SECURE YOUR
ADABAS & NATURAL ENVIRONMENT

Best Practices for Multi-Level Security

TABLE OF CONTENTS
2  Risk scenarios
3  Executive summary
4  Secure your data at rest
8  Protect data in flight
9  Control your application
11 Conclusion

Security on IBM® mainframe platforms, where many installations of Software AG’s Adabas and Natural reside, is considered robust. News stories about data breaches, hacking attempts and compromised customer information primarily seem to occur on other platforms and technologies where security practices are not as mature and established as those that apply to critical business systems requiring enterprise server processing. Indeed, security practices for mainframe platforms were well established by the time many of these newer client-server and Web-based systems were written in the 1980s and 1990s.

However, don’t be lulled into thinking the mainframe is invincible. The security risks facing these enterprise systems are increasing daily. Data is no longer locked away in a mainframe database, accessible only by a single enterprise application. More tools and applications are accessing and manipulating these volumes of data for analysis, intelligence and real-time processing. Enterprise systems are also more interconnected than ever, with businesses relying on networks and the Internet for daily operations.

In this white paper, we evaluate security practices relating to installations of Adabas & Natural and EntireX in today’s enterprise to help you:

• Better understand the scope of security risks
• Learn what best practices and tools are available to protect your Adabas & Natural environments from internal and external threats

For example, while nearly all Adabas & Natural sites have secured their Natural applications to limit access to authorized users with need-to-know-and-modify, their data remains at risk. If you develop code outside of Natural or use products that access Adabas through direct calls, your data is at great risk—Natural Security definitions do not apply in these cases. Without securing the database, users can gain full access to your data.
Secure Your Adabas & Natural Environment

We hope to help you realize potential security risks and encourage you to take proactive measures to mitigate such risks in order to preserve the reputation that Adabas & Natural installations are the most secure and reliable systems in the enterprise.

Risk scenarios

Do any of these stories resonate for you? Some are hypothetical, some have actually occurred. Each should inspire you to think about the risks that may exist in your own enterprise.

Unauthorized direct call to the database

A company’s core application has a 30-year history for data integrity. However, a recent audit revealed that some business transactions did not match the financial reporting of those transactions. Executives meet to decide how to break the news to their shareholders that the company will have to restate earnings for previous years and may potentially pay significant fines, penalties and back taxes.

Fortunately, when IT investigates, they find a non-Natural program written by a programmer from another department performed Adabas direct calls to update the transactions in question. The IT department is able to contain the program, restore the data and re-generate the reports confirming the original financial reports. Unfortunately, the reputation of this critical business system is never fully restored with the CIO and executive management.

Hacking of non-encrypted data

Hackers swiped 3.3 million unencrypted bank account numbers from a state government site. Equally worrisome is the theft of copies of 3.8 million tax returns, containing social security numbers for 1.9 million children and other dependents. The attacker gained access to department of revenue systems and databases with a stolen user’s credentials. Had the state encrypted sensitive information such as social security numbers, the data loss may not have been so great. The state reportedly has spent more than $20 million so far cleaning up the mess, including $12 million on credit monitoring services for affected citizens, and millions more on breach notification letters, security improvements, data forensics teams and IT consultants.

Lax life-cycle management oversight

In a South African government organization, a programmer was able to replace programs in production with fraudulent programs during the nightly runs. These fraudulent programs were skimming small amounts of money from all retirement accounts processed and putting those dollars into their own accounts, unnoticed. The situation was only discovered when a second participant forgot to replace the fraudulent programs with the originals after the nightly run. The total theft ran into the millions. It is notable that the DBA of this government organization had previously opted not to invest in a life-cycle management and version control tool that would have prevented this abuse.

Sensitive data in nonproduction environments

An organization’s Adabas & Natural system has tight controls on user access to data through profiles in RACF and Natural Security, thus limiting user access to sensitive data to certain business users. These controls were considered unbreakable until an employee’s salary and social security number were found in a Microsoft® Excel® file of a former employee who was not authorized to access that data.

