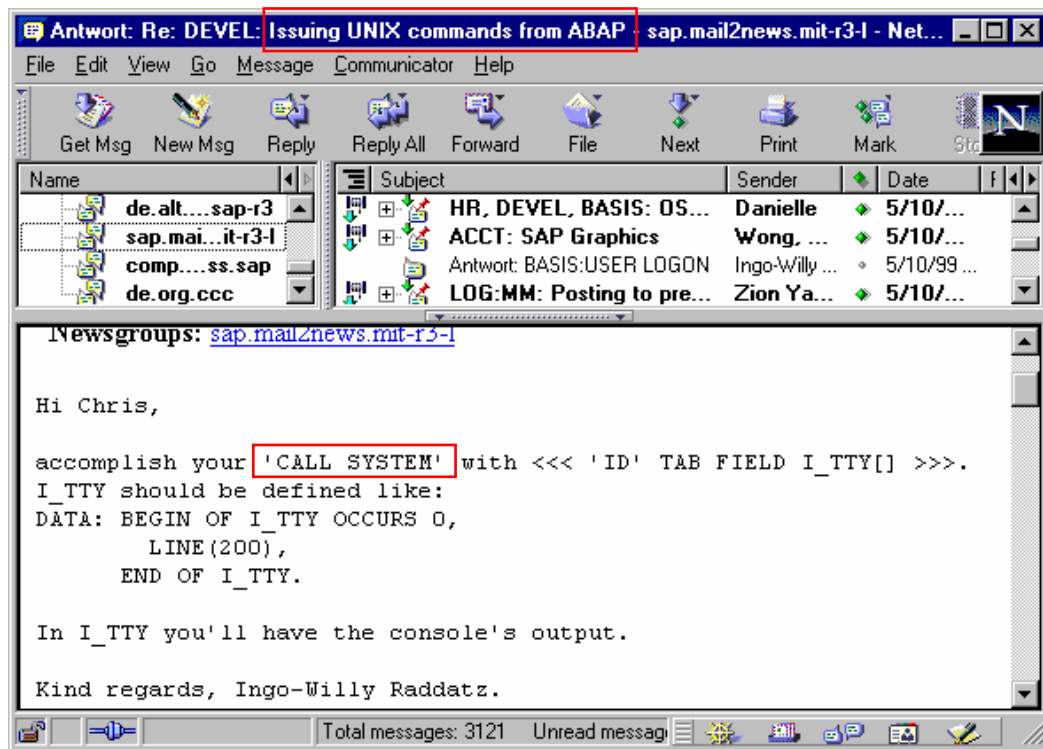
© SAP AG 2004

THE BEST-RUN BUSINESSES RUN SAP



- This newsgroup posting points to the fact, that the S\_TCODE authorization object is not checked when a transaction is started from an ABAP program with "CALL TRANSACTION".
- Therefore, the developer of a transaction never can rely on the automatic system checks on the S\_TCODE authorization object. Since 4.6 a table is available TCDCOUPLES (to be maintained via SE97) that allows a developer to input all called transaction for a calling transaction. Thus the transaction start of the called transactions will be checked. In lower releases one can only code own authorization checks for that case.

## Newsgroup Posting: OS-Command

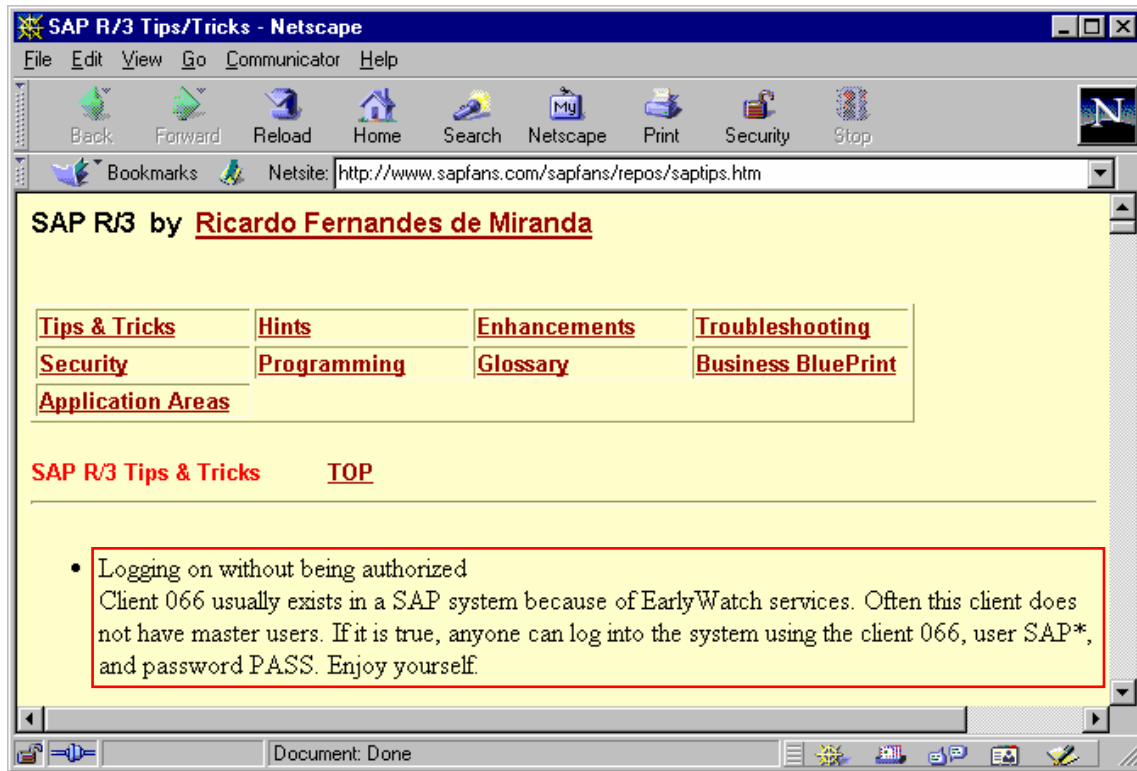


© SAP AG 2004

THE BEST-RUN BUSINESSES RUN SAP

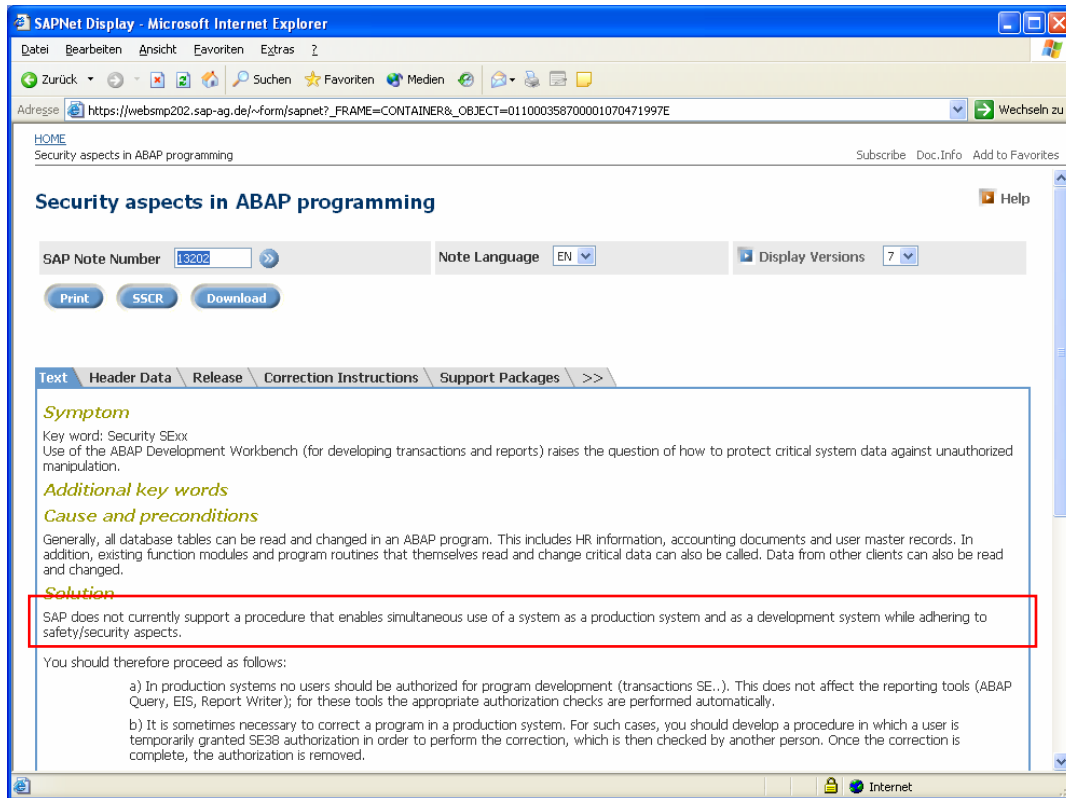


- With the “CALL SYSTEM” command, an ABAP program can easily execute operating system commands on the application server. This newsgroup posting explains, how the console output of the executed operating system call can be stored in an internal table.
- An external command is an alias defined in the SAP system that represents an operating system command.
  - For example, you can define the external command ZPING, which represents the operating system command ping <hostname>.
  - The possible set of commands is restricted to the ones defined in the SAP system.
  - External commands are maintained using transaction SM69 (maintain external commands) and executed using transaction SM49 (execute external commands).
- Both, the maintenance and the execution of external commands, are protected by authorization objects.
- The execution of external commands is checked by the authorization object S\_LOG\_COM.
- For the maintenance of external commands, you need an additional authorization based on the authorization object S\_RZL\_ADM with activity 01 and 03.



- This page on the internet site [www.sapfans.com](http://www.sapfans.com) points to the fact, that sometimes the early watch client (client 066) is not properly secured.
- If this is true, anyone can log on to the system using client 066, user SAP\*, and password PASS.

# OSS Note 13202



© SAP AG 2004

THE BEST-RUN BUSINESSES RUN SAP



- SAP note 13202 clearly states, that development and production system should be separate systems. It is not sufficient that development and testing is done with data in a special client on the production system.
- This is, because the use of the ABAP Development Workbench for developing transactions and reports allows read and write access to all data in all clients.
- Therefore, in production systems no users should be authorized for program development with transaction SExxx.



## Agenda

- Introduction
- General Rules
- Extract from Top Ten Rules
- Authentication
- Authorization
- Auditing and Logging
- Sensitive Information
- Cryptography
- Deployment
- Summary

© SAP AG 2004

THE BEST-RUN BUSINESSES RUN SAP



## Type of Data Processed by the Application

- |                                       |                          |
|---------------------------------------|--------------------------|
| ✎ <b>Personal data</b>                | ✓ <b>Online help</b>     |
| ✎ <b>Financial data</b>               | ✓ <b>Address book</b>    |
| ✎ <b>Account and credit card info</b> | ✓ <b>Product catalog</b> |
| ✎ <b>Prices and conditions</b>        |                          |
| ✎ <b>User master data</b>             |                          |
| ✎ <b>System access information</b>    |                          |
| ✎ <b>Passwords and IDs</b>            |                          |
| ✎ <b>Programs</b>                     |                          |

© SAP AG 2004

THE BEST-RUN BUSINESSES RUN SAP



- There are four major points to consider when designing the security aspects of an application:
  - Which types of data are processed by the application?
  - Which kind of functionality does the application offer?
  - What are the entry points into the application?
  - What are the exit point from the application?
- Most of the data which are processed by an application need to be secured from unauthorized access and unwanted changes. This includes, but is not limited to, especially sensitive data like:
  - Personal Data
  - Financial Data
  - Account and Credit Card Info
  - Prices and Conditions
  - User Master Data
  - System Access Information
  - Passwords and IDs
  - Program Source Code
- Only few data are public and therefore do not need special security precautions. Such data may be:
  - Online Help
  - Product Catalogs

## Type of Functionality in the Application

- **Aggregation or summation without single record processing**
- **Display „only“**
- **Display and change**
- **Restricted number of objects „only“ („Plant“, „Company code“)**
- **Restricted functionality „only“**
- **All objects of one type**
- **Object type and/or access type are parameters of the application (generic functionality), e.g. editor, table maintenance, up/download, OS access, RFC call**
- **Navigation to other applications**

