Informatica

Informatica® Dynamic Data Masking
9.8.2

Dynamic Data Masking Accelerator for use with SAP

Informatica Dynamic Data Masking Dynamic Data Masking Accelerator for use with SAP
9.8.2
October 2016

Publication Date: 2018-06-25

Table of Contents

Preface

Dynamic Data Masking Accelerator for use with SAP contains information to help administrators use the accelerator to implement Dynamic Data Masking with an SAP application. This guide assumes that you have knowledge of Dynamic Data Masking.

Informatica Resources

Informatica Network

Informatica Network hosts Informatica Global Customer Support, the Informatica Knowledge Base, and other product resources. To access Informatica Network, visit https://network.informatica.com.

As a member, you can:

- Access all of your Informatica resources in one place.
- Search the Knowledge Base for product resources, including documentation, FAQs, and best practices.
- View product availability information.
- Review your support cases.
- Find your local Informatica User Group Network and collaborate with your peers.

Informatica Knowledge Base

Use the Informatica Knowledge Base to search Informatica Network for product resources such as documentation, how-to articles, best practices, and PAMs.

To access the Knowledge Base, visit https://kb.informatica.com. If you have questions, comments, or ideas about the Knowledge Base, contact the Informatica Knowledge Base team at KB_Feedback@informatica.com.

Informatica Documentation

To get the latest documentation for your product, browse the Informatica Knowledge Base at https://kb.informatica.com/_layouts/ProductDocumentation/Page/ProductDocumentSearch.aspx.

If you have questions, comments, or ideas about this documentation, contact the Informatica Documentation team through email at infa_documentation@informatica.com.

Informatica Product Availability Matrixes

Product Availability Matrixes (PAMs) indicate the versions of operating systems, databases, and other types of data sources and targets that a product release supports. If you are an Informatica Network member, you can access PAMs at https://network.informatica.com/community/informatica-network/product-availability-matrices.

Informatica Velocity

Informatica Velocity is a collection of tips and best practices developed by Informatica Professional Services. Developed from the real-world experience of hundreds of data management projects, Informatica Velocity represents the collective knowledge of our consultants who have worked with organizations from around the world to plan, develop, deploy, and maintain successful data management solutions.

If you are an Informatica Network member, you can access Informatica Velocity resources at http://velocity.informatica.com.

If you have questions, comments, or ideas about Informatica Velocity, contact Informatica Professional Services at ips@informatica.com.

Informatica Marketplace

The Informatica Marketplace is a forum where you can find solutions that augment, extend, or enhance your Informatica implementations. By leveraging any of the hundreds of solutions from Informatica developers and partners, you can improve your productivity and speed up time to implementation on your projects. You can access Informatica Marketplace at https://marketplace.informatica.com.

Informatica Global Customer Support

You can contact a Global Support Center by telephone or through Online Support on Informatica Network.

To find your local Informatica Global Customer Support telephone number, visit the Informatica website at the following link:
http://www.informatica.com/us/services-and-training/support-services/global-support-centers.

If you are an Informatica Network member, you can use Online Support at http://network.informatica.com.

CHAPTER 1

Introduction to Dynamic Data Masking Accelerator for use with SAP

This chapter includes the following topics:

Accelerator Overview

Use the accelerator to implement Dynamic Data Masking with an SAP application. The accelerator package contains predefined Dynamic Data Masking connection and security rules for common masking requirements.

The accelerator is in the Dynamic Data Masking installation folder as an additional component that you can configure to work with an SAP application. You can use the accelerator to mask data based on the application user that accesses the SAP application.

You must use an SAP client application to use the accelerator.

Dynamic Data Masking with the SAP Client Process

SAP caches SQL requests sent to the database. Dynamic Data Masking Accelerator for use with SAP stores the modified SQL request in the SAP cache.

The Dynamic Data Masking Server is used once for each unique SQL statement. After Dynamic Data Masking modifies a statement, the SAP cache stores the modified statement. When you send a request to the database, the SAP engine searches the cache for the request. If the request is not in the cache, the request goes through the Dynamic Data Masking Server

CHAPTER 2

Accelerator Setup

This chapter includes the following topics:

Accelerator Setup Overview

Set up the accelerator to use predefined connection and security rules.