Subsequent analysis revealed this former employee commonly accessed QA to provide user acceptance for project work delivered for his department. His security profile in QA did not prevent access to such sensitive information in that environment. Additionally, the production database was replicated to another platform that did not secure the sensitive information. He could have queried all data present there that also contained customer credit card data. This organization was greatly concerned over whether this former employee may have absconded with unknown amounts of sensitive data and its potential liability if he did.
Executive Summary

Multi-level security risks
The last few years have been a wake-up call for organizations to establish a comprehensive data security policy that addresses risks at multiple levels. Securing just the application is no longer good enough. Databases are now accessible from a multitude of applications through direct calls and desktop query tools such as Crystal Reports®, Excel® and Business Objects®. Data is also constantly in flight between various applications, platforms and users, making it susceptible to interceptions. And as distributed development and processing has grown, so too has the need to control and audit internal staff, including those who have access to production, test and development data.

Your data security policy for your Adabas & Natural environment must occur on multiple levels to address:

- Data at rest
- Data in flight
- Application life cycle

Data at rest
Today there are many Adabas applications written in Natural or other languages that tap into the database, especially SQL and direct call applications. To protect your data, you must secure the database where the data resides. This not only includes the production database but anywhere the data is housed including development, test, training or archives. And don’t forget to include other non-Adabas databases or targets where data is replicated.

Adabas SAF Security and the Adabas Security utility (ADASCR) are ideal candidates for securing access and authorization to your Adabas data. Encrypting protects the sensitive files of the database should unauthorized access occur, while Data Masking for Adabas ensures that data replicated in development, test or training environments is altered in such a way to make sensitive information such as names, salaries, addresses, phone numbers, credit card numbers and account numbers unusable.

And once your data is no longer in use in an active production database, it’s time to move it off. Data Archiving for Adabas moves data off production and places historical data in a storage “vault.” There you can apply additional security protections depending on your archive platform.

Data in flight
Because hackers are becoming increasingly sophisticated, standard database and application level security are simply not enough to protect your data. Data in flight (or in queue) is also at risk. Data moves between users, applications, platforms and the database in a multitude of ways, from integration servers, service buses, terminal emulation and client-server communications.

Someone simply pinging IP addresses (or ports) are looking for a way in. Once they find a successful hit, they will search for your weak spots and “listen” to everything you do. You cannot afford any gaps.

Security solutions that address your data in flight include EntireX SAF Security, Entire Net-work SSL, Entire Net-work SAF Security and Entire Connection, which leverages SSL and SSH.

Application life cycle
Securing user access to applications and data is complemented by ensuring that program functionality is not altered outside of documented business processes. Organizations that permit ad hoc updating of production data create risk of having unknown transactional updates or data access that bypasses all the usual controls. Best practices for complying with Sarbanes-Oxley (SOX) and other regulations include reviewing code migration policy and user access. This is best accomplished with a code migration tool that enforces implementation as well as centralized management of user access control.
Secure Your Adabas & Natural Environment

**Best Practice:**

**Centralize on SAF-based repository**

Use Adabas SAF Security, Natural SAF Security, Entire Net-work SAF Security and EntireX SAF Security to ensure all authentications are carried out against a central security repository where user IDs, passwords and encryption codes are stored.

**Natural Security** and **Natural SAF Security** enable you to protect your Natural environment against unauthorized access and improper use. **Adabas SQL Gateway** provides additional security when opening your data up to desktop tools and applications. To ensure the integrity of your applications, **Predict Application Control (PAC)** controls Natural, COBOL, JCL and Assembler applications throughout the software development life cycle.

**Best practices**

As the volume of personal data grows across industries and the number of data attacks on enterprises continues to increase, organizations large and small are seeking best practices on how to protect their data.

At each level described above, there are some common best practices that demonstrate how you can:

- Centralize management of security policy
- Control user access and authentication
- Encrypt sensitive data
- Mask data in nonproduction environments
- Determine policy for record retention, archival and destruction
- Manage the application life cycle

In the following pages, we will describe in greater detail how to specifically address data security for Adabas & Natural environments for data at rest, data in flight and managing the application life cycle.

**Secure your data at rest**

While products such as Adabas, Natural, EntireX and Entire Net-work each have security controls inside them that can be used by product administrators, it becomes difficult to manage user access, authentication and encryption if security definitions are maintained by many different teams of people. Because these products tend to be used by large numbers of users, centralizing the security definitions in a single repository will help administrators provide better security, more efficiently.