© SAP AG 2004

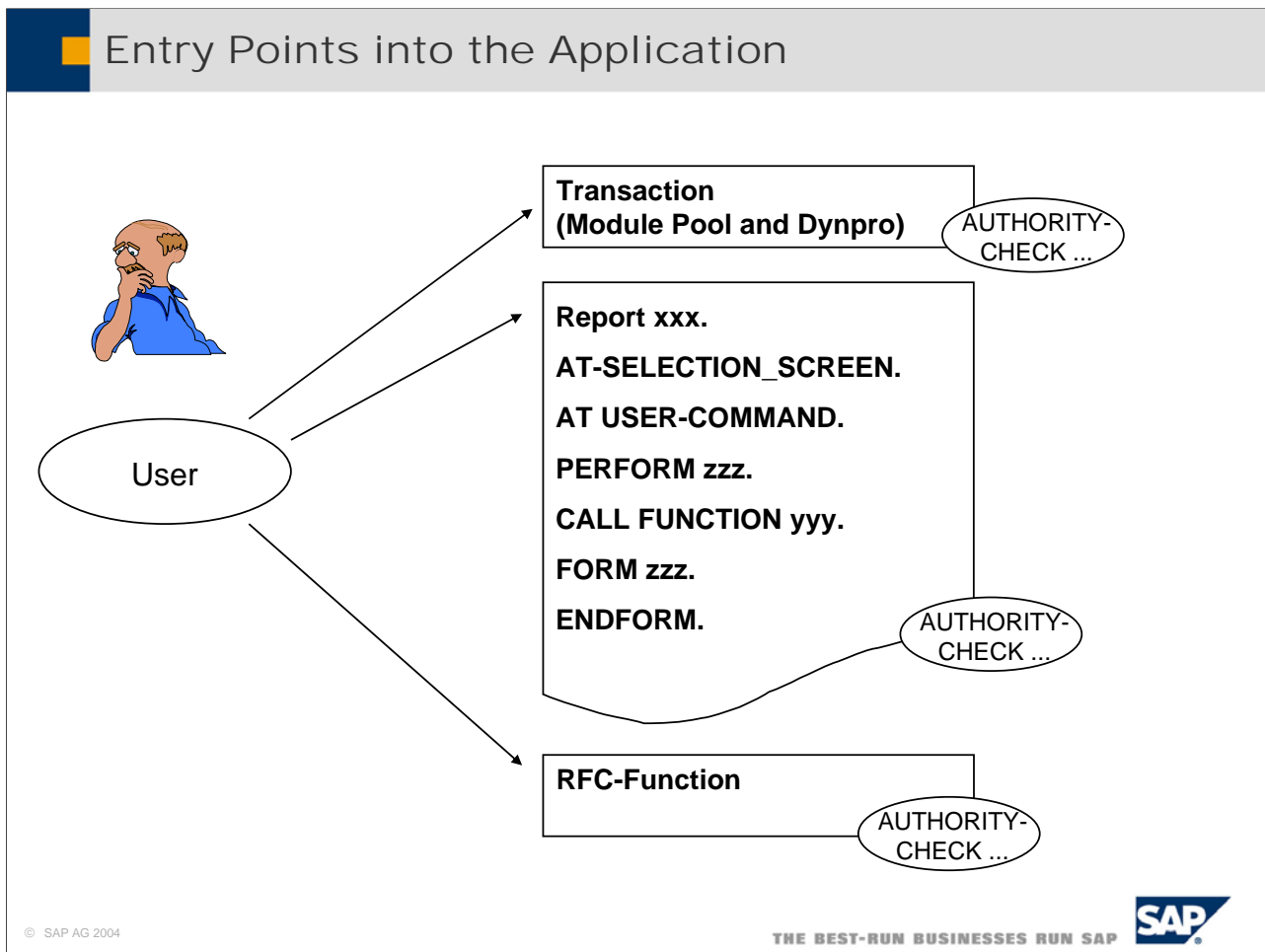
THE BEST-RUN BUSINESSES RUN SAP



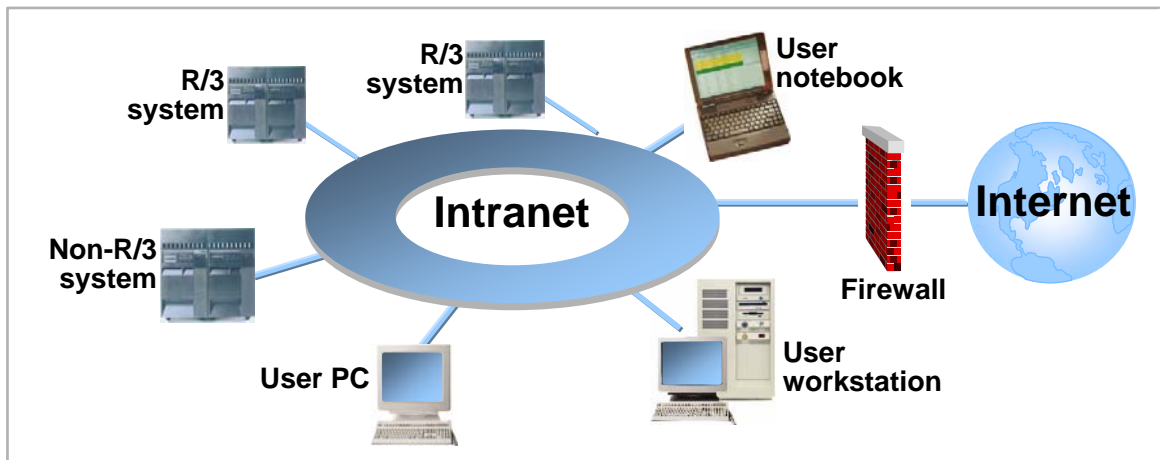
- The second important aspect to consider in application security design is the type of functionality offered by the application. Therefore functional design is tightly interrelated with security design. An application may offer several degrees of accessibility to different amounts of data.
- Four degrees of accessibility in order of increasing accessibility are:
  - Aggregation or summation of records without single record processing
  - Display of single records
  - Creation of single records
  - Change or deletion of single records
- Apart from these four typical degrees of accessibility an application might define several other modes of restricted functionality.
- The amount of data may vary from:
  - A subset of all objects of one type defined by the values of one or more attributes (e.g. Plant or Company Code)
  - All Objects of one type
- Extreme caution has to be taken in the following circumstances:
  - The object type is a parameter of the application
  - The access type is a parameter of the application
  - The application calls other applications
- Examples are:
  - Editors
  - Table maintenance application
  - Up- and download
  - OS access

© SAP AG  
RFC calls

## Entry Points into the Application



- The third important aspect in application security design are the entry points into and the exit points from an application. All these points should be secured with an authority check.
- Entry points are:
  - Every transaction calling the program
  - Every subroutine of the program
  - Every function module of the program
  - Every method of a class
- Exit points are:
  - PERFORM
  - CALL FUNCTION
  - CALL TRANSACTION
  - SUBMIT FORM
- And, of course, every request of the user within the program has to be checked, whether or not the program is allowed to perform that request



### Where is my data now, where will it be in the future ?

- ⇒ Database
- ⇒ File system on the application server
- ⇒ File system on the front end
- ⇒ Internal network (file server, mail server, internal web server, ...)
- ⇒ External network (mail server, external web server, ...)

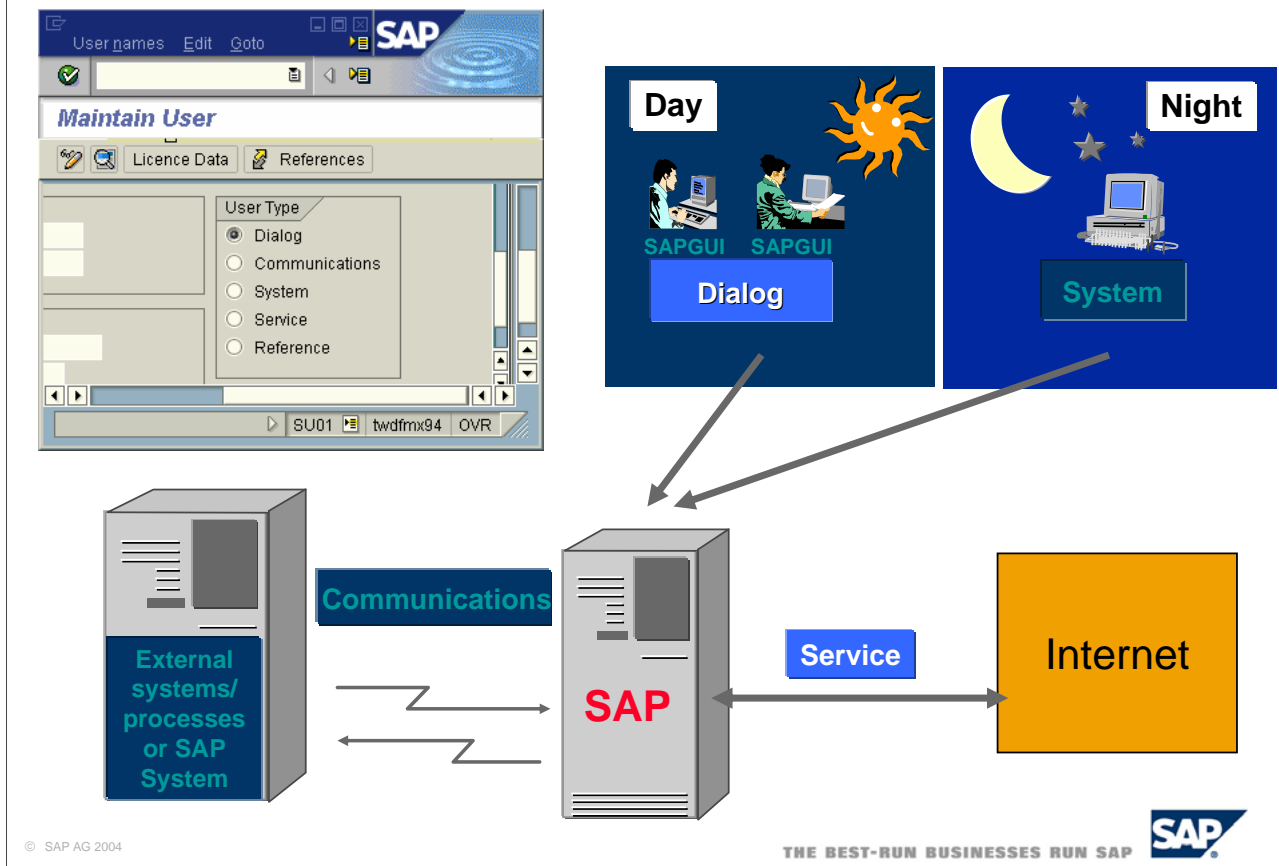
- Exit points sometimes transport the data from the current system to other systems. These systems may be
  - Database of the SAP system
  - File system on the application server
  - File System on the front end server
  - Some system in the internal network (file server, mail server, internal web server,...)
  - Some system in external networks
- Functionality sending data elsewhere has to be secured with authority checks

- **For auditing and revision purposes, it is often necessary to log changes to data**
- **This is done by writing change documents to the database, whenever sensitive data is changed or deleted.**
- **The change documents contain name of the changing user, date, time, transaction name plus the old and new values of the relevant fields**
- **Change documents can be written by the application with the help of generated FORM subroutines. These subroutines are created with transaction SCDO**

- For Auditing and Revision purposes, it is often necessary that an application logs changes to its data. This is important for financial accounting and many other applications, and often enforced by law.
- Such applications write so called Change Documents to the database at the same time, when sensitive data is changed or deleted.
- The change documents contain the following information:
  - Name of the user
  - Data and Time
  - Transaction Name
  - Old and new Values of the change-relevant fields
- A field is change-relevant, if the data element of the field's type is marked as change-relevant in the ABAP dictionary.
- Change documents are written by applications with the help of generated FORM subroutines. These subroutines are created with transaction SCDO

- Introduction
- General Rules
- Extract from Top Ten Rules
- Authentication
- Authorization
- Auditing and Logging
- Sensitive Information
- Cryptography
- Deployment
- Summary