You can find the accelerator in the following directory:

```
<Dynamic Data Masking installation>\Accelerators\SAP
```

To set up the accelerator, perform the following tasks:

1. Verify the requirements.
2. Deploy the Java servlet.
3. Create a Dynamic Data Masking database administrator user.
4. Compile the database objects.
5. Configure the Cache Reset Tool.
6. Create a Dynamic Data Masking Oracle database connection.
7. Import the accelerator connection rules.
8. Import the accelerator application security rules.
9. Create accelerator development security rules.
10. Reset the SAP cache.

Verify Requirements

Verify the following requirements before you use the accelerator:

- The Dynamic Data Masking Server and Management Console version 9.1.0 or later must be installed.
- You must have an Oracle database configured for SAP.
- The SAP Java Connector must be installed.
- Apache Tomcat version 7.0 or later must be installed.

Install SAP Java Connector

You must have SAP Java Connector installed on the machine that hosts the Dynamic Data Masking Server and the machine that hosts the Dynamic Data Masking Java servlet.

You can download SAP Java Connector from the following URL:

`http://service.sap.com/connectors`

Download SAP Java Connector version 2.1 or later and choose the file for the platform that hosts the Dynamic Data Masking server or Apache Tomcat web server.

Install Apache Tomcat

Apache Tomcat Server version 7.0 or later must be installed to use the accelerator.

You can download Apache Tomcat from the following URL:

`http://tomcat.apache.org/download-70.cgi`

Java 1.6 or later must be installed on the machine with Apache Tomcat.

Install the Apache Tomcat Windows service version on Windows. Install Apache Tomcat as the root user on Linux and UNIX.

After you install Apache Tomcat, verify that the JAVA_HOME environment variable is defined.

Deploy the Java Servlet

You must deploy the Dynamic Data Masking Java servlet to use the accelerator.

You can find the Java servlet in the following location:

`<Dynamic Data Masking installation>\Accelerators\SAP\lib`

Deploy the Java Servlet on Windows

Deploy the Dynamic Data Masking Java servlet on Windows.

The Apache Tomcat Windows operating service must be stopped.

1. Copy ddmsap.war from the following directory:

   `<Dynamic Data Masking installation>\Accelerators\SAP\lib\ddmsap.war`

2. Paste ddmsap.war into the following directory:

```
<Apache Tomcat installation>\webapps
```

3. Copy sapjco.jar from the SAP Java Connector installation directory.
4. Paste sapjco.jar into the following directory:

   ```
   <Apache Tomcat installation>\lib
   ```

5. Start the Apache Tomcat Windows operating system service.

   The ddmsap war file deployment on Apache Tomcat creates the following directory:
   ```
   <Apache Tomcat installation>\webapps\ddmsap
   ```

6. Stop the Apache Tomcat Windows operating system service.
7. Open logon.properties from the following location:

   ```
   <Apache Tomcat installation>\webapps\ddmsap\WEB-INF\classes\logon.properties
   ```

8. Modify the logon.properties properties.

   The following table describes the properties:

Property	Description
jco.client.client	SAP client number.
jco.client.user	SAP user name.
jco.client.passwd	SAP user password.
jco.client.language	Language. Default is en.
jco.client.ashost	IP address of the SAP environment server.
jco.client.sysnr	System number in the specified host. Range is 00 to 99.

9. Start the Apache Tomcat Windows operating system service.
10. Open a browser and navigate to the following URL:

    http:<Apache Tomcat IP address>:<port>/ddmsap/getsapuserdetails?processid=123

    The browser displays null.

Deploy the Java Servlet on Linux and UNIX

Deploy the Dynamic Data Masking Java servlet on Linux and UNIX.

The Apache Tomcat server must be stopped.