**Centralize security management**

Security teams generally include both mainframe and server components in one location, making it easier to manage definitions and report user access. On the mainframe, this is accomplished by leveraging IBM® System Authorization Facility (SAF) to interface with leading industry-standard security packages such as IBM RACF®, CA-ACF2® and CA-Top Secret®.

These security packages allow the system administrator to:

- Maintain user identification credentials such as user ID and password
- Establish profiles determining the datasets, storage volumes, transactions, reports and other resources available to a user
- Identify and verify system users
- Identify, classify and protect system resources
- Log and report unauthorized attempts at gaining access to the system and to the protected resources

With these security management packages, organizations can centralize user identity administration, provisioning and access management across the enterprise to improve IT efficiency, reduce IT costs and enhance user productivity.

We consider this a best practice for Adabas & Natural environments as it also enables security administrators to view consolidated cross-platform security events on industry-standard security packages, widely familiar and accepted by auditors and regulators, for enhanced auditing and compliance and faster response to security risks and incidents.
Imagine having to demonstrate to an auditor that you have completely revoked all access rights for a developer who left your company. Instead of visiting each product’s security package to disable his authorizations, a single trip to RACF, ACF or Top Secret can be made to disable his authorizations across the enterprise.

Leveraging SAF security repositories maximizes the investment that most z/OS sites have made in their mainframe security repository. And because these standard security repositories are recognized and proven across the industry, you can face regulators and auditors with confidence.

**Protect Adabas production data**

**Adabas SAF Security** puts your Adabas resources under the control of leading security packages used by most z/OS sites for rigorous control of system resources, including databases, files, commands, utilities, storage volumes and CICS transactions. By using definitions stored in RACF, ACF2 or Top Secret repositories, Adabas SAF Security verifies the requesting user’s authorization to access the database and file and what functions he may perform. Securitre is another product that provides similar functionality.

Users can also be identified based upon where the call originates, such as production or test and CICS or TSO/Batch. A distinction can be made even when two user IDs are the same. For example, USER1 originating in test is allowed to take an action under CICS, but not TSO.

Using the online administration program, which is a standard Natural application, you can monitor SAF operation in any nucleus. You can check cache efficiency, review parameters or even log users off, giving you the ultimate knowledge of what’s going on with your system and total control of your data assets.

**Secure at the field and value level**

With SAF security, you can secure at the database and file level. However, if you wish to manage and secure access down to the field and value level, you will want to add the **Adabas Security** utility (ADASCR). The use of a field within the file is restricted to those users having appropriate access/update profile definition and password. Access/update permission values are defined for each password. Each protected field is given equivalent access/update “threshold” protection values. Only a user whose permission value is equal to or greater than the protection level of the specified field is permitted to perform that operation on the file or field.

Value-level protection restricts the type and range of values that can be accessed or updated in specific fields. The restrictions are applied according to user password. They can be for specific values or value ranges and can either “accept” or “reject” criteria.

For example, you may allow everyone to see all the data except for “designated” information that can only be viewed or modified by management or designated individuals or groups. At the value level, users can be restricted to viewing only the data tied to particular value parameter. For example, users from London can only see London data, New York users only sees New York data, Tokyo users only see Tokyo data, but certain levels of management may see data from all regions.

To provide this level of authorization and access, all calls to the database (i.e., create, read, update and delete) have to pass the appropriate ADASCR password. If you use Adabas SAF Security with RACF in conjunction with ADASCR, the password will be stored and automatically passed, based on appropriate authentication, when needed. Unfortunately, this capability is not available for ACF2 or Top Secret.

Note: If you are unable to leverage RACF with Adabas SAF Security, Natural Security can store and pass the appropriate ADASCR password and encryption codes with Natural programs only. However, this approach will not protect your data from non-Natural program access such as direct calls and SQL.