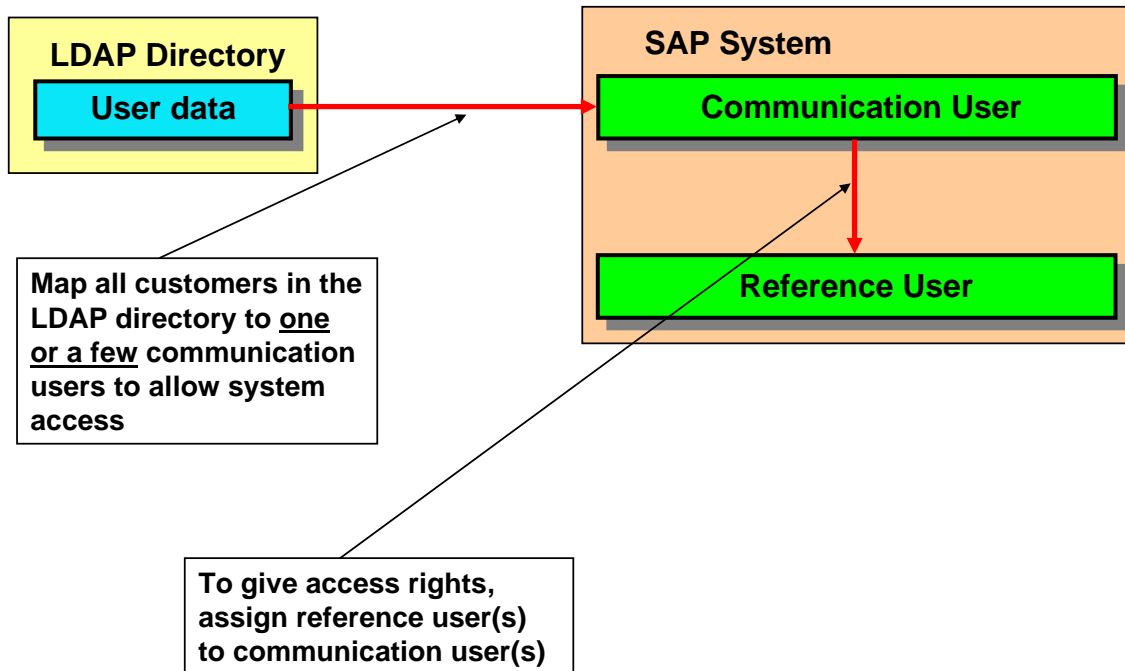
## SAP User Types



- The Dialog user is the most common user type. Users are always restricted to specific clients, and are subject to an SAP password check. Like all users, dialog users require authorization profiles, in addition to a password, to perform system and/or business tasks.
- The Communications user type should be used for communication without dialog between different systems (RFC/CPIC).
- The System user type can be used to run background jobs. These users cannot log onto the system and work interactively. A user of this type is excluded from the general settings for password validity.
- The Service user type allows multiple logon. The system does not check for expired and initial passwords. For example, the Service user is used for anonymous system access during Internet scenarios. The anonymously started session can later be continued as a personal session with a Dialog user.
- The Reference user is, like the Service user, not assigned to a particular person. A Reference user is used only to assign additional authorizations to Dialog users. Dialog users can refer to a Reference user and then inherit its authorizations.
- Only Dialog and Service users can work interactively on the SAP System. The Dialog users are required to change their passwords at certain intervals, if this is defined in the profile parameters.



## The Reference User



© SAP AG 2004

THE BEST-RUN BUSINESSES RUN SAP



- The reference user is a new user type since 4.6. The reference user allows to easily grant roles or profiles (authorizations) to users.
- Why should you use it?
- To give authorization checks a higher performance and easily assign the same rights to a multitude of users.
- Why is that?
- To answer the question, we must first understand how authorization checks work. When a user logs on, his/her authorization data are written into the user buffer and all authorization checks look into the buffer to see what a user is allowed to do or not. Thus if you have a multitude of users in your system (usually ESS-users, or users that browse in an internet catalogue), you assign either the user a reference user. Thus when the first user logs on the reference user's buffer is filled only once (and not everybody else's user buffer)
- The above mentioned example shows what you can recommend customers who want to give thousands of customer user's access to their system. Instead of creating all users in an SAP system (as service users), the customer creates them in an LDAP directory and maps the LDAP user on a communication user. The communication user then is assigned a reference user in the SAP system. The SAP system should be release 6.10 (4.6c with minor functionality) the list to allow for convenient LDAP user mass synchronization.

■ **For BSP-applications (Business Server Pages) in the Web Application Server, there are four possible authentication procedures:**

■ **Anonymous log on**

- With transaction SICF a service user is mapped to the service

■ **Log on with ID and password in a browser dialog box**

- Language and client are preset

■ **HTML log on form**

- Anonymously accessible page with fields *sap-client*, *sap-user*, *sap-password*, and *sap-language*

■ **Single sign-on**

- Using X.509 client certificates

- The SAP Web Application Server provides four possible logon procedures:
- The user does not need to log on; the application may be used anonymously. In order to enable anonymous log on to a service, a service user has to be created with transaction SU01 and mapped to that service in transaction SICF.
- The user is queried for his or her user ID and password in a dialog box in the browser. The language and client are preset.
- An HTML Log On Form is created with the fields *sap-client*, *sap-user*, *sap-password*, and *sap-language*. This page has to be accessible for an anonymous user.
- The users are identified through Single Sign On or X.509 client certificates.

## BSP: Anonymous Logon and SICF

ICF path: /default\_host/sap  
 ICF object: Test Service  
 Description: in EN (Not maintained) Other languages

**Service data**

**Anonymous Logon Data**  
 Logon Data Required:   
 Client:   
 User:   
 Password: \*\*\*\*\* still initial  
 Language:

**Security requirements**  
 Standard  
 SSL  
 Client Certificate w. SSL

**Service options**  
 SAP Authorizatr:  ErrorType:   
 Session Timeout: 00:00:00 (HH:MM:SS)

**Basic Authentication**  
 Standard R/3 User  
 Internet Users

**Administration**  
 Last Changed by: SHANAGHY CreatedBy: SHANAGHY  
 Changed On: 03.08.2001 Created on: 03.08.2001

© SAP AG 2004

THE BEST-RUN BUSINESSES RUN SAP



- Call transaction SICF. The initial screen displays the services that already exist. The existing services are located under "sap" and are provided by SAP.
- To create your own service, position your cursor on the object in the tree that you want to be the parent of the new service. This can be an existing service or a Virtual Host. Choose Service / Virt. Host / Create service (or click on it with the right mouse button and choose New Sub element). In the following dialog box for your service or the alias for an existing service, enter a name and decide whether you want to create a service (Independent Service) or an internal alias (Alias for an existing service). Note that the service name can be 15 characters maximum! Since services are transported, they are subject to the transport system's restrictions.
- You now have the option of setting logon data for the service. This data will be automatically read and processed when the service is called, in order to authenticate the client. You can set the following logon data: "Client", "User", "Password", "Language".
- Use a user name that is marked in transaction SU01 as a service user. If you set the option Logon data required, the anonymous user data is also used for all sub services; however, individual fields can be overwritten in the sub services.
- You can set permissions for using the server. If text has been entered in the field SAP Authorization (for example, 'CHECK'), when this ICF service is used, the user's permissions are checked against the value entered in this field. The permissions object used is S\_ICF. In this example, the user must have the following permissions:  
 "S\_ICF-ICF\_FIELD = 'SERVICE' " and "S\_ICF-ICF\_VALUE = 'CHECK' "
- The following security requirements can be set for a service or an alias:
  - Standard (default)
  - SSL
  - Client certificate with SSL
- If Basic Authentication is used to log on to SAP Web Application Server (in other words, if no Anonymous Logon Data is set and the standard logon procedure is used), either the standard SAP user or an Internet user can be used.
  - The default SAP user is the user name that is entered in Transaction SU01.
  - The Internet user is the alias name that can be longer than the normal SAP user name. This is also set in SU01.
- Depending on the settings, the input in the Basic Authentication popup is interpreted as either a user name or an alias.

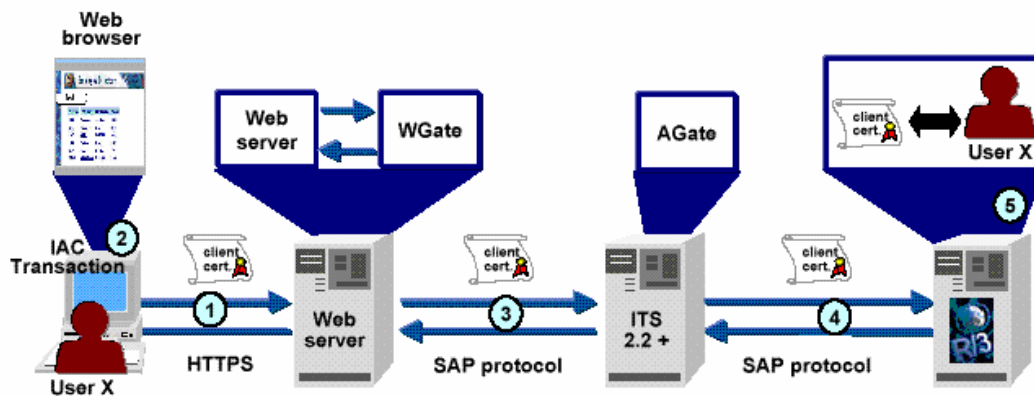
- **Create internet users with SU01 as service users**
  - User type is service
  - Assign a reference user
- **Do not use SU05**

- You can create external users as Internet users (called service users in SU01). Here only few personal data items are entered.
- Assign a reference user to the user you want to use as internet user. Reference users extend authorizations and are used to give internet users identical authorizations.
- For various classes of Internet users, you may require different reference users with different authorizations.
- Some Web applications require an individual SAP user name and password, most do not. However even these applications may require identification. A user can, for example, navigate anonymously in a product catalog; but must identify him or herself as a customer to place an order.

- **Provide a publicly accessible HTML page where the user can enter his or her personal data**
- **Use function module SUSR\_USER\_INTERNET\_CREATE to create an internet user**

- An administrator can create Internet users. However, an application may provide the means that external users create Internet users for themselves. A job application internet service would be an example for such a scenario.
- To do this, create a publicly accessible HTML page and use function module SUSR\_USER\_INTERNET\_CREATE.

## X.509 Certificates with ITS



1. Internet user is authenticated by web server via X.509 certificate
2. Web transaction (IAC) is started
3. X.509 certificate is passed from web server to W-Gate to A-Gate
4. X.509 certificate is passed to the SAP system
5. Log on to SAP system by using the client certificate  
⇒ "named user"

© SAP AG 2004

THE BEST-RUN BUSINESSES RUN SAP



- This slide shows, how X.509 Certificates with IST work:
- The internet user is authenticated by the web server with a X.509 certificate.
- When the user start a web transaction (an Internet application Components) the web server passes the X.506 to the W-Gate, which passes it to the A-Gate, which passes it to the SAP system.
- The Log on to the SAP system is done by using the client certificate. Therefore the IAC is executed by a "named" user.



### If an ABAP application needs to perform or repeat authentication

- ... for the same or another user ID
- **Solution:** write an RFC enabled function module and call it. The logon screen will be displayed.



### If an ABAP application needs to perform authorization checks for a different user ID

- **Solution:** Use function module `AUTHORITY_CHECK`



### If an external program needs to manage passwords

- ... needs to change password, or check for password expiration
- **Solution:** use RFC enabled function modules `SUSR_USER_CHANGE_PASSWORD RFC`

- In some cases, your application might want to perform user checks or secure data with additional passwords. There are various scenarios, in which this might be needed:
- If an ABAP application needs to perform or repeat authentication for the same or another user ID, you can write an RFC enabled function module and call it with destination `SELF`. A logon screen will be displayed and the user has to log on again.
- If an ABAP application needs to perform authorization for a different user ID use function module `AUTHORITY_CHECK`.
- If an external program needs to manage password, i.e. needs to change passwords or check for passwords, use RFC function module `SUSR_USR_CHANGE_PASSWORD RFC`.
- By all means do not check for hard-coded passwords, or accept e.g. the user name as a password, as done in report `RPUDELPN` (in Releases 4.5 and lower).

- Introduction
- General Rules
- Extract from Top Ten Rules
- Authentication
- Authorization
- Auditing and Logging
- Sensitive Information
- Cryptography
- Deployment
- Summary



### The authorization concept defines rules on

- ... how to create users in a system
- ... who may execute which actions, especially how to
  - ◆ ... restrict display and change of data depending on user roles. This enhances the security of the system.
  - ◆ ... show the user only those actions, which are relevant for his role. This simplifies the usage of the system.