1. Copy ddmsap.war from the following directory:

   ```
   <Dynamic Data Masking installation>/Accelerators/SAP/lib/ddmsap.war
   ```

2. Paste ddmsap.war into the following directory:

   ```
   <Apache Tomcat installation>\webapps
   ```

3. Copy sapjco.jar from the SAP Java Connector installation directory.
4. Paste sapjco.jar into the following directory:

   ```
   <Apache Tomcat installation>/lib
   ```

5. Navigate to the following directory:

   ```
   <Apache Tomcat installation>/bin
   ```

6. Run the following command to start the Apache Tomcat server:

   ```
   ./catalina.sh start
   ```

   The ddmsap war file deployment on Apache Tomcat creates the following directory:

   ```
   <Apache Tomcat installation>/webapps/ddmsap
   ```

7. Stop the Apache Tomcat server.
8. Open logon.properties from the following location:

   ```
   <Apache Tomcat installation>\webapps\ddmsap\WEB-INF\classes\logon.properties
   ```

9. Modify the logon.properties properties.

   The following table describes the properties:

Property	Description
jco.client.client	SAP client number.
jco.client.user	SAP user name.
jco.client.passwd	SAP user password.
jco.client.language	Language. Default is en.
jco.client.ashost	IP address of the SAP environment server.
jco.client.sysnr	System number in the specified host. The range is 00 to 99.

10. Start the Apache Tomcat server.
11. Open a browser and navigate to the following URL:

    ```
    http:<Apache Tomcat IP address>:<port>/ddmsap/getsapuserdetails?processid=123
    ```

    The browser displays null.

Create Dynamic Data Masking Database Administrator User

Create a Dynamic Data Masking database administrator user to use the accelerator.

Connect to the database with the sys user and run the following database commands:

CREATE USER DDMUSER IDENTIFIED BY XXXX

ALTER USER DDMUSER QUOTA UNLIMITED ON USERS

GRANT BECOME USER TO DDMUSER

GRANT CREATE SESSION TO DDMUSER

GRANT ALTER SESSION TO DDMUSER

GRANT SELECT ANY TABLE TO DDMUSER

GRANT SELECT ANY DICTIONARY TO DDMUSER

GRANT EXECUTE ON UTL_HTTP TO DDMUSER

GRANT DBA TO DDMUSER

Note: The database commands create a database user with the user name DDMUSER and password XXXX. You can change the values of the user name and password.

Compile the Database Objects

Use the SQL scripts in the SAP folder to compile the database objects.

You can find the SQL script files in the following directory:
`<Dynamic Data Masking installation>\Accelerators\SAP\sql\Oracle`

1. Connect to the database with the Dynamic Data Masking database user you created to use the accelerator.
2. Navigate to the following directory:

   `<Dynamic Data Masking installation>\Accelerators\SAP\sql\Oracle`
3. Open DDM_SAP_MATCHERS_TBL.sql and replace `<schema>` with the Dynamic Data Masking user you created to use the accelerator.
4. Save DDM_SAP_MATCHERS_TBL.sql and close the file.
5. Create the DDM_SAP_MATCHERS_TBL table with the following script:

   `DDM_SAP_MATCHERS_TBL.sql`
6. Compile the SAP matchers package spec with the following script:

   `DDM_SAP_MATCHERS_spec.sql`
7. Compile the SAP matchers package body with the following script:

   `DDM_SAP_MATCHERS_body.sql`
8. Grant execute privileges to all users with the following database command:

   `GRANT EXECUTE ON DDM_SAP_MATCHERS TO PUBLIC`
9. Create a public synonym for the SAP matchers package with the following database command:

   `CREATE PUBLIC SYNONYM DDM_SAP_MATCHERS FOR DDM_SAP_MATCHERS`
10. Call the DDM_SAP_MATCHERS.SET_BASE_URL function to set the base URL of the Apache Tomcat Web Server with the following command:

    ```
    Begin
    Ddm_Sap_Matchers.Set_Base_Url(<Base Web Server URL>)
    End;
    ```

    The base URL is in the following format:

    `http://<Server IP/Name>:<Port>`

    The following URL is an example of a base URL:

    `http://localhost:8080`
11. Verify that you entered the correct base URL with the following select statement:

    `Select DDM_SAP_MATCHERS.GET_BASE_URL From Dual`
12. Verify that the database objects are compiled with the following select statement:

    `Select DDM_SAP_MATCHERS.user_match('abcd') from dual`

    The returned value is zero.

