CCN-CERT

BP/25

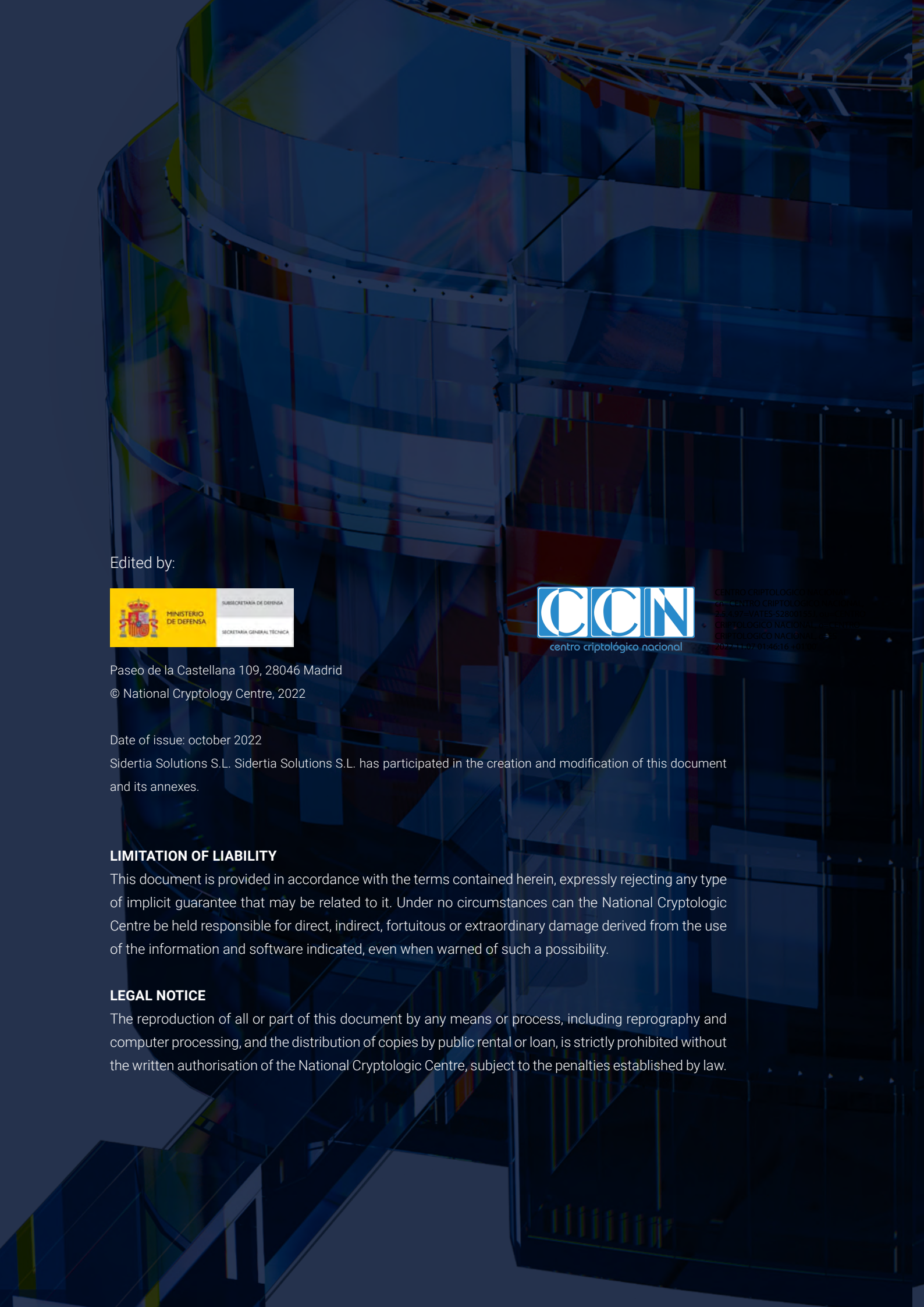
Security Recommendations for Db2 Databases over zOS

GOOD PRACTICE REPORT

FEBRUARY **2022**

ccn-cert centro criptológico nacional

CCN centro criptológico nacional



Edited by:

MINISTERIO DE DEFENSA
SUBSECRETARÍA DE DEFENSA
SECRETARÍA GENERAL TÉCNICA

CCN
centro criptológico nacional

Paseo de la Castellana 109, 28046 Madrid

Date of issue: october 2022
Sidertia Solutions S.L. Sidertia Solutions S.L. has participated in the creation and modification of this document and its annexes.