- The SAP authorization concept defines to sets of rules:
- How users in a system are created
- Who is authorized to execute which actions

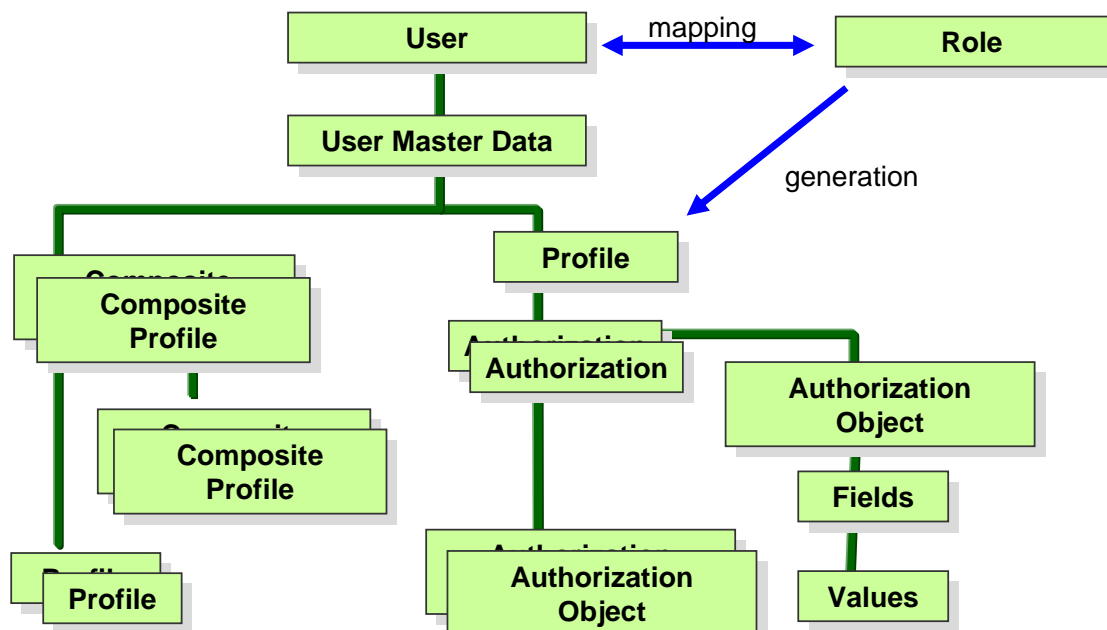
- **Authorization object**
  - **Designed to fit to the business process to be secured.**
  - **Consists of authorization fields and their possible values, e.g.**
    - **Activity: which activities are allowed (display, change, ...)**
    - **Name: which business objects may be operated on (e.g. which company code)**
- **Authorization**
  - **An instance of an authorization object with specific values for the authorization fields**
- **Authorization profile**
  - **A collection of authorizations which are necessary to perform a certain task or function.**

- Authorization Objects are at the heart of the SAP authorization concept. When a business module is designed, the module designers also design the authorizations objects to fit into the business processes of that module.
- An Authorization Object consists of Authorization Fields and their possible values.
  - As an example, consider you want to secure the access to company codes. Your application is able to create, change, display and delete company codes. You want to secure what is done to a business code, and to which business codes something is done. Therefore you define two authorization fields for your authorization object:
  - Activity: The contents of this field define, what action is allowed. The possible values are for example 01 (create), 02 (display), 03 (change), 05 (delete).
  - Company Code: The contents of this field define, which company codes can be operated on. The possible values are \* (all company codes), or a list of one or more company codes
- An Authorization is an instance of an authorization object with specific values for the authorization fields. Consider these to authorizations:
  - Activity = 02, Company Code = 1000. Users with this Authorization are allowed to display company code 1000. They are not allowed to display other company codes, and they are not allowed to do something else besides displaying.
  - Activity = 03, Company Code = \*. Users with this Authorization are allowed to change all company codes.
- For maintainability reasons, an authorization is not directly assigned to the master data of a user. Rather, authorizations are collected in profiles, and these profiles are then assigned to the master data of a user.
- Nowadays, profiles are generated with transaction PFCG. In order for PFCG to be able to generate profiles, the developer of the authorization object has to maintain default values for the authorization fields.

- **What is a “role”**  
(naming since Rel. R/3 4.6c, formerly called “activity group”  
Rel. 4.6b and lower)
- **A role / activity group consists of**
  - ... a menu
  - ... Authorizations  
→ specific values for the fields of authorization objects.  
These authorizations are used to generate an authorization  
profile
  - ... a list of users belonging to it (may change over time)

- A role (called activity group before Release 4.6C) consists of
  - A menu containing entries with all the transactions which are relevant to users which act in that role
  - All authorizations necessary to use these transactions. These authorizations are generated/defined in the profile generator and combined in an authorization profile
  - A list of users belonging to that role
- From a business point of view, a role specifies a set of tasks, which have to be performed by a person. Roles are, e.g., Database Administrator, Warehouse Operator, Payroll Accountant or Sales Manager.
- SAP delivers pre-defined roles to its customers.

## Definition of Terms



© SAP AG 2004

THE BEST-RUN BUSINESSES RUN SAP



- This slide shows again the association between the various objects:
- Every user has a set of master data.
- Every user also is member of one or more roles
- The user master data holds profiles. Profiles may be simple profiles or composite profiles. Composite profiles contain other composite profiles or simple profiles.
- The profiles are usually generated from a role with the profile generator PFCG. A user who is member of a role has the profile belonging to that role
- A Profile is a collection of Authorizations.
- Each Authorization is an instance of an Authorization Object, where the Authorization Fields are assigned specific values.

## Creation of Authorization Objects

- **Authorization fields are created with transaction SU20**
- **Authorization objects are created with transaction SU21**
- **Default values for authorization fields for use with the profile generator are created with transaction SU22**
- **Use transaction SE93 to associate a transaction with an authorization object that will be checked by the system on start of that transaction**

- Authorization Fields are created with Transaction SU20. You should reuse existing fields as much as possible. E.g. you can almost always reuse the existing activity field ACTVT. In the HR application the Activity field is not called ACTVT, but AUTHC.
- Possible Activities of Authorization Field ACTVT are stored in table TACT. When you use field ACTVT you have to assign your Authorization Object to the used activities. This association is stored in table TACTZ. Do not invent new activities without the urgent need to do so.
- The Authorization Field Values are stored in table AUTHX from Release 4.6 onwards. In earlier releases, these values were stored in tables AUTHA for applications and AUTHB for the SAP basis.
- Authorization Objects are created with transaction SU21.
- You can define default values for the Authorization Fields of an Authorization Object, and associate these values to a specific Transaction with Transaction SU22. These default values are used by the profile generator.
- With transaction SE93 you can assign an Authorization Object to a transaction. This Authorization Object is checked automatically by the system on transaction start. The other authorizations have to be checked by the programs of the module with AUTHORITY CHECK.
- Think carefully in advance how you want to build new authorization objects. Because you as a developer can easily change it later, BUT customers who have set up an authorization concept on your prior design of an authorization object run into loads of reworking and adaptation effort, if you add new activities to an object later, or split activities or even enter new fields.

# How do you maintain default authorization values?

## Transaction su22 -> check indicators

U	N	C	CM	Check ID	Object	ObjectDescription
✓				Check	B_ALE_LSYS	ALE/EDI: Maintaining Logical Systems
✓				Check	B_ALE_RECVC	ALE/EDI: Receiving IDocs via RFC
✓				Check	C_KLAH_BKP	Authorization for Class Maintenance
✓				Check	K_KC_DSK	EC-EIS: Authorization for Structures and Key Figures
✓				Check	PL06	Personnel Planning
✓				Check	P_TCODE	HR: Transaction codes
✓				Check	S_ADMI_FCD	System Authorizations
✓				Check	S_CTS_ADMI	Administration Functions in the Change and Transport System
✓				Check	S_DATASET	Authorization for file access
✓				Check	S_DEVELOP	ABAP Workbench
✓				Check	S_DOKU_AUT	SE61 Documentation Maintenance Authorization
✓				Check	S_GUI	Authorization for GUI activities
✓				Check/maintain	S_HIERARCH	Hierarchy maintenance authorization check
✓				Check	S_IDOCCTRL	WFEDI: S_IDOCCTRL - General Access to IDoc Functions
✓				Check	S_IDOCDEFT	WFEDI: S_IDOCDEFT - Access to IDoc Development
✓				Check	S_IDOCPART	WFEDI: S_IDOCPART - Access to partner profile (IDoc)
✓				Check	S_IMG_ACTV	IMG: Authorization to perform functions in IMG
✓				Check/maintain	S_NUMBER	Number Range Maintenance
✓				Check	S_OLE_CALL	OLE calls from ABAP programs
✓				Check/maintain	S_PRO_AUTH	IMG: New authorizations for projects
✓				Check	S_SPO_DEV	Spool: Device authorizations
✓				Check	S_SPO_PAGE	Spool: Restriction on Maximum Number of Pages
✓				Check/maintain	S_TABU_CLI	Cross-Client Table Maintenance
✓				Check/maintain	S_TABU_DIS	Table Maintenance (via standard tools such as SM30)
✓				Check	S_TCODE	Transaction Code Check at Transaction Start
✓				Check	S_TRANSLAT	Translation environment authorization object
✓				Check/maintain	S_TRANSPRT	Transport Organizer
✓				Check	S_USER_AUT	User Master Maintenance: Authorizations
✓				Check	S_USER_PRO	User Master Maintenance: Authorization Profile

### What do the *check indicators* mean?

'U' means unchecked. This means that you, the developer, have not maintained any check indicators yet. You should do so and never deliver a transaction with unchecked check indicators.

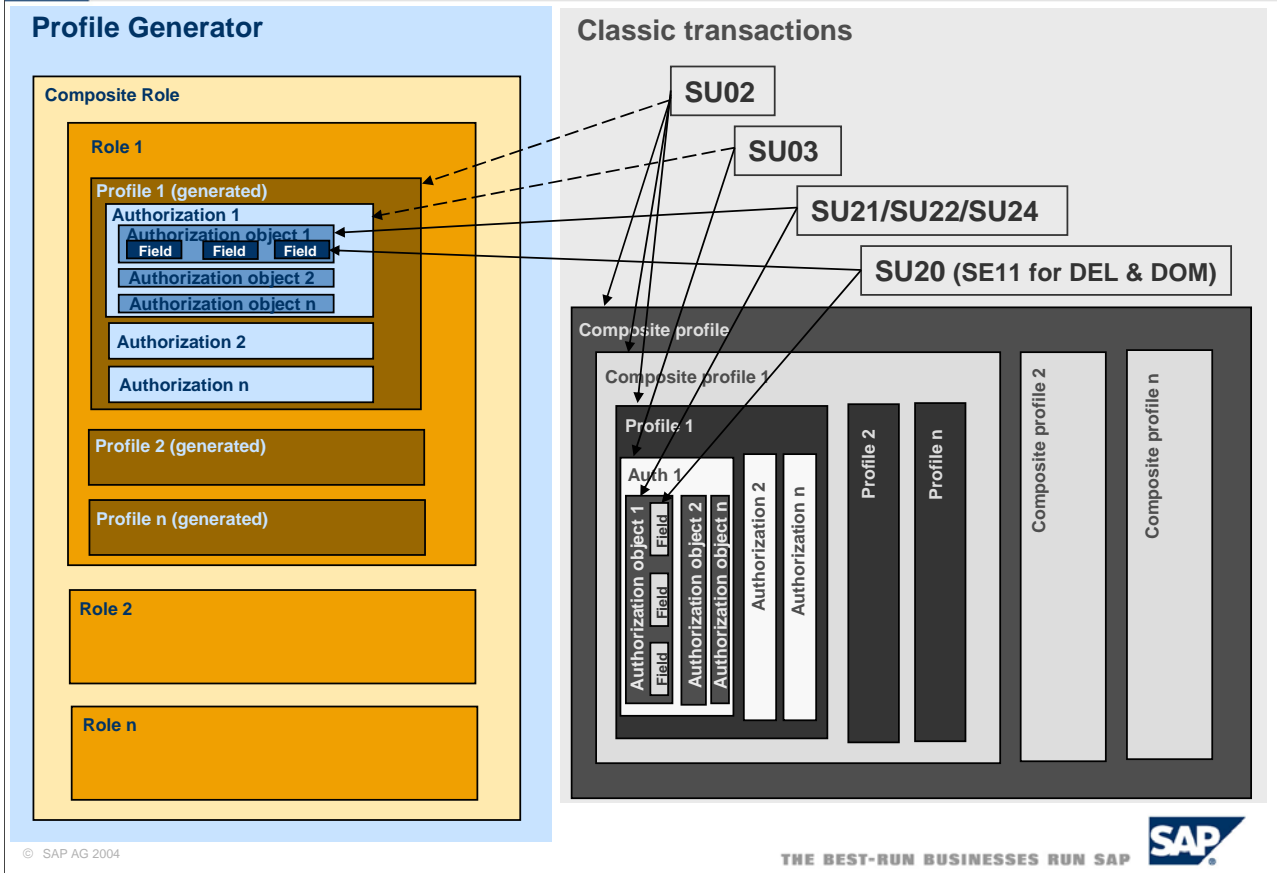
'N' means not checked. Though one performs a transaction, an authorization object connected with the transaction will not be checked.

'C' means that the authorization object for the transaction is checked but there won't be any authorization default values in the profile generator.

'CM' means that the authorization object will be checked and default authorization values for the profile generator will be created.

- For more details on check indicators for default authorization values, please take a look at SAP note: 0449832.

# The Profile Generator in Context



- This slide shows the relations of the previous slide again and lists the transactions used in creating the various entities.
- This slide also shows, how the profile generator uses the information which was created with SU21, SU22, SU24, SU20 (and SE11 for fields DEL and DOM) to generate profiles and authorizations.
- To know, which authorization objects are used by a transaction, system administrators can switch on an authorization trace with by setting the profile parameter auth/authorization\_trace to y. The use of Authorizations by transactions and RFC function modules will be traced in table USOBX. See also note 0543164.
- Paths to the documentation of SU22 can be found via notes 0155283, and 0093769.