**BEST PRACTICE:**

**Authenticate users with passwords**

Always implement ADASCR with Adabas SAF Security to ensure access and authorization is based on both the user and the password. ADASCR alone only assigns authorization to the password which could pose a risk if shared with other users.
**Single sign on**

Best practices are just being developed with regards to bringing Single Sign On (SSO) to the Adabas & Natural environment. There has been success in some sites with the incorporation of Kerberos as the central component to enterprise-wide authentication. Kerberos is a technology that has been utilized on IBM z/OS platforms with ties to MQ channel access, DB2 and to RACF, ACF2 and Top Secret through IBM Network Authentication’s ability to map Kerberos user IDs to the classic mainframe user IDs. SSO is a growing trend in enterprises and further fulfills the best practice of centralized user access control and security definition. Look for maturing best practices to be utilized in more sites in the future.

**Encrypt sensitive data**

Tightly managing who has authorized access to your data is not always enough. Encrypting files that contain sensitive data provides an extra layer of security should unauthorized access occur. Instead of waiting on mandates and regulations (of which there already are many) to demand encryption of sensitive data, organizations can easily implement encryption keys or other data encryption products to protect data at rest and data in flight.

Encrypting at the hardware level is an industry best practice. However, to implement you may be required to upgrade your hardware and take on additional expenses. When you cannot or do not wish to purchase hardware-level encryption packages, consider using the data encryption feature of Adabas. The feature is free of charge.

To encrypt existing Adabas files, you simply unload and decompress the data then run the Adabas Compress utility (ADACMP). During the compression process, the data is encrypted using an encryption code provided by the user. Then you reload the encrypted data.

Encryption is done at the file level and every file should have a different encryption code. Once you have encrypted your data, everything you use (all “input access”) will have to pass the encryption code. All calls to the database, regardless if “access” (read) or “modify” (add/update) require the encryption code to be passed.

The encryption code is not stored, so don’t lose it. However, if you use Adabas SAF Security with RACF, the encryption codes may be stored and passed automatically, based upon appropriate security authentication, when needed. This removes the need to make the codes known to users, and protects the file from corruption that can occur by adding data that is encrypted with the wrong encryption code. Just another great reason to include Adabas SAF Security in your environment!

**Mask nonproduction data**

Data copied from one environment to another, even within the same technology, presents security risks if sensitive fields are not handled appropriately. When copying data from production to nonproduction environments, it is a best practice to conceal private data so that application developers, testers, privileged users and outsourcing vendors are not exposed to such data.

Private data includes:

- Names
- Addresses
- Social security numbers
- Credit card numbers
- Financial data
- Healthcare information
- Employee or consumer information
- Any other type of confidential information applicable to the organization

---

**BEST PRACTICE:**

**Scalable data masking solutions**

Data masking is the method most recommended by industry experts for protecting nonproduction environments. The goal is to have a masking solution that is scalable while concealing the masking logic so that it cannot be used to decode or recreate the original data.
Masking your data is a must in order to protect your customers and company while also complying with data protection legislation, such as HIPAA, PCI DSS, GLBA and EU Data Protection Directive. Data masking is important for your organization because it not only protects private information but gives you the freedom to conduct robust testing with realistic data and outsource testing and development.

**Data Masking for Adabas** is a fast, simple-to-use tool that enables you to de-identify (or mask) data to protect sensitive information while the referential integrity—how the data interrelates with other data and systems—remains intact. This allows testing and development teams to carry out project work against masked “live” data efficiently while ensuring regulatory compliance.

Sensitive data is obfuscated using an extensive range of powerful built-in and user-defined masking functions, including: randomization, substitution, hashing, shuffling, date aging and decryption/encryption. These masking rules are stored in CSV files allowing you to define additional masking rules without programming by simply recording the rules in these easy-to-understand spreadsheets.

Each masking algorithm is fully automated and applied across all data sources, tables and executions within the project to ensure that referential and business integrity is maintained. For example, if a specific address is used across multiple files, the mask of that address will be consistent across all files. The result is that consistent, high-quality test data that reflects reality is provisioned and distributed throughout the project, enabling better and more efficient testing.

If you have a typical enterprise environment, you likely use many types of DBMSs. Software AG’s data masking solution is also available for DB2®, Oracle®, MySQL®, SQL Server®, Sybase®, Ingress®, flat files and mainframe data sources.

**Archive unused data**

Data retention and archival of eligible data for intermediary storage or destruction creates organizational risk if data is not retained long enough, not able to be recovered or is retained too long as government audits require data be made available as mandated by law. Audits also present risk when data is kept too long as it is eligible for discovery and may be required to turn over to regulatory agencies.