Index

1. About CCN-CERT, National Governmental CERT 4

2. Introduction 5

3. Best Practices 7

4. Software review 8

5. Data access control 10

5.1 Permissions by user ID 11

5.2 Permissions by role 12

5.3 Access based on ownership 13

5.4 Multi-level Access 14

5.5 External acces 14

5.6 User level access 15

6. Access to the Db2 Subsystem 16

6.1 Access control with RACF 16

6.2 Access control with IMS Terminal Security 17

6.3 Access control with CICS Transaction Code Security 17

6.4 Local and remote calls 18

6.4.1 Calls from local systems 18

6.4.2 Remote system calls 19

7. Audit 20

8. Protection of communications 23

9. Encryption 24

10. Backup policies 26

10.1 Backup, recovery and reboot 27

11. Glossary **30**

1. About CCN-CERT, National Governmental Cert

The CCN-CERT is the Computer Emergency Response Team of the National Cryptologic Centre, CCN, attached to the National Intelligence Centre, CNI. This service was created in 2006 as the Spanish National Governmental CERT and its functions are set out in Law 11/2002 regulating the CNI, RD 421/2004 regulating the CCN and in RD 3/2010, of 8 January, regulating the National Security Framework (ENS), modified by RD 951/2015 of 23 October.

Its mission, therefore, is to contribute to the improvement of Spanish cybersecurity, being the national alert and response centre that cooperates and helps to respond quickly and efficiently to cyber-attacks and to actively face cyber-threats, including the coordination at state public level of the different existing Incident Response Capabilities or Cybersecurity Operations Centres.

All of this, with the ultimate aim of achieving a safer and more reliable cyberspace, preserving classified information (as set out in art. 4. F of Law 11/2002) and sensitive information, defending Spain's Technological Heritage, training expert personnel, applying security policies and procedures and using and developing the most appropriate technologies for this purpose.

In accordance with these regulations and Law 40/2015 on the Public Sector Legal Regime, the CCN-CERT is responsible for the management of cyber-incidents affecting any public body or company. In the case of critical public sector operators, cyber-incident management will be carried out by the CCN-CERT in coordination with the CNPIC.

The CCN-CERT is the Information Security Incident Response Team of the National Cryptologic Centre.

2. Introduction

Databases have become the fundamental engine of business, storing sensitive data and all the data needed to manage organisations, from customer names to product prices, to temporary data stored for modifications to critical applications in the environment.

As it is an access point for all kinds of cyber-attacks, there should be clear guidelines to follow to avoid exposure to potential vulnerabilities.

The need for this document stems from the point where best practices and guidelines should be imposed on the implementation, modification and maintenance of the database.

The following document provides best practices for the secure use of IBM Db2 version 12 on Z systems.

Over the years, Db2 has recognised and addressed the following security issues:

- **Theft of privileges or mismanagement**
- **Handling of applications or application servers**
- **Manipulation of data or records**
- **Theft of storage media**
- **Unauthorised access to objects.**

It should be noted that the keys to this task will be based on the following **key points:**

- **Authentication**
- **Authorisation**
- **Data integrity**
- **Confidentiality**
- **System integrity**
- **Audit**

But it is not only the database engine that must be taken into account, but also the environment where it is deployed. This point will be really important for the security and management of the databases.

In this document, as already mentioned, the focus will be on the z/OS operating system, with all that can be involved in deploying a database on one of the most secure systems on the market.

3. Best practices

Regardless of the database that is deployed on the z/OS operating system, there are always security guidelines that must be taken into account.

The engine where the database is deployed provides the building blocks for communications and access to the database, hence the importance of updating and maintaining the operating system itself.

Once the operating system is installed, the following recommended steps should be followed for a best practice:

1. Updating the operating system where the database is deployed.
2. Application of recommended patches and updates to address vulnerabilities.
3. Subscription to security news about the z/OS operating system and the Db2 database.
4. The use of authorisations with user and password authentication.
5. Specific security criteria recommended by the operating system vendors themselves.

The engine where the database is deployed provides the building blocks for communications and access to the database, hence the importance of updating and maintaining the operating system itself.

4. Software review

Once the product has been installed or patched, it must be identified that everything has been done correctly and that the security level is as high as possible, taking into account the manufacturer's recommendations for their products.

It is important that when any kind of update is made, the documentation provided is read carefully so that you know what is involved and what to expect.