- **The implementation of an authorization concept is difficult for administrators!**

- **They need good tool support**



- **High quality of data for the profile generator**
- **Useful and tested templates for roles**
- **Automatically generated authorization concepts for distributed applications**



**And they need good application documentation!**

- To Design and Implement an Authorization Concept is very difficult for administrators.
- The Authorization Concept specifies in detail:
  - Which users act in which roles
  - Which Authorizations are in which roles
  - Which naming conventions exist for users and roles
  - To whom does a user have to apply to be granted membership to a role.
  - Who is responsible to grant membership to a role.
- To create an Authorization Concept, Administrators need
  - High quality data for the profile generator
  - Useful and tested templates for roles
  - Automatically generated Authorization Concepts for Distributed Applications
  - Very good Application Documentation, stating clearly which Authorization Objects are checked and what Authorizations are needed to perform which tasks.
- The Authorization Concept needs to be signed off by the customer



## Goal of Authorization Checks

- **At start:** Check for authorization to start the transaction
- **PBO 1. Dynpro:** Checks how to modify the GUI dynamically
- **PAI 1. Dynpro:** Checks whether the selected activity may be performed on the selected object
- **AT USER-COMMAND UPDATE ...:**

Checks before  
executing critical  
functions

### Syntax AUTHORITY-CHECK

```
AUTHORITY-CHECK OBJECT <Authorization Object>
  ID <Authorizationfield1> FIELD <ValueToCheckFor1>
  ID <Authorizationfield2> FIELD <ValueToCheckFor2>
  ....
  ID <Authorizationfieldn> DUMMY .

IF SY-SUBRC NE 0.
  ...
ENDIF.
```

© SAP AG 2004

THE BEST-RUN BUSINESSES RUN SAP



- In the AUTHORITY-CHECK you have to provide values for all fields of the authorization object. Otherwise you will get a sy-subrc ne 0. If you do not want to check one of the fields you have to provide DUMMY as value:  
Example: For a changing transaction it makes sense to check whether the user is allowed to change entries for at least one carrier:  
AUTHORITY-CHECK OBJEKT 'S\_CARRID'  
ID 'ACTVT' FIELD '02'  
ID 'CARRID' DUMMY.
- Important Return Codes of AUTHORITY-CHECK are:
  - 0: The User has an Authorization with the supplied values.
  - 4: The User does not have an Authorization with the supplied values.
  - 8: You did not supply values for all fields of the Authorization Object. Therefore the Check could not be successful.
- After FIELD you can provide only a single field, not a selection table. For this purpose there are function modules that perform then AUTHORITY-CHECK for all Values in the selection table.

### Automatic authorization checks

- Start of transaction                    S\_TCODE
- Start of report                         S\_PROGRAM
- Start of RFC function group         S\_RFC

### Further automatic kernel checks

- Trusted RFC FM start                 S\_RFACL
- File access                             S\_DATASET
- CPIC communication                 S\_CPIC

### Standard function modules of BC for authorization checks

- Function module AUTHORITY\_CHECK\_TCODE
- Function module VIEW\_AUTHORITY\_CHECK
- Function module AUTHORITY\_CHECK\_DATASET

- Some automatic authorization checks are already performed by the system:
  - S\_TCODE is checked on start of a transaction
  - S\_PROGRAM is checked at start of a report
  - S\_RFC is checked at start of a RFC Function Group (NOT at start of a specific RFC Function Module!)
- Further automatic kernel checks are performed:
  - S\_RFACL on trusted start of an RFC Function Module
  - S\_DATASET on File Access
  - S\_CPIC on CPIC Communication
- SAP Basis provides some standard function modules for Authorization Checks by the programmer
  - AUTHORITY\_CHECK\_TCODE
  - VIEW\_AUTHORITY\_CHECK
  - AUTHORITY\_CHECK\_DATASET

## Authorization Check at Start of Transaction (1)

- **Several of our customers' administrators use only this check. Other checks are disabled**
- **The profile generator uses transactions as starting point for generation of authorizations. Therefore the profile generator does not support reports and RFC function modules.**
- **For new single screen transactions which bundle several old transactions, the authorization concept usually has to be redesigned.**



- At start of a transaction, the following checks occur:
  - The system checks, whether the transaction exists in table TSTC
  - The system checks, whether the transaction has been locked (via transaction sm01).
  - If those two checks have passed, the system checks for authorizations of S\_TCODE in the user profiles
  - However, it is planned to support default authorization values for services like function modules.

- **This authorization check is done automatically at...**
  - Call via transaction code in the command field: /n...
  - Call via Menu, if function code is of Type T.
  - Call via ABAP command LEAVE TO TRANSACTION



### **This authorization check will NOT be performed at CALL TRANSACTION**

- **Usage of CALL TRANSACTION:**
  - ◆ For navigation purposes
  - ◆ To form modularization units
  - ◆ For automating tasks using BTC data
- **History:**

Navigation used to be provided with the help of menus and type T function codes.  
Today type ' ' function codes and CALL TRANSACTION is used to keep the navigation stack and enable return to the caller.

- These automatic checks are performed whenever
  - The user enters the transaction code in the command field
  - The user starts the transaction by triggering a function code of type T (from the GUI)
  - The ABAP command LEAVE TO TRANSACTION is executed
- Automatic checks **WILL NOT BE PERFORMED** if the transaction is called with CALL TRANSACTION!!!
  - Therefore the developer has to use function module `AUTHORITY_CHECK_TCODE` before CALL TRANSACTION
  - Or since release 4.6 table `TCDCOUPLES` must be maintained via SE97 and all transactions `Bn`, which are called by a transaction `A`, have to be entered into this table. Note 0367547 describes that CALL TRANSACTION checks, whether a pair of calling and called transactions are entered in table `TCDCOUPLES`. Four cases are distinguished:
    - The record does not exist: It is written into the table and Field `OKFLAG` is set to `SPACE` (Trace Functionality is switched on). Authorization to start the called transaction is not checked.
    - The record does exist, and `OKFLAG` is `'X'`: The authorization to start the called transaction is checked.
    - The record does exist, and `OKFLAG` is `'N'`: The authorization to start the called transaction is not checked.
    - The Record exists and `OKFLAG` is `SPACE`: The authorization to start the called transaction is not checked.
  - The entries in `TDCOUPLES` are maintained with transaction SE97, which can be reach from SE93 (Transaction maintenance) via Utilities->Authorization for called transactions. Transaction SE97 is documented in SAP Note 358122.

- Before using CALL TRANSACTION, the developer has to perform an authorization check





- Instead of AUTHORITY-CHECK OBJECT S\_TCODE use function module AUTHORITY\_CHECK\_TCODE.



- Alternatively you can use function module ABAP4\_CALL\_TRANSACTION

```
CALL FUNCTION 'AUTHORITY_CHECK_TCODE'  
  EXPORTING  
    TCODE = 'SM59'  
  EXCEPTIONS  
    OK      = 0  
    NOT_OK = 1  
    OTHERS = 2.  
IF SY-SUBRC NE 0.  
  MESSAGE E172(00) WITH 'SM59'.  
ENDIF.
```

- As described before, no automatic authority check is performed by the system on CALL TRANSACTION.
- Therefore, the developer has to perform an Authorization Check on Authorization Object S\_TCODE.
- Alternatively, the developer can use either function module AUTHORITY\_CHECK\_TCODE or call the transaction with the help of function module ABAP4\_CALL\_TRANSACTION.

- **Special case report transactions**
  - Usually a report can be executed directly. Therefore you cannot rely on the automatic authorization check at start of the report transaction.
  -  Therefore authorization checks have to be programmed in each report.
- **Special case parameter transactions**
  - Parameter transactions provide values for fields on the entry screen
  -  Automatic authorization check is performed for the transaction code of the parameter transaction only. The base transaction itself will not be checked! (See note 67766)
- **Special case variant transaction**
  - Allows modifications of one or more Dynpro
  - Automatic authorization check is performed for both variant transaction and the base transaction. (See note 109182)

- There are some special cases to automatic authorization check at start of a transaction.
- Report Transactions: Since reports can be executed directly, you cannot rely on the automatic authorization check at start of the report transactions. Therefore, authorization checks have to be programmed in each report.
- Parameter Transactions: They provide values for fields on the entry screen. Automatic authorization check is performed for the transaction code of the parameter transaction only! The base transaction itself will not be checked!!!
- Variant Transactions: They allow modifications of one or more Dynpro. Automatic authorization checks are performed for both the variant transaction and the base transaction.

## Authorization Check at Start of Report (1)

- **There is no fundamental difference in functionality between reports and transactions**
- **Therefore our customers' administrators also have to secure reports**
- **Automatic authorization check is done during SUBMIT against the report authorization group. This check cannot be bypassed, but it may be without effect.**
- **Mapping of reports to authorization groups**
  - **Is done with report RSCSAUTH by the customer**
  - **SAP does not deliver a template**

© SAP AG 2004

THE BEST-RUN BUSINESSES RUN SAP



- Since there is no fundamental difference in the possible functionality between reports and transactions, our customers' administrators also have to secure reports.
- An automatic check is done during SUBMIT against the report authorization group.
- A report is mapped to a report authorization group with report RSCSAUTH. This is done by the customer and SAP delivers only few pre-defined report groups and only few reports are assigned to those. The assignments are stored in table SREPOATH.



### Alternative: Reporting Trees

- No individual mapping to report authorization groups
- Standard Transaction for submitting reports (SA38 ...) must be locked
- Mapping of report authorization groups to higher levels in the reporting tree is possible
- But this is not sufficient!
  - ◆ Start of reports by applications
  - ◆ Name of reports in customizing tables (e.g. Report-report interface)
  - ◆ Customer has to customize all reporting trees (ca. 60 are delivered)



**Reports should not rely on the automatic authorization checks at start of a transaction or a report!**

- Report Trees provide an alternative, but they are used no longer and replaced either by area menus or PFCG generates a report transaction for every report in a role.
- And usually, using report trees is far from sufficient:
  - SA38 has to be disabled
  - Reports may be started by applications
  - Names of reports appear in customizing tables



- **ABAP commands: OPEN, READ, TRANSFER, CLOSE**



**These commands give access to the file system of the application server with the <sid>adm account.**

- **Authorization object S\_DATASET is checked during OPEN DATASET.**

- **But: This check is mostly without effect, because the authorization fields require strict restrictions which do not comply with the composition of path and file names.**



**Therefore you should use function module AUTHORITY\_CHECK\_DATASET for an authority check against authorization object S\_PATH.**

- The ABAP Commands OPEN, READ, TRANSFER, and CLOSE give an ABAP program access to files on the application server. The files are accessed with the privileges of the <sid>adm-account.
- The privileges of the <sid>adm-account are usually very far-reaching.
- Therefore, the authorization object S\_DATASET is checked during the OPEN DATASET operation.
- However: This check is usually without effect, because the authorization fields require strict restrictions which often do not comply with the composition of path and file names.
- Therefore you should use function module AUTHORITY\_CHECK\_DATASET for an authority check against authorization object S\_PATH.

- **BAPIs are implemented by RFC enabled function modules. These are covered in the next unit.**



**Each BAPI is a public entry point into an SAP system. It can be called directly and therefore has to perform suitable authority checks.**

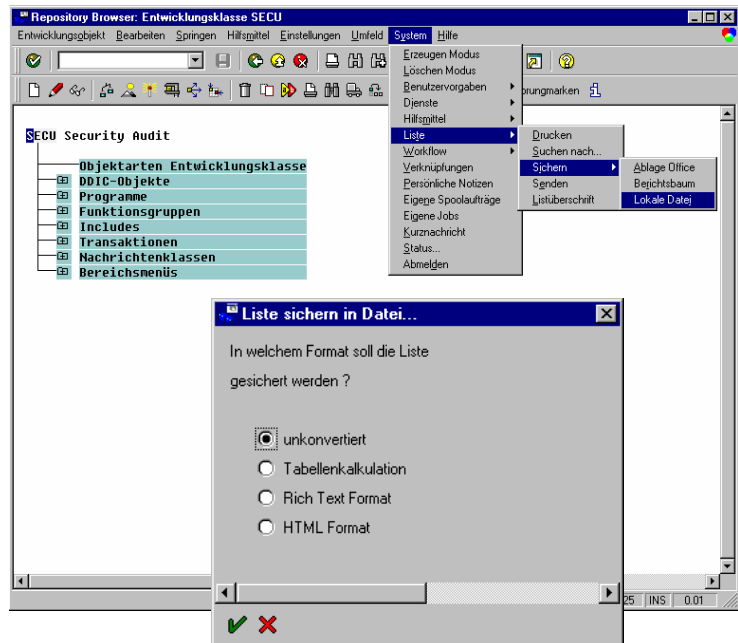
- BAPIs are public entry points into an SAP system, like transactions or reports. BAPIs can be called directly by remote clients and therefore have to perform suitable authority checks, as a transaction would do.
- BAPIs are implemented by RFC Function Modules. Therefore the same security guidelines hold for both RFC function modules and BAPIs. We will cover the security guidelines for RFC function modules in the next unit

## Upload/Download

- Download is possible for each List
- These function modules check against S\_GUI:  
WS\_DOWNLOAD  
WS\_UPLOAD  
GUI\_DOWNLOAD  
GUI\_UPLOAD  
LIST\_DOWNLOAD



You should perform these checks in your programs, too!