Don’t keep inactive data on your production environment. Move it off. **Data Archiving for Adabas** does not deal with copies of data, it really moves data off the production side and places that historical data in a storage “vault.”

If you need to keep data available over a long period of time, even decades, in order to comply with legal regulations and internal audit requirements, archiving is the answer. As an integrated solution in the Adabas family, Data Archiving for Adabas helps you execute your archive policy by executing actions that determine what data to archive, how often and for how long to retain it in the archive.

**BEST PRACTICE:**

Create policy for record retention, archival and destruction

An overall organizational policy for record retention, archival and destruction is needed to prevent inconsistencies in archive practices between siloed data administrators. Policies not uniformly enforced and understood may be perceived as non-compliance during an audit.
Data Archiving for Adabas also takes care of changes to the data structure. Since files never remain static, the solution stores the Adabas FDT (file layout) along with each new subset of the data and archives business data along with its metadata that describes the new file layout. With Data Archiving for Adabas there is no need to create procedures to take care of data structure changes and handle them. It happens automatically.

When historical data is required to be retrieved, Data Archiving for Adabas allows you to search for data in the archive (vault). There is no need to recall a large volume of data to just view a subset of the data. After finding the data required, a recall can be initiated and only the necessary subset of data is restored into Adabas.

When data is archived, it can be deleted from the source database creating efficiencies in storage, improved application performance and greater data security.

**Protect data in flight**

**Authenticate access to queues**

EntireX Security secures distributed application components running with Broker or Broker Services. It will verify who can pull information off from specific queues. EntireX Security covers authentication of user, authorization of client and server, publish and subscribe and Command and Information Services, and encryption of application data.

EntireX Security is installed at specific points where communication between client and server/publish and subscribe application components are protected, using definitions located in the central security repository of your organization (SAF-based RACF, ACF2 or Top Secret) under z/OS, and for UNIX® and Microsoft® Windows® either the local security system of the machine or an LDAP repository. As the trend towards distributed computing increases, both across platforms within a company and between partners, security of data has become ever more critical.

**Secure public networks**

Entire Net-work SAF Security protects data in flight from remote users coming in from the web or a PC. To block unauthorized users from accessing this data on z/OS, Entire Net-work with Entire Net-work SAF Security will automatically detect anyone trying to access the z/OS and require a Mainframe ID and password.

Entire Net-work SAF Security addresses this issue by providing point-of-access verification of incoming requests. It allows Entire Net-Work clients to access SAF data sources. Validation is carried out against the SAF repository, thus maximizing the investment that most z/OS sites have made in their mainframe security repository.

By leveraging SAF repositories, the following features minimize the overhead required for the administration, operation and execution of a mainframe security system:

- Categorized access—For example, all access from mainframe clients can be verified against the same security profile.
- Locality of security checking—Entire Net-work SAF Security can be activated on a link by link basis. For example, an installation may have several Entire Net-Work nodes, of which only one communicates externally. Security checking can be activated for that node alone and only for the external links.

**Encryption for Entire Net-Work** protects your “data in flight” on public networks, including the Internet, between Adabas and any application.

To encrypt data over public networks, you must install Entire Net-Work at both the application and Adabas sides of the communication. When on the mainframe, deploy the Entire Net-Work TCP/IP options and extend the Simple Net-Work Connection driver definitions using settings that enable encryption.

When an application sends a request to Adabas, the Entire Net-Work client checks the target URL and determines whether a secure connection needs to be established. If security is required, the encryption certificate is read and the Net-Work encryption component is called. Once encrypted, the data is sent to the network for transport to Adabas. On the Adabas end, encryption works the same way, just in reverse.

With Encryption for Entire Net-Work, you can secure data sent and received by mobile employees who use the Internet, where data is most exposed to potential theft.
Authenticate users for terminal sessions

Entire Connection handles user authentication to the server with the logon dialogs provided by the host system. A user ID and a password must be provided to open a host session. User IDs and passwords are defined and maintained in the SAF security system used by the host.

To open a terminal session with the host system the user must specify user ID and password. A user must always authenticate to logon to his Windows system and again to open a host session. To implement an SSO with Entire Connection, the host user credentials can be stored encrypted in the share file, then by using the Entire Connection script language, the logon to the host can be done without having to enter user ID and password manually.