At the software level, the following actions should be carried out:

- **Maintenance of the product on the latest version.**
- **Product maintenance with the Db2 intrinsic FL upload and the APARs necessary for the correct functioning of Db2 and its utilities.**
- **Updating of security and evolutionary patches of the product.**
- **Check, from time to time, the accounts that have root privileges, to see if they are the appropriate ones and to check if they are being reused, cloned, or used inappropriately.**

It is important that when any kind of update is made, the documentation provided is read carefully so that you know what is involved and what to expect.

- **Check for possible vulnerabilities in both the software and the operating system where it has been deployed.**
- **If it is considered that a vulnerability has been found, the manufacturer should be notified as soon as possible with a detailed description of the problem and the situations where vulnerabilities have been found.**
- **If the manufacturer uploads a new patch for a vulnerability and the people in charge of updating the patches have not done the update, you should contact the systems team to get the patch done as soon as possible.**
- **Clean up temporary files.**
- **Maintain up-to-date systems that talk to the database.**
- **Maintain the database engine up to date.**
- **Keep the channels through which the database is accessed up to date and secure.**

5. Data access control

Access to the data can be given by users who want to call specific information, such as processes in the environment itself. That is, from interactive terminals to local or remote Store Procedures, utilities or CICS or IMS transactions. It could also come from applications running in batch, to applications using DDF or CLI or connections via JDBC.

For all this to work properly, it is recommended to use different users and roles that can access the data with different security privileges, these should be given according to the access needed by each user or programme with a previous study of each use case.

Authorisations must be given for each view, table, schema, object, etc. To do so, it will be necessary to:

- Define authorisation rules for Db2 systems.
- Define security levels within Db2 objects.
- Define user profiles and roles that can access the data and how to access them.

- It must be possible to audit data entries to check that appropriate security rules have been created.
- The system must always be able to know the origin of the request or "who" is making the request.
- The creation, modification and deletion of accounts, identifiers, roles or groups must be centralised to a single role of the database administrator. In addition, he/she shall be responsible for accrediting the different security levels as agreed with the security system roles.

5.1 Permissions by user ID

One of the ways to scan Db2 entries is through the main or primary user identifier, giving privileges from Db2 to those users.

It is recommended to use a unique primary identifier for each system/person who wants to access the data, and a secondary identifier that will be associated to the primary one and to which different security layers will be associated. This is one of the ways of being able to identify the access permissions of each user on a massive scale. Granting an ID the privilege to execute a plan or package can provide a finely detailed set of privileges and can eliminate the need to grant other privileges separately.

It is recommended to use a unique primary identifier for each system/person who wants to access the data, and a secondary identifier that will be associated to the primary one and to which different security layers will be associated.

5.2 Permissions by role

In addition, it is recommended to create user roles where privileges can be guaranteed at this level. This grouping created to categorise users or identifiers will help in the creation of security profiles by job type grouping within the database.

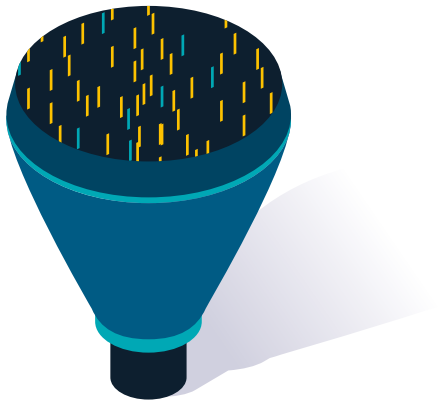
Among the roles that can be found, there are some basic and common roles that should be taken into account and created accordingly (these are not the only roles that can be used, but they are the minimum recommended).

- **Security Administrator Role:**
  These are the security managers, in addition to user accounts, audits to be done or the management of encryption keys.

- **Database Administrator Role:**
  Involves a wide range of tasks, from database generation, updating the software, checking performance, starting and shutting down the database, and even creating the necessary database backups. These users should have database administration and modification privileges.

  It is a role that should only be given to administrators, and that once the database is created, access should be revoked for all but the actual administrators who keep the role.

  This role should be reviewed periodically to check that the identifiers associated with this role are correct.



- **Data mining user role:**
  This role is suitable for those users who are in charge of doing business data mining and exploitation. These users usually have read-only access.

- **Application administrator role:**
  These can be accounts that only have the privileges to update and patch the application.

- **Application role:**
  This category could encompass Identifier accounts that come from external or internal applications that need to view data. In this case, they usually have read and write access to the tables or data necessary for their application.

- **Role of regular database users:**
  This includes identifiers with restricted access to their specific data, tables, views and objects.

5.3 Access based on ownershipd

Object ownership carries with it a set of related privileges on the object. Db2 provides separate controls for object creation and ownership.