© SAP AG 2004


THE BEST-RUN BUSINESSES RUN SAP





- Download to the front-end is possible for each list which is displayed by the SAP GUI.
- The function modules WS\_DOWNLOAD, WS\_UPLOAD, GUI\_DOWNLOAD, GUI\_UPLOAD, and LIST\_DOWNLOAD check against authorization object S\_GUI.
- In your program, you should perform these checks, too!
- However: There is no way to prevent users to write down sensitive information on paper and sell this information outside. Or to disable Copy-and-Paste on the Users' workstation. This clearly shows, that security is not merely a technical problem

- **Standard transactions of customizing: SE16, SM30, SCU0 ...**
- **Authorization objects S\_TABU\_DIS and S\_TABU\_CLI as well as function module VIEW\_AUTHORITY\_CHECK is used by these transactions**

 **The mapping of authorization groups to customizing tables and views is done via view maintenance of view V\_DDAT**

 **If you implement special customizing transactions you should perform the same checks!**

- There are some standard transactions used for customizing the contents of tables. These transactions include SE16, SM30, SCU0 and many others.
- These transactions check against authorization objects S\_TABU\_DIS and S\_TABU\_CLI. They also use the function VIEW\_AUTHORITY\_CHECK to perform authorization checks.
- The mapping of authorization groups to customizing tables and view is done via view maintenance of view V\_DDAT.
- If you implement special customizing transactions you should perform the same checks!
- By all means avoid undocumented and unchecked switching from table display to table maintenance, like it is done for example in SE16N.

- **Access to other clients or system via RFC**
- **There are many examples of these RFC enabled function modules: GET\_TABLE\_RFC ...**
-  **In your own RFC enabled function modules you should usually perform the same checks by calling VIEW\_AUTHORITY\_CHECK!**
-  **In addition, you should check authorization object S\_TABU\_RFC!**

- There are quite a few RFC function modules that give access to table contents. One example is GET\_TABLE\_RFC.
- If you build similar functions you should perform the same checks as when writing customizing transactions: You should perform a CALL to function module VIEW\_AUTHORITY check.
- In addition, you should check authorization object S\_TABU\_RFC.

## Agenda

- Introduction
- General Rules
- Extract from Top Ten Rules
- Authentication
- Authorization
- Auditing and Logging
- Sensitive Information
- Cryptography
- Deployment
- Summary

© SAP AG 2004

THE BEST-RUN BUSINESSES RUN SAP

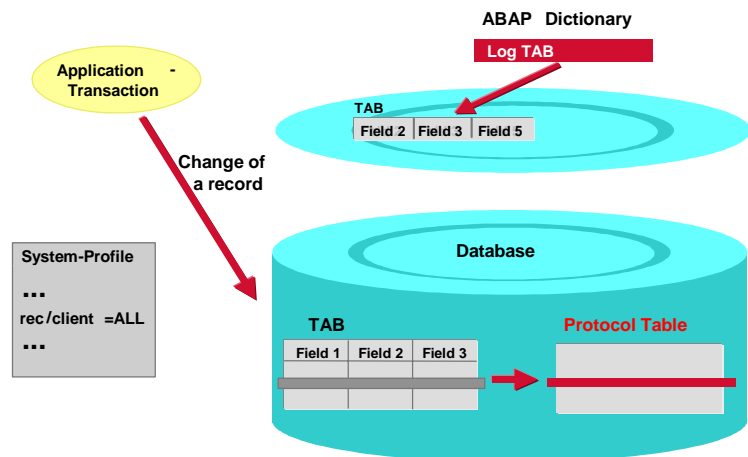


## Table logging

- By using table logging, changes to the records of a table can be logged.
- This is useful for customizing tables



Data dictionary → technical settings  
Set flag for logging



© SAP AG 2004

THE BEST-RUN BUSINESSES RUN SAP



- Using the logging flag you can define whether changes to the data records of a table should be logged. If logging is switched on, each change to an existing data record (with UPDATE, DELETE) by the user or application program is recorded in the database in a log table (DBTABPRT).
- To switch on logging, the R/3 System must be started with a profile containing parameter rec/client. This parameter defines whether all clients or only selected clients should be logged.
- The parameter can have the following values:
  - rec/client = ALL                      Log all clients.
  - rec/client = 000[...]      Log the specified clients.
  - rec/client = OFF                      Do not log.
- Logging slows down accesses that change the table. First of all, a record must be written in the log table for each change. Secondly, a number of users access this log table in parallel. This can cause lock situations although the users are working with different application tables.
- Logging is independent of the update.
- The existing logs can be displayed with Transaction Table History (SCU3).
- Attention! Using logging creates a bottleneck in the system:
  1. There is additional write access for each changing access to a logged table.
  2. The logging table is accessed by many users in parallel. Therefore you get lock situations even if users access different application tables!

## Agenda

- Introduction
- General Rules
- Extract from Top Ten Rules
- Authentication
- Authorization
- Auditing and Logging
- Sensitive Information
- Cryptography
- Deployment
- Summary

© SAP AG 2004

THE BEST-RUN BUSINESSES RUN SAP





- **Must be used to save passwords from internal applications accessing external applications for example access to an LDAP directory**
- **New feature in release 4.6C**
- **Kernel prohibits that customers can access the secure storage via function modules for example**
- **The report SECSTORE01 is a demo report containing all function modules that developers can use to store data in the secure storage**
- **It is forbidden by German export law to store cryptography or anything else in this secure storage than passwords**

- A feature introduced in release 4.6C is secure storage.
- Secure storage is used to store passwords which can be used by the kernel and by ABAP.
- This technique is of course not good for encrypting application data. It is just meant to securely store passwords.

## Agenda

- Introduction
- General Rules
- Extract from Top Ten Rules
- Authentication
- Authorization
- Auditing and Logging
- Sensitive Information
- Cryptography
- Deployment
- Summary

© SAP AG 2004

THE BEST-RUN BUSINESSES RUN SAP



### Used to secure SAP data and documents when...

- Data leaves the SAP system  
**(electronic orders, payments, business information)**
- Data is saved to insecure devices  
**(external database, diskettes, archives)**
- Data is transmitted via insecure networks  
**(for example: the Internet!)**
- Data need to be mapped to specific persons  
**(digital signatures)**



**SAP technology supports PKI by accepting X.509 certificates and by offering function modules for digital signatures.**

- Data may leave or enter an SAP system via various ways:
  - IDocs carrying electronic orders, payments or other business information.
  - Files stored on disks, archives, or databases
  - Data transmitted via insecure networks.
- As soon as SAP data and documents leave the SAP system, you no longer can control access to these data and documents. Therefore, you need to make sure that:
  - Sensitive Data are encrypted so that only the true addressee can read them.
  - Data leaving the system can be signed as to prove their origin.
  - Data entering the system carry a signature that proves their origin.
- Secure Store and Forward (SSF) offers a framework to digitally sign data. These functions are implemented by external security providers and are used by calling function modules.
- SNC offers a framework to encrypt network paths between SAP servers, to the SAP router, to the printer.

## SSF ABAP Function Modules

<b>SSF_SIGN</b>	<b>create digital signature(s)</b>
<b>SSF_VERIFY</b>	<b>verify digital signature(s)</b>
<b>SSF_ENVELOPE</b>	<b>encrypt for recipient(s)</b>
<b>SSF_DEVELOPE</b>	<b>decrypt for recipient</b>
<b>SSF_ADDSIGN</b>	<b>add a digital signature</b>
.....	
<b>SSFS_CALL_CONTROL</b>	<b>starts the signature control</b>
<b>SSFS_GET_SIGNATURE</b>	<b>gets the signature value from the control</b>
...	
<b>SSF_KRN_...</b>	<b>done directly by the AS</b>

© SAP AG 2004

THE BEST-RUN BUSINESSES RUN SAP



- This is a list of SSF ABAP function modules:
- SSF\_SIGN is used to create one or more digital signature used to sign documents before sending them.
- SSF\_VERIFY is used to verify one or more digital signatures of received documents are valid.
- SSF\_ENVELOPE is used to encrypt outgoing documents.
- SSF\_DEVELOPE is used to decrypt incoming documents.
- SSF\_ADDSIGN is used to add a digital signature to an outgoing document
- To start signature control use function module SSFS\_CALL\_CONTROL. To get the signature value from the control, use SSFS\_GET\_SIGNATURE
- Usually, you do not directly call the various SSF\_KRN\_xxx function modules. These are used directly by the application server.

### **Application server has a public key for signing**

- Archivelink II / SAP Content Server  
**(Signed URLs for access control)**
- SAP Business Connector  
**(Orders are signed)**
- Web Single Sign-On  
**(Digital certificates for user logon)**

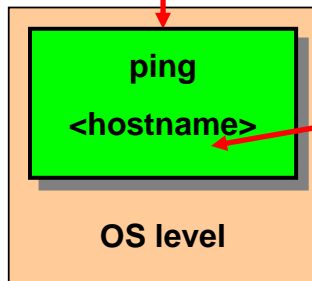
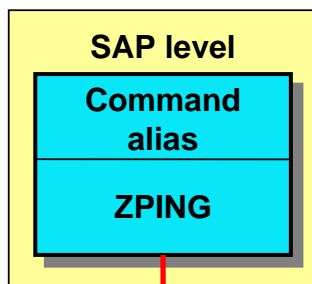
### **Individual users have their own public key**

- **Product Data Management (PDM)**
- **Process Industry (PP-PI)**
- **Quality Management (QM)**

- There are several applications, that use digital signatures. These applications can be classified in two groups:
- The application publishes a key and signs documents with its private key:
  - Archivelink II /SAP Content Server. These use signed URLs for access control
  - SAP Business Connector: Orders are signed
  - Web Single Sign-On: Uses digital certificates for user logon
- Individual users publish a key and sign documents with their private keys:
  - Product Data Management (PDM)
  - Process Industry (PP-PI)
  - Quality Management (QM)

- Introduction
- General Rules
- Extract from Top Ten Rules
- Authentication
- Authorization
- Auditing and Logging
- Sensitive Information
- Cryptography
- Deployment
- Summary

## External Commands: Definition



The screenshot shows the SAP configuration screen for the external command "ZPING". The command name is "ZPING", the operating system is "Windows NT", and the type is "Customer". The operating system command is "cmd /c ping". The parameters for the operating system command are empty. The "Additional parameters allowed" checkbox is checked and circled in red. The "Trace" checkbox is unchecked. The "Check module" field is empty. The "Created by" and "Last changed by" fields both show "TUBBESING" on "18.11.2002 17:29:10". A status bar at the bottom indicates "Changes were made" and shows transaction codes "SM69" and "twdfmx04 INS".

