Adabas Security

This section describes general considerations for database security and introduces the security facilities provided by Adabas and the Adabas subsystems. Detailed information about the facilities discussed in this section may be found in other parts of this documentation and in the Adabas Security documentation.

This chapter covers the following topics:

- Security Planning
- Password Security
- Security by Value
- Ciphering
- Adabas SAF Security
- Natural andAdabas Online System Security

Security Planning

Effective security measures must take account of the following:

- A system is only as secure as its weakest component. This may be a non-DP area of the system: for example, failure to properly secure printed listings;
- It is rarely possible to design a "foolproof" system. A security system will probably be breached if the gain from doing so is likely to exceed the cost;
- Security can be expensive. Costs include inconvenience, machine resources, and the time required to coordinate the planning of security measures and monitor their effectiveness.

The cost of security measures is usually much easier to quantify than the risk or cost of a security violation. The calculation may, however, be complicated by the fact that some security measures offer benefits outside the area of security. The cost of a security violation depends on the nature of the violation. Possible types of cost include

- loss of time while the violation is being corrected;
- penalties under privacy legislation, breach of contracts, and so on;
- damage to relationships with customers, suppliers, employees, and so on.

Password Security

Password security allows the DBA to control a user’s use of the database by

- restricting the user to certain files;
- specifying for each file whether the user can access and update, or access only;
- preventing the user from accessing or updating certain fields while allowing access or update of other fields in the same file;
- restricting the user's view of the file to records that contain specified field values (for example, department code).

The DBA can assign a security level to each file and each field within a file. In the following table, *x/y* indicates the access/update security level. The value 0/0 indicates no security.

File	**Fields**	
1 (2/3)	AA (0/0)	BB (4/5)
2 (6/7)	LL (6/7)	MM (6/9)
3 (4/5)	XX (4/5)	YY (4/5)
4 (0/0)	FF (0/0)	GG (0/15)

A user must supply an appropriate password to access/update a secured file. In the following table, *x/y* indicates the password access/update authorization level.

	Passwords	
	ALPHA	**BETA**
File 1	2/3	4/5
File 2	0/0	6/7
File 3	4/5	0/0

Assuming the files, fields, and passwords shown in the above tables, the following statements are true:

- Password ALPHA
  - can access and update field AA in file 1, but not field BB;
  - can access and update all fields in file 3;
  - cannot access or update file 2.
- Password BETA
  - can access and update all fields in file 1;
  - can access all the fields in file 2 and can update field LL, but not field MM;
  - cannot access or update file 3.

- No password is required to access any field in file 4, or to update field FF.
- Field GG in file 4 can be read only. Its update security level is 15 and the highest possible authorization level is 14.

If password BETA can access a field that password ALPHA cannot, then password BETA can also access all the fields in the same file that password ALPHA can access. There is no way in which ALPHA can be authorized to access field AA but not field BB and password BETA to access BB but not AA. The same restriction applies to update (although not necessarily to the same combinations of fields or to the advantage of the same password). ALPHA could be permitted to update all the fields which BETA can update and some others which BETA cannot update.

This restriction does not apply to file-level security. For example, ALPHA can use file 3 but not file 2, and BETA can use file 2 but not file 3. When a record is being added to a file, Adabas only checks the update security level on those fields for which the user is supplying values. For example, the password ALPHA could be used to add a record to file 1 provided that no value was specified for field BB. This could represent the situation where, for example, a customer record is only to be created with a zero balance. For record deletion, the password provided must have an authorization level equal to or greater than the highest update security level present in the file. For example, an update authorization level of 9 is required to delete a record from file 2, and, it is not possible to delete records from file 4.

Security by Value

It is also possible to limit access/update fields within a file based on the contents of the field in the file. See the Adabas Security documentation for more information.

Ciphering

Adabas is able to cipher (encrypt) records when they are initially loaded into a file or when records are being added to a file. Ciphering makes it extremely difficult to read the contents of a copy of the database obtained from a physical dump of the disk on which the database is contained. Ciphering applies to the records stored in Data Storage only. No ciphering is performed for the Associator.

Adabas SAF Security

Adabas SAF Security, a selectable unit, can be used with Software AG’s Com-plete and with the following non-Software AG security environments:

- CA-ACF2 (Computer Associates);
- CA-Top Secret (Computer Associates);
- RACF (IBM Corporation)

For more information about Adabas SAF Security, contact your Software AG representative.

Natural andAdabas Online System Security

The Natural Security system may also be used to provide extensive security provisions for Adabas/Natural users. See the Natural Security documentation for additional information.

Access to the DBA facility Adabas Online System (AOS) can also be restricted. AOS Security requires Natural Security as a prerequisite.