Control your application

Manage the application life cycle

There should be very few people who have the ability to move programs into production. Predict Application Control (PAC) is a flexible tool for controlling Natural, COBOL, JCL and Assembler applications throughout the software development life cycle and for ensuring the integrity of applications in the production environment.

PAC allows you to monitor, secure and control an application’s life cycle by defining a series of statuses through which it is promoted. Each status defines a phase in the application’s life cycle and the environment in which the application functions in that phase. These phases typically include development, testing, production and maintenance.

Controls and facilities are built into the migration process to ensure that the application is migrated to the correct location, the set of objects to be migrated is complete and correct, and the migration is made with the approval of the appropriate authority. Each step in the migration process can be control by signing off the promotion to the next status using counter signatures to provide a higher level of security.

When an application is placed into production, it is protected and audited by the utility Predict Application Audit. All activities are documented and reports are available for auditing purposes showing who accessed or changed what and when.

PAC provides an interface to state-of-the-art versioning tools like Subversion (SVN). Using this interface, Predict Application Control can take program source changes and promote it to the various life-cycle phases.

SSL/SSH Support

Entire Connection supports SSL for IBM host connections and SSH for Linux/UNIX connections. Supported modes are user-id/password and trusted user certificate which allows automated logon.
Secure Natural applications

Natural Security is a comprehensive system to control and check access to a Natural environment. Natural Security enables you to protect your Natural environment against unauthorized access and improper use. You may define exactly who will be allowed to do what. You may restrict the use of whole libraries and Natural utilities, as well as individual programs, functions and DDMs. You may further define the conditions and times of use.

By defining objects and the relationships between these objects, you in essence are providing a custom-made Natural environment for each individual user. Objects are defined to Natural Security by creating a security profile for each object. The main types of objects supported by Natural Security are users, libraries, DDMs/files and the Natural utilities.

In addition, access to Natural can be made environment-specific. A Natural environment is determined by the combination of the system files FNAT, FUSER, FDIC and FSEC. Natural environments can be defined in the external security system. By defining environments and controlling their accessibility, it is possible, for example, to fully separate the protection of a Natural development environment from that of a Natural production environment. At the same time, this avoids system-file mix-ups (for example, a test-environment FSEC file in conjunction with a production-environment FUSER file).

Nearly all Adabas & Natural sites also utilize Natural Security with the AUTO=ON setting so that access to Natural utilizes RACF, ACF2 or Top Secret authentication and further limits user access to appropriate libraries within Natural and limits access to Software AG product functions found in Adabas Online Services, SYSMAIN, Predict and Natural utilities.

Natural SAF Security allows you to control user access to Natural based on user and resource definitions kept in a central SAF-compliant security system (e.g., RACF, ACF2 and Top Secret). When you use Natural SAF Security, you only need to define users in the external security system.

However, using SAF-compliant security systems in conjunction Natural Security can provide additional benefits. Only user groups are defined in Natural Security while individual access is controlled centrally in the SAF-compliant repository. When Natural SAF Security is active and a user logs on to Natural, the user authorization checks are executing using the user ID and user password from the external security system. After the authorization, further security checks—particularly concerning the use of Natural libraries and utilities—will be based on the user group definitions in Natural Security.

Although library protection via an external security system is possible, the Natural Security library security profiles provide more sophisticated mechanisms for protecting Natural libraries.

Secure SQL access

Adabas SQL Gateway provides real-time access to Adabas data by creating a virtual data services layer that abstracts data from disparate data sources into a single layer that is commonly understood and accessible. You gain the power to grant users immediate access to Adabas data, no matter where that data resides.

Users that come through SQL Gateway have full transaction capabilities (create, read, update and delete). Luckily, security established at the database or operating system is not bypassed when using Adabas SQL Gateway to provide access to standard SQL applications, such as Business Objects, IBM® Cognos®, CONNX InfoNaut®, Crystal Reports, Access®, Excel® and Powerbuilder®. For example, if a user has been given read-only access to a table by Adabas SAF Security or ADASCR, the user continues to have read-only access to the table through Adabas SQL Gateway.