If you want to prevent users from getting implicit privileges from object ownership, you can make a Db2 role the owner of the object. To do this, you must create the object in a trusted context that is defined with the ROLE AS OBJECT OWNER AND QUALIFIER clause.

5.4 Multilevel access

Also known as multi-layered access, it allows users and roles to be classified with layers of security. These layers are based on a hierarchy of security levels rather than security categories.

This way of accessing is further enhanced by the use of z/OS's own multi-level security functionality, preventing unauthorised users from reaching classifications where they should not have access.

If used at the row level, you can define very strong security policies for Db2 objects and perform security checks at the row level. These checks help to visualise which users are allowed to view, modify or perform any other action on rows of data.

The use of this multi-level access is recommended for highly sensitive data.

5.5 External access

You can control access to Db2 using an output routine provided by Db2 or an output routine that you write.

If your installation uses one of the access control authorisation exit routines, you can use it to control authorisation checking and authentication, instead of using other techniques and methods.

5.6 User level access

When you create a user who must log in with a password, the database administrator will send you a temporary password that you must change the first time you log in to the system. The password change must come with the following guidelines:

- The password must not contain words such as: USER, ADMINS, PUBLIC, GUESTS, any SQL reserved words.
- If you use TSO, RACF, or any of the other security applications that can be connected, you will need to follow the guidelines set by each of these security applications, as passwords will need to follow the guidelines, such as length, of each of these systems.
- For a password to be reliable and secure it is recommended that:

1.	At least 12 characters and no more than 15 characters.
2.	Alternate between uppercase and lowercase letters.
3.	Containing at least one number.
4.	Containing at least one special character (! # # $ $ % & ‘ (( ) * + , - . / : ; < = > ? @ ^ _).
5.	That does not contain the user’s name or any other name of any role.
6.	Not containing dates of birth, in general, dates.
7.	There must be a limit to the number of attempts to log in with a wrong password.
8.	It must be possible to lock a user after a number of unsuccessful attempts (the limit of attempts should not be more than six).
9.	There should be a time limit for external sessions that expire and the system can be shut down. (The recommended time limit is 90 minutes, but the case of each of the applications or users performing this session should be studied).
10.	Passwords should be renewed at least every three months. It is recommended to create reminders at user level.

6. Access to the Db2 subsystem

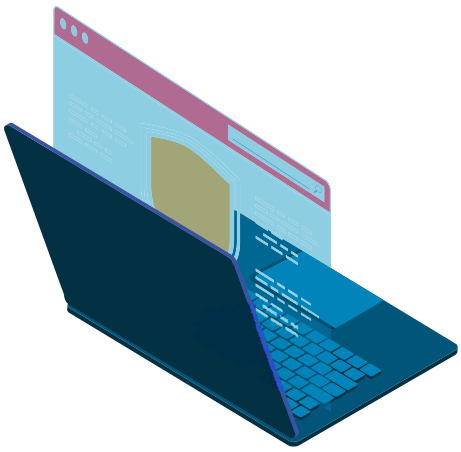
Access to an external Db2 subsystem can be controlled with products such as RACF.

6.1 Access control with RACF

The advantages of controlling the input to the subsystems with RACF will be shown below:

- Identify and verify the identifier associated with the process attempting to enter.
- Connect identifiers to roles and groups registered in RACF. The security database that is applied on z/OS.
- Audit different attempts to access protected resources.

It is recommended to use database access through the RACF as long as the system is operational in the same subsystem where the database is located.



6.2 Access control with IMS Terminal Security

IMS Terminal Security allows you to control and limit the entries of a transaction code to an LTERM or groups of LTERMS in the system. To protect a particular programme, you must authorise a transaction code to be entered in the list of LTERMS. Alternatively, you can associate each LTERM with a list of transaction codes that a user can enter from that LTERM.

This code will be the one that is passed to Db2 as identifier and is registered.

The use of this product is recommended, as long as it is installed on an engine operating system.

6.3 Access control with CICS Transaction Code Security

CICS Transaction Code Security works with RACF to control which transactions and programs can access Db2, the option can be enabled or disabled in bind operations to limit access to specific CICS subsystems.

The use of this product is recommended whenever CICS transactions are used for database calls and it is implemented within the system.

6.4 Local and remote calls

6.4.1 Calls from local systems

The TSO logon could be used to log into internal systems.

If you are running Db2 with TSO and you are using the TSO ID logon as the primary Db2 ID, it is TSO itself that checks whether the ID has access or not.