Create Access Control List for Oracle 11G

If you have an Oracle 11G database, you must create an access control list to enable an outgoing network with utl_http.

A database administrator must create the access control list because it impacts applications that use utl_http.

Use the following script to create the access control list:

```
begin
DBMS_NETWORK_ACL_ADMIN.CREATE_ACL (
    acl          => 'www.xml', --access control list file name (.xml)
    description  => 'Permissions to access tomcat',
    principal    => 'SYSTEM', --replace with the DDM user
    is_grant     => TRUE,
    privilege    => 'connect');
   COMMIT;


   DBMS_NETWORK_ACL_ADMIN.ASSIGN_ACL (
    acl          => 'www.xml',   --the file name you gave in the precedent
command
    host         => '*'); --which host is available for the access control
   COMMIT;
End ;
```

Configure the Cache Reset Tool

Configure the Cache Reset Tool to clear the SAP cache. If you use the Cache Reset Tool, you do not need to restart the SAP environment to clear the SAP cache.

Reset the SAP cache when you modify the Dynamic Data Masking accelerator security rules.

You must have the SAP Java Connector installed before you can use the Cache Reset Tool.

Before you configure the Cache Reset Tool, import the transport functions into the SAP environment. You can find the transport functions in the following location:

```
<Dynamic Data Masking installation>\Accelerators\SAP\ThirdParty
```

Configure the Cache Reset Tool on Windows

Configure the Dynamic Data Masking Cache Reset Tool on Windows.

1. Open resetsapcache.bat from the following directory:

   ```
   <Dynamic Data Masking installation>\Accelerators\SAP\resetsapcache.bat
   ```

2. Add sapco.jar to the Java class path in resetsapcache.bat.

   The following sapco.jar excerpt shows the Java class path:

   ```
   SET CP="$myRoot/lib/ResetSapCache.jar:jar;<SAP Java Connector installation>/sapjco.jar"
   ```

3. Open sample_logon.properties from the following directory:

   ```
   <Dynamic Data Masking installation>\Accelerators\SAP\sample_logon.properties
   ```

4. Edit the sample_logon.properties properties.

The following table describes the properties:

Property	Description
jco.client.client	SAP client number.
jco.client.user	SAP user name.
jco.client.passwd	SAP user password.
jco.client.language	Language. Default is en.
jco.client.ashost	IP address of the SAP environment server.
jco.client.sysnr	System number in the specified host. The range is 00 to 99.

5. Save sample_logon.properties as logon.properties in the following directory:

   `<Dynamic Data Masking installation>\Accelerators\SAP`

6. Run the following command to clear the SAP cache:

   `<Dynamic Data Masking installation>\Accelerators\SAP` \resetsapcache.bat

Configure the Cache Reset Tool on Linux and UNIX

Configure the Dynamic Data Masking Cache Reset Tool on Linux and UNIX.

1. Open the resetsapcache file from the following directory:

   `<Dynamic Data Masking installation>/Accelerators/SAP`

2. Add sapco.jar to the Java class path in the resetsapcache file.

   The following sapco.jar excerpt shows the Java class path:

   ```
   SET CP="lib\ResetSapCache.jar;<SAP Java Connector installation>\sapco.jar
   ```

3. Open sample_logon.properties from the following directory:

   `<Dynamic Data Masking installation>/Accelerators/SAP/sample_logon.properties`

4. Edit the sample_logon.properties properties.

   The following table describes the properties:

Property	Description
jco.client.client	SAP client number.
jco.client.user	SAP user name.
jco.client.passwd	SAP user password.
jco.client.language	Language. Default is en.
jco.client.ashost	IP address of the SAP environment server.
jco.client.sysnr	System number in the specified host. The range is 00 to 99.

5. Save sample_logon.properties as logon.properties in the following directory:

   ```
   <Dynamic Data Masking installation>/Accelerators/SAP
   ```

6. Run the following command to clear the SAP cache:

   ```
   <Dynamic Data Masking installation>/Accelerators/SAP/ResetSapCache
   ```

Create a Database Connection

Create a Dynamic Data Masking database connection for Oracle in the Management Console.