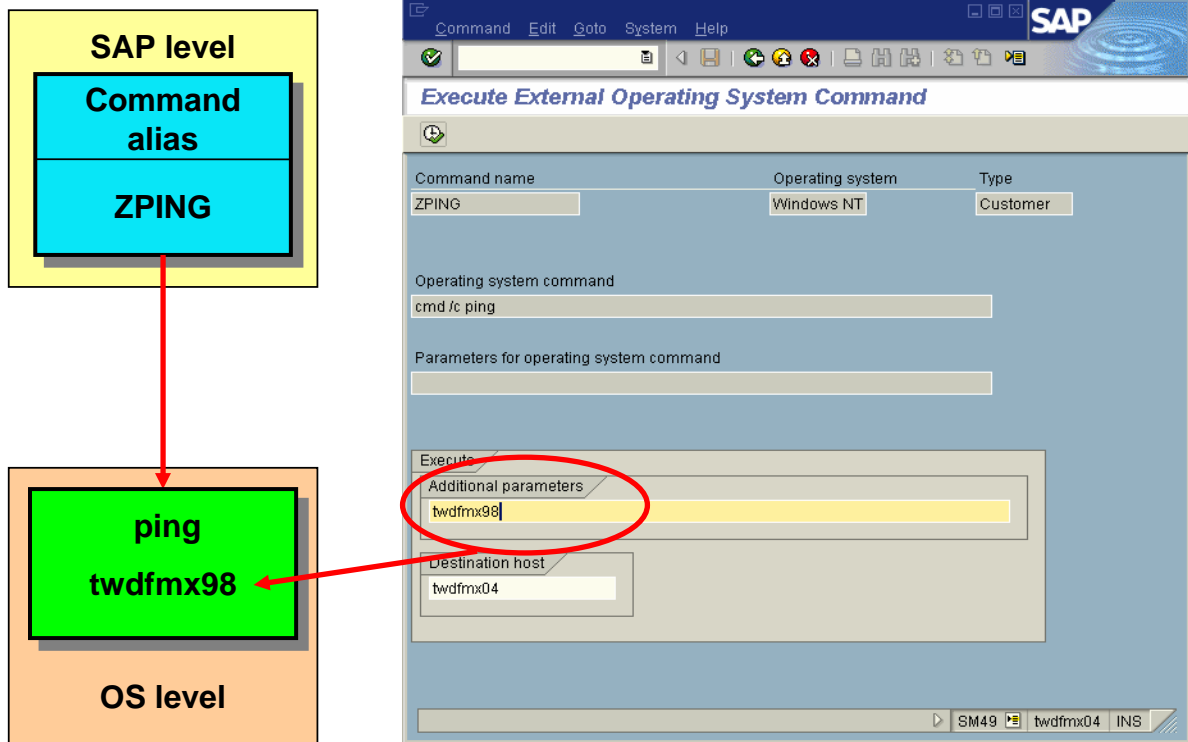
© SAP AG 2004

THE BEST-RUN BUSINESSES RUN SAP



- An external command is an alias defined in the SAP system that represents an operating system command.
  - For example, you can define the external command ZPING, which represents the operating system command ping <hostname>.
  - The possible set of commands is restricted to the ones defined in the SAP system.
  - External commands are maintained using transaction SM69 (maintain external commands) and executed using transaction SM49 (execute external commands).
- Both the maintenance and the execution of external commands are protected by authorization objects.
- The execution of external commands is checked by the authorization object S\_LOG\_COM.
- For the maintenance of external commands, you need an additional authorization based on the authorization object S\_RZL\_ADM with activity 01 and 03.
- You can also include a check module (function module) to validate the additional parameters

## External Commands: Execution in Dialog



© SAP AG 2004

THE BEST-RUN BUSINESSES RUN SAP



- Before you execute the command, you can specify additional parameters, as well as the name of a target server.
- External commands can also be executed in ABAP programs using special function modules or as a step in a background job.
- The execution of external commands is checked by the authorization object S\_LOG\_COM. This object has three fields command, operating system, and host, where the administrator can specify which command can be executed for which operating system on which host.
  - Administrators must control who has authorization based on authorization object S\_LOG\_COM, since programs can be accessed at the operating system level.



### RFC and Distributed Systems

- Examples
- Principle of remote function calls
- RFC with remote dialogs
- Scope of remote function calls
- Logon data in RFC customizing
- Trusted system RFC
- RFC function group start check
- Callback mechanism
- Data transfer and processing

## Examples

- **ALE**
- **TMS (Transport Management System)**
- **Workflow**
- **CUA (Central User Administration)**
- **BW (Business Warehouse)**

© SAP AG 2004

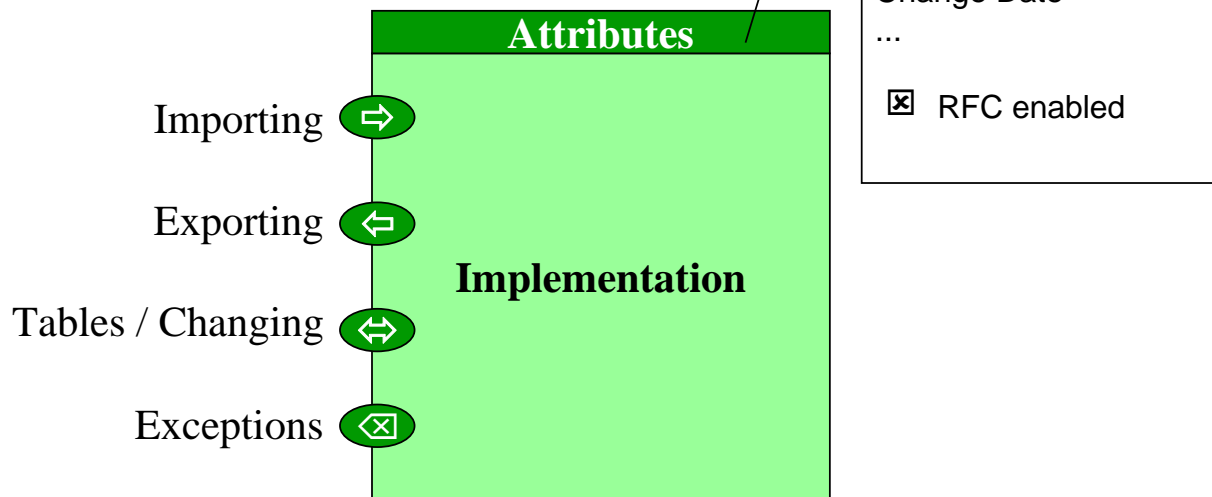
THE BEST-RUN BUSINESSES RUN SAP



- There are many, many applications and uses for remote function modules. Examples include, but are not limited to, the following:
- ALE uses the RFC function module IDOC\_INBOUND\_ASYNCHRONOUS to receive IDocs.
- TMS, the Transport workflow management system uses RFC function modules to distribute transport files.
- Workflow uses BAPIs, which are RFC function modules.
- Central user administration uses RFC to synchronize user information.
- BW, Business Warehouse uses RFC to extract data from the backend systems for reporting.
- These are just a few examples, there are many, many more!

Interface for passing data and events

RFC enabling via an attribute



© SAP AG 2004

THE BEST-RUN BUSINESSES RUN SAP



- A function module is a modularization unit of the ABAP programming language. A function module can be called with the CALL FUNCTION statement. Data are passed to and received from the function module via parameters passed to its interface.
- RFC can be enabled at any time without affecting the interface or implementation by just setting a flag.
- The display of Dynpros in the implementation bypasses the interface and violates its contract. This should be avoided as explained in the next slide.

## RFC with Remote Dialog

→ These commands in the implementation part will display

### Dynpros

- CALL SCREEN
- CALL TRANSACTION
- SUBMIT AND RETURN
- CALL DIALOG



**Avoid if possible!**

### Resource usage (tunneling)

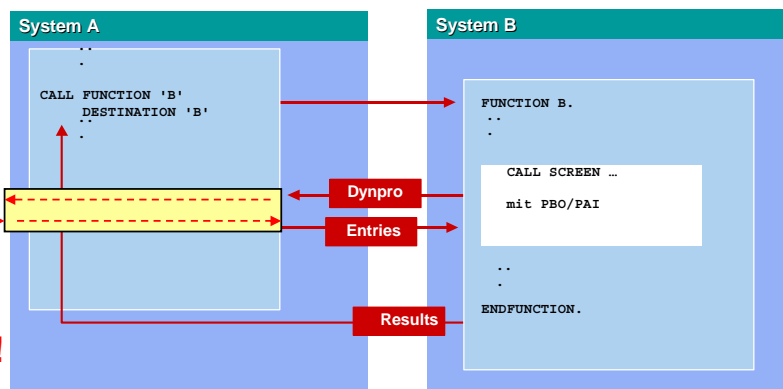
Time zone, currency, ...



**Dialog user needed!**



**Interface is bypassed!**



© SAP AG 2004

THE BEST-RUN BUSINESSES RUN SAP



- The following commands in the implementation part of a function module will display Dynpros:
  - CALL SCREEN
  - CALL TRANSACTION
  - SUBMIT AND RETURN
  - CALL DIALOG
- These techniques should be avoided, because there are several draw-backs involved:
  - They use a lot of resources, because the Dynpro and the user entries have to be tunneled between the two systems.
  - There are issues with the time zone, currency, display of numbers and so on.
  - A dialog user is needed!
  - The interface is bypassed!!!

## Scope of Remote Function Calls (RFCs)

1) local RFCs

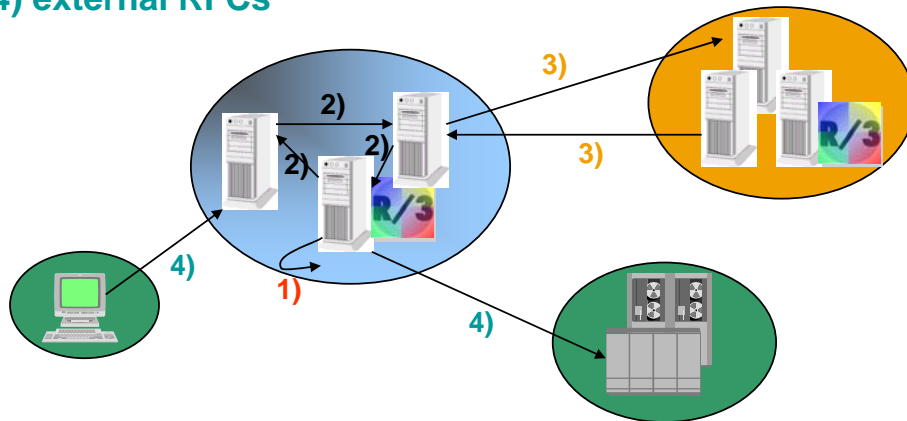
2) internal RFCs

3) trusted RFCs

4) external RFCs



RFC enabling of a function module allows all these scenarios!



Authority checks should always take the external scenarios into account!

© SAP AG 2004

THE BEST-RUN BUSINESSES RUN SAP



- Any RFC enabled function module can be called by any client.
- Therefore you should always take the most general case (external RFC) into account. An RFC function module is an access point to the system and has to be treated in the same way as transactions or reports.

**RFC is a system access ⇒ authentication necessary**

**Logon data can be stored in RFC customizing SM59**



→ Can be used by anybody!



→ Makes sense only for service users

→ It must be possible to (extremely) restrict authorizations



→ No generic functionality must be offered!

**Authentication will be “bypassed” ...**

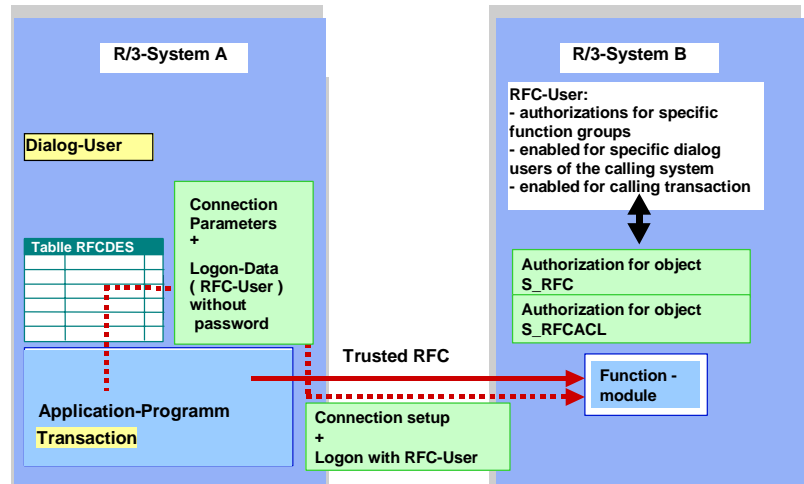
→ In trusted system landscapes

→ By using external security products (SNC)

- RFC provides system access and therefore authentication is necessary.
- Administrators use SM59 to set up remote destinations. In SM59, logon data are stored to log on to the destination system. This has severe draw-backs:
  - The destinations maintained with SM59 can be used by anybody!!! Therefore anybody can gain access to the remote system with the user maintained in SM59!!!!
  - In SM59 it makes only sense to use service users of the remote system. Do not use dialog users!!!!
  - These service users need to have extremely restricted authorizations in the remote system.
  - No RFC function module whatsoever must offer generic functionality, e.g. functionality where table names to be accessed are passed as parameters.
- Authentication in the remote system with a user provided in SM59 of the calling system will be performed different in the following scenarios:
  - In trusted system landscapes (see next two slides)
  - If external security products (SNC, Secure Network Connection) are used.

## Trusted System RFC

- Different SAP systems may be coupled into one RFC context. The calling system is customized as “trusted system” in the target system. Trusted systems can log on to a trusting system without providing a password.
- Trusting/Trusted-relations are defined with transactions SMT1 and SMT2.



© SAP AG 2004

THE BEST-RUN BUSINESSES RUN SAP



- Authorization object S RFCACL is relevant only in connection with trusting/trusted relationships.
- Authorizations of this objects define per RFC User in a trusting system, which dialog users of the trusting system may use those RFC users. Furthermore the authorizations specify, from which transactions in the trusted system these users may be used.
- S RFCACL has 7 fields:
 

RFC_SYSID	ID of the calling system
RFC_CLIENT	client of the calling system
RFC_USER	ID of the calling user
RFC_EQUUSER	Flag specifying, weather the RFC user may be used by a dialog user with same ID
RFC_TCODE	Calling transaction code
RFC_INFO	Additional information of the calling system (not used)
ACTVT	Only allowed value is 16 (Execute)
- This authorization object is not linked to the profile generator. The administrator has to maintain authorizations for this object manually.