The use of TSO logon is recommended for all users who have access through this system. It is also recommended to assign the TSO user as the primary ID and continue to maintain a secondary ID for the application of security layers.

After performing these actions, the authorisation ID can again use the services of an external security system.

NOTE:

Passwords used by these identifiers should follow the same advice as a TSO user.

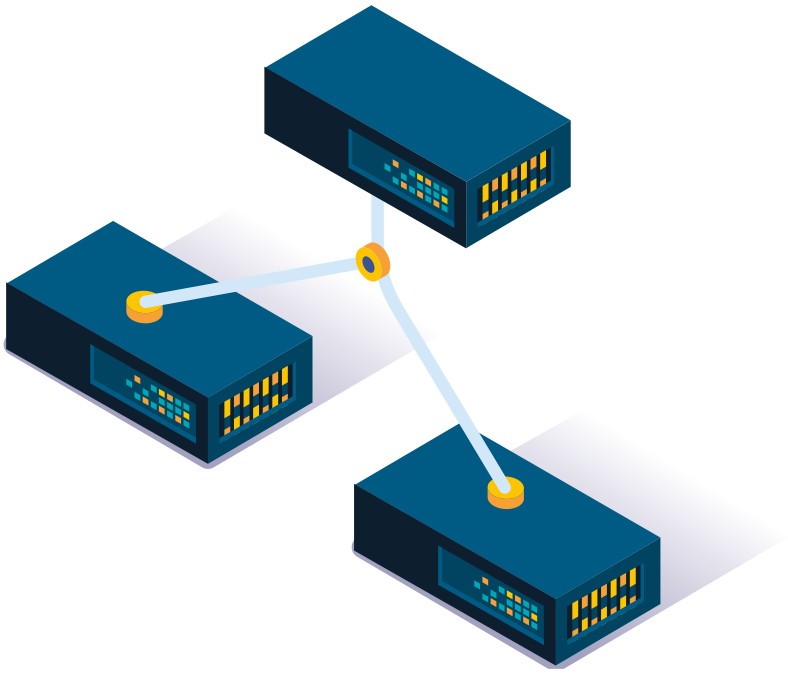
6.4.2 Remote system calls

The use of security systems that can manage, maintain and audit such entries is recommended.

If a security system such as RACF is deployed on the engine, this system should be used for external access since, through its functionalities, it is able to validate several security checks before being able to enter the system.

The following tasks are recommended for an engine deployed on a RACF safety system:

- Verify an ID associated with a remote request and check the ID with a password.
- Generate a "PassTicket" on the requesting side. This avoids sending passwords over the network.
- Verify a Kerberos ticket if your distributed environment uses Kerberos to manage user access and perform user authentication.
- Authenticate entries with Db2 Communication Database (CDB). This is nothing more than tables in the Db2 catalogue that are used to establish conversations with remote systems.



7. Audit

Auditing access and permissions will be a key element of a good security configuration within the Db2 12 implementation on z/OS.

It will attempt to monitor any input from both users and applications and differentiate between them so that the reviewers of this audit will be able to distinguish between common threats or common inputs.

Monitoring attempts to answer questions such as: What sensitive data requires authorisation, who has authorisation to access the data, who has accessed the data, who is attempting to gain privileges to access the data and what attempts are being made to gain unauthorised access.

The Db2 catalogue contains critical authentication and authorisation information. This information provides the main audit clue for the Db2 subsystem. You can retrieve the information from the catalogue tables by issuing SQL queries.

Most of the catalogue tables describe Db2 objects, such as tables, views, tablespaces, packages and plans. Other tables, particularly those with the string "AUTH" in their names, contain records of all privileges and authorisations granted. Each catalogue record of a grant contains the following information:

Auditing access and permissions will be a key element of a good security configuration within the Db2 12 implementation on z/OS.

☑	☑	☑	☑	☑
Name of the object	**Type of privilege**	**IDs receiving the privilege**	**IDs granting the privilege**	**Time of award**

7. Audit

The Db2 audit trail can help you monitor and track all access to your protected data. Audit trail logs provide another important trace for the Db2 subsystem. You can use the audit trail to record the following access information:

- Changes to authorisation IDs
- Changes to the data structure, such as deleting a table
- Changes to data values, such as updating or inserting records
- Attempted access by unauthorised IDs
- Results of GRANT statements and REVOKE statements
- Assigning Kerberos ecurity tickets to IDs
- Other activities of interest to auditors

It should be possible to audit at instance level as well as at database level, the recommendations to be followed are as follows:

- Enable Db2 trace so that you can write to the logs.
- Before activating it, it will not save old data. It will be necessary to take into account the cleaning of the logs from time to time.
- Choose the elements to be audited, it is recommended to activate event categories such as login, modifications, deletions, etc. According to your needs, but you will have to activate the events to be able to see all the information in the traces.
- Audit trace uses the primary ID to keep track of modifications and entries that are made, so it is recommended that the primary ID be understandable and visible to know who the person is entering (and use the secondary ID to layer security according to privilege).