1. Log in to the Dynamic Data Masking Management Console.
2. Highlight the Dynamic Data Masking Server in the Management Console tree and select **Tree** > **Add DDM Services**.

   The **Add DDM Services** window appears.
3. Check **DDM for Oracle** and click **OK.**

   The **DDM for Oracle** node appears in the Management Console tree.
4. Highlight a domain node or the Management Console tree root node and select **Tree** > **Add Database**.

   The **Add Database** window appears.
5. Select the Oracle database type.
6. Configure the Oracle database connection parameters.

   The following table describes the database connection parameters:

Parameter	Description
Database Name	Logical name defined for the target database.
Instance Name	Instance name for the target database.
Listener Address	Server host name or TCP/IP address for the target database.
Listener Port	TCP/IP listener port for the target database.
Listener Service Name	Service name for the target database.
DBA Username	The user name of the user you created to use the accelerator.
DBA Password	Password for the user you created to use the accelerator.

7. Click **Test Connection** and verify that Dynamic Data Masking is connected to the database.
8. Click **OK**.

   The database node appears in the Management Console.

Import Connection Rules

Import the accelerator connection rules into the Management Console.

1. Highlight the DDM for Oracle node in the Management Console tree and select **Tree** > **Connection Rules**.

   The Rule Editor opens.
2. In the Rule Editor, highlight the DDM for Oracle node in the tree and select **Action** > **Import**.

   The **Import** window opens.
3. Navigate to the following directory:

   `<Dynamic Data Masking installation>\Accelerators\SAP\rules\Oracle`
4. Select SAPConnectionRules.xml and click **Import**.

   The SAPAppConnRule and SAPDevConnRule connection rules appear in the Rule Editor.
5. Select **File** > **Update Rules** to save the connection rules.
6. Select **File** > **Exit** to close the Rule Editor.

Note: If you modify the SAP connection rules, you must log out of the SAP application and log in again.

Import Application Security Rules

Import the accelerator application security rules into the Management Console.

1. Highlight the Management Console tree root node and select **Tree** > **Add Rule Set**.

   The **Add Rule Set** window opens.
2. Enter SAPAppRuleSet as the rule set name and click **OK**.

   The SAPAppRuleSet rule appears in the Management Console tree.
3. Highlight the SAPAppRuleSet rule set and select **Tree** > **Security Rule Set**.

   The Rule Editor opens.
4. In the Rule Editor, select **Action** > **Import**.

   The **Import** window opens.
5. Navigate to the following directory:

   `<Dynamic Data Masking installation>\Accelerators\SAP\rules\Oracle`
6. Select SAPAppRuleSet.xml and click **Import**.

   The MatchSAPTables rule folder appears in the Rule Editor.
7. Expand the MatchSAPTables rule folder to view the UserHandling folder.
8. Expand the UserHandling folder to view the CustomerMasking, EmployeeMasking, VendorMasking, BlackList, and WhiteList folders.
9. Define masking rules in the Rule Editor. Highlight a rule and select **Action** > **Edit** to open the **Edit Rule** window.
10. Select **File** > **Update Rules** to save the security rules.
11. Select **File** > **Exit** to close the Rule Editor.

If you modify the accelerator security rules, you must reset the SAP cache.

Create Development Security Rules

Create the accelerator development security rules in the Management Console.

1. Highlight the Dynamic Data Masking Server node in the Management Console tree and select **Tree** > **Add Rule Set**.

   The **Add Rule Set** window opens.
2. Enter SAPDevRuleSet as the rule set name and click **OK**.

   The SAPDevRuleSet rule set appears in the Management Console tree.
3. Highlight the SAPDevRuleSet rule set and select **Tree** > **Security Rule Set**.

   The Rule Editor opens.
4. Highlight the SAPDevRuleSet rule set and select **Action** > **Import** to open the **Append Rule** window and edit the rule.
5. Create security rules for SAP development tools. Select **Action** > **Insert Rule** to create new rules.
6. Select **File** > **Update Rules** to save the security rules.
7. Select **File** > **Exit** to close the Rule Editor.