Adabas SQL Gateway provides additional security management. View, table and column security levels can be assigned to an individual user, or to a group of users and stored in Adabas SQL Gateway (CONNX) data dictionary. The data dictionary is encrypted to only allow access to authorized users.

To simplify administration and access, integrated security helps consolidate and manage passwords to multiple databases. Users may specify a single user name and password to
access multiple databases that each requires a distinct user name and password. When a target password expires or changes, Adabas SQL Gateway will prompt the user for a new password.

Additional licenses are available for SQL Gateway giving users secure real-time, read/write SQL access to multiple data sources whether they are non-relational, relational or both. More than 40 targets are supported including IBM VSAM, IBM IMS, IBM C-ISAM, Micro Focus®, Oracle, IBM® DB2®, Sybase and MySQL. A wide variety of platforms are supported, including the mainframe, Solaris®, HP-UX®, AIX and Linux®.

**Secure direct call applications**

While Natural Security will protect your Natural application, it does not apply to other direct call applications. Applications primarily written in COBOL, Assembler, FORTRAN and other common third-generation languages (3GLs) use direct calls to interface with Adabas. These “direct calls” are a proprietary interface to Adabas. They inherently have no security associated with them.

Since the applications using direct calls have full transactional capabilities to Adabas, your data is at risk. You must implement database security such as Adabas SAF Security or ADASCR to protect your data from applications that use direct calls. In fact, if Adabas is properly secured, you don’t need to worry about data security risks no matter what the application is written in.

---

**Conclusion**

Security is effective only if governed by an enterprise-wide data security policy where all disciplines are trained and have a common understanding of it. Managing the security rules for access, authentication and encryption in a central, industry-standard security package (SAF or LDAP-compliant) will go a long way in helping execute an organization-wide policy consistently and efficiently.

It is also important to share best practices across the organization’s technical areas so that they can be adopted and followed. The evolutionary nature of enterprise systems and infrastructure no longer permits a static security policy. The key to success is to facilitate cross-organizational discussions of security issues and policy development in order to create and maintain a comprehensive enterprise-wide data security policy that addresses the multiple levels of security risks.

Learn how to best secure your Adabas & Natural environment. Contact your local Software AG representative for more details and ask about getting a security assessment from our consulting team.
Secure Your Adabas & Natural Environment

About the authors

Brian Johnson is employed by Eaton as an IT specialist since 1999. Brian has nearly 25 years of experience with Software AG products, predominantly as an Adabas DBA and product administrator, and previously as a Natural programmer. He served as president and North American area representative of the Software AG Group Executive Committee from 1998 through 2012. He is now the liaison for the Software AG Group Presidents Committee. He currently resides and works in Perth, Ontario, Canada with his wife, Carolyn, and four shelties.

Becky Albin is chief architect for Software AG, and is a leading authority on all that is Adabas. During her career of more than 25 years with Software AG, Becky has influenced product development, marketing direction and partnership pursuits. Her insights into real-world data management challenges and their solutions are based on her daily experience working directly with customers, Software AG consultants and developers to fine tune their databases and data strategies for optimal performance.

Prior to Software AG, Becky was with a Federal security agency for nine years. Becky continues to travel the world, helping customers develop and refine their data management strategy. Her proactive problem solving continues to guide development of new features for the Adabas family of products.

Additional contributions provided by:

Bruce Beaman is senior director of Adabas & Natural product marketing worldwide. Bruce has been with Software AG for 26 years and most of that time he’s worked with Adabas & Natural. Many of our long-time customers know Bruce from the seven years he spent in Software AG training as an Adabas instructor. After his stint in education, Bruce was part of the Technology Marketing Group for Software AG Americas, with other senior level product specialists. In 2000, Bruce joined Software AG product marketing and has spent the last 12 years planning and positioning Adabas & Natural and the related family of products for continued growth.

Karlheinz Kronauer is the senior director of product management for Natural and Natural add-on products at Software AG. He has more than 30 years of experience working in the IT industry, specializing in the areas of product development, product marketing and product management. Karlheinz recently led the successful product management of NaturalONE, the new Eclipse™-based integrated development environment for Natural.

Wolfgang Weiss is a director of product management on the Adabas & Natural team at Software AG. He is responsible for the Adabas Family of products. Wolfgang has more than 20 years of experience at Software AG.