- Generate daily and weekly reports on acquired traces in which the following elements can be defined:
  - **Consumption of sensitive data.**
  - **Higher privileges to different IDs**
    (It is recommended to monitor IDs with special authorities and to carefully control IDs with privileges on confidential data You can consult the Db2 catalogue to determine which IDs have privileges and authorisations at any given time).
  - **Failed logins and number of attempts**
    (If you have sensitive data, always use audit trail class 1).
- It is recommended to create an auditor role who is the person who has access to this data and can review it.
- It is recommended that the reports generated cannot be modified by the rest of the users, not even by the auditor. Furthermore, they should not be deleted without a special operation.
- It is recommended to audit all actions of the database administrator (in charge of giving or removing privileges to the other roles).
- It is recommended that access to sensitive data be specifically audited.
- It is recommended to use some type of system that is capable of alerting if necessary. A SIEM-type system that can generate security alarms.

It is recommended that the reports generated cannot be modified by the rest of the users, not even by the auditor. Furthermore, they should not be deleted without a special operation.

8. Protection of communications

In order to further protect database accesses, it is recommended to take action on the communication channels to the database.

- It is recommended to use certificates issued by a trusted certification authority and to use encryption algorithms endorsed by the National Cryptologic Centre.
- It is recommended to make use of vulnerability management tools, where regular scans for threats are planned.
- The Db2 deployment on z/OS supports TLS 1.0, SSL 3.0 and SSL 2.0 protocols.

9. Encryption

La recomendación contempla que se cifren los datos más sensibles de base de datos, para ello se pueden seguir las recomendaciones aportadas por el Centro Criptográfico Nacional (CCN) en sus documentos de referencia sobre Db2.

Db2 ya viene preparado para encriptar de manera transparente datos en reposo, como pueden ser los logs, catálogos, directorios, tablas e índices. La recomendación es utilizar las propias funcionalidades de Db2 para poder proteger esos datos en reposo.

Si se utiliza DFSMS, la recomendación es ampliar sus funciones para que puedan encriptar los datos dentro de Db2. De esta manera se puede optimizar la encriptación haciendo uso del hardware intrínseco del Z. (Disponible en zOS 2.2, RACF o ICIS u otro producto de seguridad).

Antes de poder utilizar el cifrado de conjuntos de datos DFSMS de z/OS para cifrar conjuntos de datos de Db2, asegúrese de que su sistema cumpla con los siguientes requisitos:

- The operating system is z/OS 2.2 or later. For z/OS 2.2, the PTFs for APAR OA50569 and APAR OA53951 must be applied.
- The necessary hardware is installed.
- ICSF and RACF or equivalent security products.
- The user ID of the initiated Db2 task and any user ID that is required to read or write to an encrypted dataset has permission to use any key tags that are used to protect Db2 datasets.
- Any key tag used to protect Db2 datasets is defined on all members of a data exchange group and on any backup system that can read or write from an encrypted dataset.

La recomendación contempla que se cifren los datos más sensibles de base de datos, para ello se pueden seguir las recomendaciones aportadas por el Centro Criptográfico Nacional (CCN) en sus documentos de referencia sobre Db2.

- Any user ID required to run any of the stand-alone utilities is authorised to use any key tags that are used to protect the Db2 datasets.

- Prerequisite updates to make your security product support z/OS dataset encryption.

In addition, and as part of the z/OS operating system, data can be secured with the use of RACF discussed earlier in this document.

If you want to transport data from one system to another, copying, creating new databases, or sharing it with other systems, the following must be taken into account:

- If the data is sensitive and there is a need to share it, a secure channel using data-in-transit encryption must be created, and the necessary authorisations must be in place to securely handle the data at the destination.

- If the data is non-sensitive, it is nevertheless recommended to create a secure channel for sharing the data and, where possible, to have encryption keys for the data in transit.

- For a disaster recovery situation, if the data is to be accessed at another physical site, then the ICSF keys and RACF profiles must be configured similarly to the source site. The same rule applies to Db2 proxy and source sites in the GDPS® zero data loss continuous availability solution environment.