## RFC function group start check

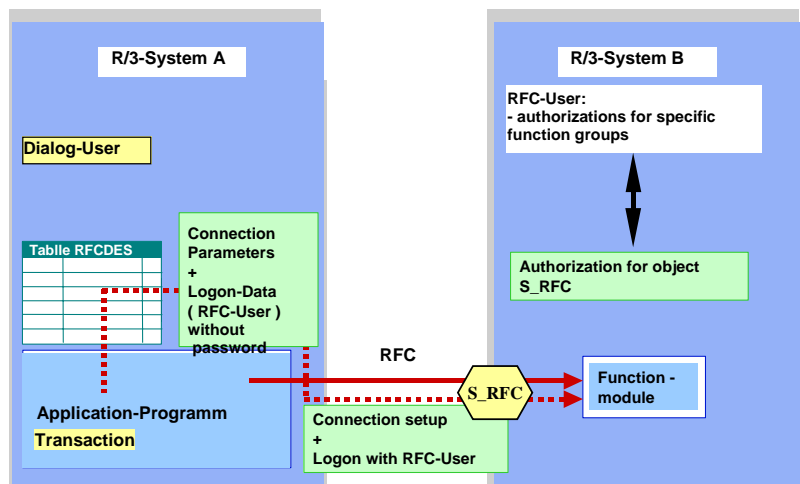
- The authorizations of authorization object S RFC define for each RFC user, which function groups may be used by this user.
- By default this check is active, but can be disabled with a profile parameter.



**The administrators need documentation of the function module in order to activate this check!**



**You need to document which RFC function modules are needed in a specific scenario**



© SAP AG 2004

THE BEST-RUN BUSINESSES RUN SAP



- Authorization object S RFC has 3 fields:
  - RFC\_NAME name of the function group
  - ACTVT only possible value is 16 (execute)
  - RFC\_TYPE only possible value is FUGR (function group)
- This object is not linked to the profile generator. Administrators have to create authorizations for this object manually.
- Administrators need to be aware of the fact, that certain function groups are needed for a certain scenario. They need exact documentation of the function modules needed in these scenarios.
- Application developers need to provide a standard role for the scenario: They should trace, which authorizations are needed exactly, to perform the function, and put these authorizations into the standard role. As note 0543164 states, authorizations are now also traced for RFC function modules, if the profile parameter auth/authorization\_trace is set to y.



## Callback Mechanism

- Possible with synchronous RFC calls:

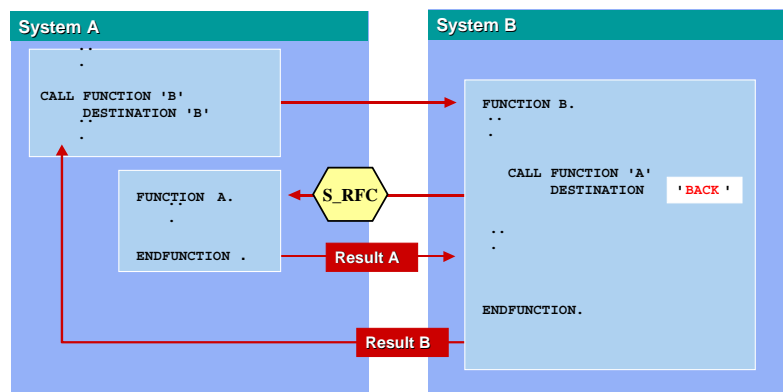


- “Client” (System A) becomes “Server”!

→ Administrators have to protect against unwanted RFC calls with S RFC authorizations for users of system A



**Avoid callbacks  
in your design!**



© SAP AG 2004

THE BEST-RUN BUSINESSES RUN SAP



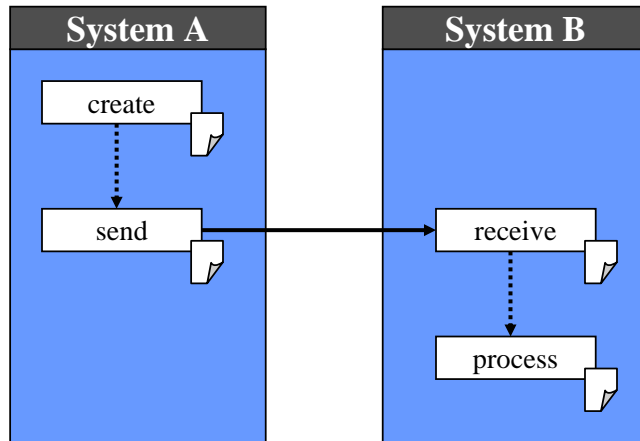
- There are a number of destinations with defined functions:
- BACK: Use this destination to call back to an RFC caller during processing the implementation of an RFC function module.(only possible for synchronous RFC).  
SPACE: The function module will be executed locally.  
NONE: The function module will also be executed locally, but by communication via the gateway.
- Avoid callbacks in your design, and if you absolutely need to do so, document it clearly!
- Administrators need to protect against unwanted RFC calls with S RFC authorizations for users of system A!

### General Principle: Separate

- Data transfer
- Data processing



- different authorizations
- service users possible
- callbacks avoidable



© SAP AG 2004

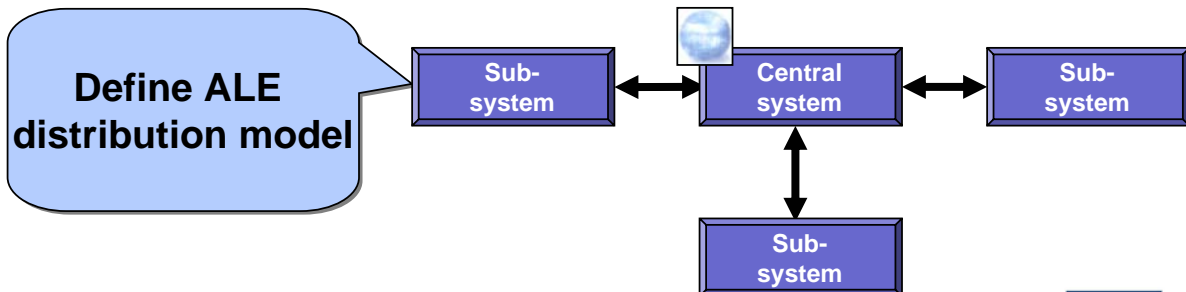
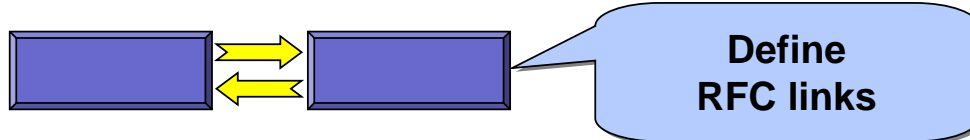
THE BEST-RUN BUSINESSES RUN SAP



- As a general principle, separate data transfer from data processing. This has several advantages:
  - To send data (which is usually by calling a receive function module in the receiving system) much fewer authorizations are needed.
  - To send data a simple service user can be provided.
  - Callbacks are avoidable with this design.

## ALE Setup

- Definition of logical systems
- Assignment to clients



© SAP AG 2004

THE BEST-RUN BUSINESSES RUN SAP



- Communications partners are addressed in the ALE scenario with aliases, which are called logical systems. The central system itself and every sub-system is defined by name in the central system in transaction BD54. The sub-system itself and the central system are defined in the sub-systems. The logical system names are assigned to the client definitions in the corresponding systems in transaction SCC4. Each logical system also identifies a certain client of an R/3 system.
- Communications between the central system and the sub-systems uses the network with an RFC. The technical definition of the link is maintained in transaction SM59. All the links to the sub-systems must be maintained in the central system plus a link to the central system itself, and the link to the central system must be maintained in the sub-systems. The RFC link names must be the same as the names of the logical systems. Communications should never be based on users with SAP\_ALL authorization in the target system. You should always deliver either a (number of) pre-defined profiles or roles that contain exactly the intended functionality to be executed and nothing else.
- What data is sent from where to where is defined in the ALE distribution model. The distribution model is maintained, generated and distributed in the central system in transaction BD64. It only has to be generated in all the sub-systems.

## Protecting ALE Applications

- **General recommendations**
  - **Set up special users in target system for using ALE**
  - **Assign user type communication**
  - **Restrict access to distribution model**
  - **Protect RFC connection**
- **Background processing**
- **No application authorizations required**
- **Requires authorization to receive IDocs (object B\_ALE\_RECV)**
- **Immediate processing**
- **Assign only application authorizations required to process IDoc**
- **Restrict access to necessary function groups via authorization object S\_RFC**

© SAP AG 2004

THE BEST-RUN BUSINESSES RUN SAP



- To enable communicating over RFC, you need to enter a user and his logon information in the RFC destination of the sending system. To keep the risk of an improper use to a minimum:
  - Use special users of type communication or service in the target system. These users cannot be used to execute dialog transactions.
  - Always restrict authorizations of these users in the target system
- If a non-SAP system sends IDocs to a SAP system using (transactional) RFC, it must also send a SAP user and password. In this case, the user and password are stored outside. Ensure that this information is protected appropriately.
- You only need application authorizations in the target system if IDocs have to be processed immediately. If IDocs are processed as a background job the ALE user needs only authorizations for receiving IDocs (authorization object B\_ALE\_RECV). This object contains the field EDI\_MES which enables you to specify the message type that the user is authorized to receive.
- If inbound IDocs have to be transferred immediately to the application, the ALE user should only be assigned those application authorizations required to post the application document from the IDoc. Perform a trace (transaction ST01) to determine which authorizations are needed.

## Agenda

- Introduction
- General Rules
- Extract from Top Ten Rules
- Authentication
- Authorization
- Auditing and Logging
- Sensitive Information
- Cryptography
- Deployment
- Summary

© SAP AG 2004

THE BEST-RUN BUSINESSES RUN SAP



**Our customers are concerned about the security of SAP systems**

**All developers (-ABAP and others-) are responsible for secure applications**

- ➔ **Contribute to the quality of your products by providing secure mechanisms**
- ➔ **Use the help of the responsible persons in security development**

- Our customers and the public are concerned about the security of SAP systems.
- We as ABAP developers are directly responsible for delivering secure applications.
- Therefore, we have to contribute to the quality of our products by providing security features.
- Please, do not hesitate to use the help of those who are responsible for security development.
- Also check out the sites on <http://service.sap.com/security>

## Security Homepage

http://service.sap.com/security

© SAP AG 2004

THE BEST-RUN BUSINESSES RUN SAP

- Additional information can be found on [sapneth1.wdf.sap.corp/security](https://sapneth1.wdf.sap.corp/security). Here we have the following topics:
- News provide latest information on recent developments.
- SAP Security Notes gives access to many security related notes.
- Security in Detail contains detailed Security Guides for Solution Production. These are How-To documents covering topics like secure programming, cross-site scripting, user management and much more.
- Auditing and Revision provides information about how to enable auditing in your application and how to provide means to allow revision of data. This includes information on change documents.
- Data Protection and Privacy provides information about how to secure data leaving the SAP system. This includes SSF.
- FAQs is a list of frequently asked questions plus answers.
- Security Partners gives some information about partners of SAP in developing and providing security infrastructure.
- Education & Workshops contain training material plus information on upcoming Seminars.
- The Media Library contains relevant documents, slide-shows and other media.

- No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.
- Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.
- Microsoft, Windows, Outlook, and PowerPoint are registered trademarks of Microsoft Corporation.
- IBM, DB2, DB2 Universal Database, OS/2, Parallel Sysplex, MVS/ESA, AIX, S/390, AS/400, OS/390, OS/400, iSeries, pSeries, xSeries, zSeries, z/OS, AFP, Intelligent Miner, WebSphere, Netfinity, Tivoli, and Informix are trademarks or registered trademarks of IBM Corporation in the United States and/or other countries.
- Oracle is a registered trademark of Oracle Corporation.
- UNIX, X/Open, OSF/1, and Motif are registered trademarks of the Open Group.
- Citrix, ICA, Program Neighborhood, MetaFrame, WinFrame, VideoFrame, and MultiWin are trademarks or registered trademarks of Citrix Systems, Inc.
- HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.
- Java is a registered trademark of Sun Microsystems, Inc.
- JavaScript is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.
- MaxDB is a trademark of MySQL AB, Sweden.
- SAP, R/3, mySAP, mySAP.com, xApps, xApp, SAP NetWeaver and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary.
- These materials are subject to change without notice. These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.