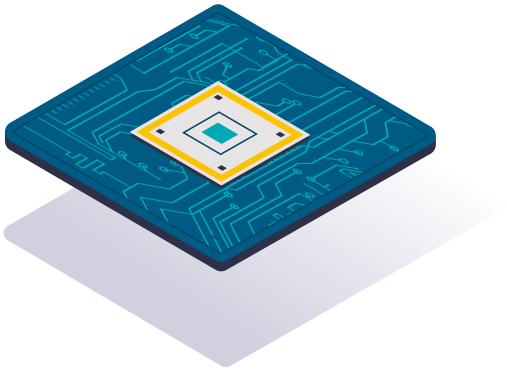
In addition, encryption is recommended:

- **The data used for the backup.**
- **Archive images.**
- **Data protected by any of the data protection laws.**
- **Logs involving the modification or regeneration of commands.**

10. Backup policies

Some general best practices are outlined regardless of product or version:

- Restoration of any backup requires controlled access to the encryption key and must be audited, both access and restoration.
- It is recommended that backups be made on a regular basis, at least one incremental copy per day, and should be kept for at least seven days. It is recommended that an incremental copy be created on a weekly basis and kept for twelve months. And an annual one to be kept for five years.
- It is recommended that the storage of these copies should not be in the same physical location as the main system.
- Periodic restoration tests (at least twice a year) are highly recommended to check that the restoration process is working properly.
- Data maintenance and consistency is very important, it is recommended to use Db2's referential integrity to check that the consistency of the data, both in the backups and in transit, is reliable and correct.



10.1. Backup, recovery and restart

Although high data availability is a goal for all Db2 subsystems, unplanned outages are difficult to avoid completely. However, a good backup, recovery and restart strategy can reduce the elapsed time of an unplanned outage.

To reduce the likelihood and duration of unplanned outages, you should regularly back up and reorganise your data to maximise data availability for users and programmes.

Many factors affect the availability of databases. Here are some key points to consider:

- You must understand the options of utilities such as COPY and REORG.
  - You can retrieve online structures such as tablespaces, partitions, datasets, a range of pages, a single page and indexes.
  - You can retrieve tablespaces and indexes at the same time to reduce retrieval time.
  - With some options in the COPY utility, you can read and update a tablespace while copying it.

- I/O errors have the following effects:
  - I/O errors in one data range do not affect the availability for the rest of the data.
  - If an I/O error occurs when Db2 is writing to the register, Db2 continues to operate.
  - If there is an I/O error in the active record, Db2 moves to the next data set. If the error is in the archive record, Db2 dynamically allocates another data set.

- Documented disaster recovery methods are crucial in the event of disasters that may cause a complete shutdown of your local Db2 subsystem.

- If Db2 is forced into a single mode of operations for the boot data set or logs, you can usually restore the dual operation while Db2 is still running. Db2 provides extensive methods for recovering data after errors, failures, or even disasters. It can recover data to its current state or to a previous state. The data units that can be recovered are table spaces, indexes, index spaces, partitions, and data sets. You can also use the recovery functions to back up an entire Db2 subsystem or a data exchange group.

- The development of backup and recovery procedures is essential to avoid costly and time-consuming data loss. In general, ensure that the following procedures are implemented:
  - It creates a point of consistency.
  - Restore the system and data objects to a point of consistency.
  - Back up and recover the Db2 catalogue and its data.
  - Recovering from out-of-space conditions.
  - Recovering from a hardware or power failure.
  - Recover from a z/OS component error.

In addition, your site should have a recovery procedure at a remote site in case of disaster.

Specific problems requiring recovery can range from unexpected user error to failure of an entire subsystem. A problem may occur with hardware or software; the damage may be physical or logical. Here are some examples:

- If a system failure occurs, a restart of Db2 restores data integrity. For example, a Db2 subsystem or an attached subsystem may fail. In either case, Db2 automatically restarts, reverts the uncommitted changes, and completes processing of the committed changes.

You can also use the recovery functions to back up an entire Db2 subsystem or a data exchange group.

- If a media failure occurs (such as physical damage to a data storage device), you can recover the data to the current point.
- If the data is logically damaged, the goal is to recover the data to a point in time before the logical damage occurred. For example, if Db2 cannot write a page to disk due to a connectivity problem, the page has a logical error.
- If an application program terminates abnormally, you can use utilities, logs and image copies to recover data to an earlier point in time.

Recovery of Db2 objects requires adequate image copies and reliable log data sets. You can use a number of utilities and some system structures for backup and recovery. For example, the REPORT utility can provide some of the information needed during recovery. You can also obtain log data set inventory information from the boot data set (BSDS).

11. Glossary

TLS: Transport Layer Security is a communications protocol whose main purpose is to provide privacy and data integrity between two communicating applications. The protocol is composed of two layers: the TLS registration protocol and the TLS handshake protocol. During TLS negotiation, a public key algorithm is used to securely exchange digital signatures and encryption keys between a client and a server. The identity information and the key are used to establish a secure connection for the session between the client and the server. Once the secure session is established, the data transmission between the client and the server is encrypted using a symmetric algorithm, such as AES.

RCAC: Row and Column Access Control. It allows access to a table to be controlled at row level, column level or both and can be used to complement the table privilege model, ensuring that information is adequately protected and that users only have access to the subset of data that is required to perform their job tasks and comply with specific rules and regulations.

LBAC: Label-Based Access Control. It is a security model that is primarily intended for government applications or applications with known classification grades, as it requires data and users to be classified with a fixed set of rules that are implemented.

DBA: Database Administrator.

FIPS: Federal Information Processing Standards (FIPS) Publication 140-2 is a U.S. government standard that defines the minimum-security requirements for cryptographic modules in information technology products, as defined in Section 5131 of the Information Technology Management Reform Act of 1996.

LDAP: Lightweight Directory Access Protocol refers to an application-level protocol, which allows access to an ordered and distributed directory service to search for information in a network environment.

SSL: Secure Sockets Layer, the standard technology for keeping an Internet connection secure, as well as for protecting any sensitive information sent between two systems and preventing criminals from reading and modifying any data being transferred, including information that could be considered personal.

Kerberos: A computer network authentication protocol created by MIT that allows two computers on an insecure network to prove their identity to each other in a secure manner.

OTP: One-time password used for authentication.

Social Sign-In Authentication: Social Sign-In is a single sign-on for end users. With existing login information from a social media provider such as Facebook, Twitter or Google, the user can log in to a third-party website instead of creating a new account specifically for that website.

AES: Advanced Encryption Standard (AES), is a block cipher scheme adopted as an encryption standard by the United States government, created in Belgium. AES was announced by the National Institute of Standards and Technology (NIST) as US FIPS PUB 197 (FIPS 197) on 26 November 2001 after a 5-year standardisation process. It became an effective standard on 26 May 2002. Since 2006, AES is one of the most popular algorithms used in symmetric cryptography.

DES: Data Encryption Standard (DES) is an encryption algorithm, i.e., a method for encrypting information, chosen as a FIPS standard in the United States in 1976, and whose use has spread widely around the world.

Triple DES: In cryptography, Triple DES is the name given to the algorithm that does triple DES encryption.

SHA: Secure Hash Algorithms are a family of cryptographic hash functions published by the National Institute of Standards and Technology (NIST) as a US Federal Information Processing Standard (FIPS).

SIEM: Security Information and Event Management (SIEM) is a cyber security term where services and software products combine two systems: Security Information Management (SIM) and Security Event Management (SEM).

DDoS: In computer security, a distributed denial-of-service attack, also called a DoS (Denial of Service) attack, is an attack on a computer system or network that causes a service or resource to be inaccessible to legitimate users.

CVE: Common Vulnerabilities and Exposures (CVE) is a list of recorded information on known security vulnerabilities, in which each reference has a CVE-ID number, description of the vulnerability, which versions of the software are affected, possible workaround (if any) or how to configure to mitigate the vulnerability and references to publications or forum or blog posts where the vulnerability has been made public or its exploitation is demonstrated. In addition, a direct link to information from the NIST Vulnerability Database (NVD), where more details of the vulnerability and its assessment can be obtained, is usually also displayed.

NIST: The National Institute of Standards and Technology (NIST), called the National Bureau of Standards (NBS) between 1901 and 1988, is an agency of the Technology Administration of the U.S. Department of Commerce. The mission of this institute is to promote innovation and industrial competition in the United States through advances in metrology, standards and technology in ways that enhance economic stability.

RLS: Row-Level Security. Row-level security allows you to use group membership or execution context to control access to rows in a database table.

Db2: Represents the Db2 licensed program or a particular Db2 subsystem. IBM renamed DB2 to Db2, and Db2 for z/OS is the new name for the offering formerly known as 'DB2 for z/OS'.

RACF: Represents the functions provided by the RACF component of z/OS Security Server.


CCN
centro criptológico nacional
CCN-CERT
centro criptológico nacional


cn-cert
centro criptológico nacional
CCN
centro criptológico nacional