If you modify the accelerator security rules, you must reset the SAP cache.

Reset SAP Cache

If you modify the SAPAppRuleSet or SAPDevRuleSet security rules, you must reset the SAP cache. You must configure the Cache Reset Tool before you use the tool.

Run the following command to reset the SAP cache:

```
<Dynamic Data Masking installation>\Accelerators\SAP\ResetSapCache.bat
```

CHAPTER 3

Accelerator Rules

This chapter includes the following topics:

Accelerator Rules Overview

The accelerator contains an Oracle connection rule set and multiple security rule sets. The rule sets contain predefined rules that are common masking techniques.

The connection rule set contains rules that direct the SQL statement to the security rule sets. The security rule sets determine how the data is masked. You can modify the rules to alter the masking techniques.

The following table describes the accelerator rule sets:

Rule Set	Description
Connection Rule Set	An Oracle connection rule set that directs SQL requests to the SAPDevRuleSet and SAPAppRuleSet security rule sets.
SAPDevRuleSet	A security rule set that masks rules that do not come from an SAP client. You use the SAPDevRuleSet for SQL requests from the development team.
SAPAppRuleSet	A security rule set that masks rules that come from an SAP client. The SAPAppRuleSet masks requests sent from non-development team users.

Connection Rules

The accelerator connection rules direct SQL requests to security rule sets. The connection rules are SAPDevConnRule and SAPAppConnRule.

The SAPDevConnRule connection rule matches SQL requests that do not come from an SAP client. It directs requests to the SAPDevRuleSet security rule set. You can use SAPDevConnRule to direct requests from development tools such as Toad and SQL Developer.

The SAPAppConnRule connection rule matches SQL requests that come from an SAP client. It directs requests to the SAPAppRuleSet security rule set. You can use SAPAppConnRule to direct requests that come from non-development tools.

The following table describes the accelerator connection rules:

Rule	Description
SAPDevConnRule	Directs requests that do not come from an SAP client to the SAPDevRuleSet security rule set.
SAPAppConnRule	Directs requests that come from an SAP client to the SAPAppRuleSet security rule set.

SAPDevConnRule Connection Rule

The SAPDevConnRule connection rule directs all database requests that do not come from an SAP client to the SAPDevRuleSet security rule set. You can use the SAPDevConnRule rule to direct requests that come from the development team.

The SAPDevConnRule connection rule uses the Client/Application Information matcher to identify incoming requests. The matcher contains items on the exclude list that identify SAP clients. The connection rule is applied to any request that comes from a tool that is not on the exclude list. The rule action specifies that Dynamic Data Masking uses the SAPDevRuleSet security rule set for the request.

For example, if the development team accesses the database through SQL Developer, the SAPDevConnRule rule directs the request to the SAPDevRuleSet security rule set.

SAPAppConnRule Connection Rule

The SAPAppConnRule connection rule directs all database requests from an SAP client to the SAPAppRuleSet security rule set. You can use the SAPAppConnRule rule for non-development purposes.

The SAPAppConnRule connection rule uses the Client/Application Information matcher to identify incoming requests. The matcher contains items on the include list that identify SAP clients. The connection rule is applied to any request that comes from a tool that is on the include list. The rule action specifies that Dynamic Data Masking uses the SAPAppRuleSet security rule set for the request.

For example, if a user accesses the database through an SAP client, the SAPAppConnRule rule directs the request to the SAPAppRuleSet security rule set.

SAPDevRuleSet Security Rule Set

The SAPDevRuleSet security rule set receives requests from the SAPDevConnRule connection rule if the request does not come from an SAP client.

The SAPDevRuleSet security rule set does not contain any rules. Create masking rules in the SAPDevRuleSet security rule set that mask data when a request comes from a development tool.

For example, the development team works from 8:00 a.m. to 5:00 p.m. You want to block access to the database from development tools when the development team is not at work. Create a security rule in the SAPDevRuleSet security rule set that uses the Time of Day matcher to block requests to the database at all other times of day.

SAPAppRuleSet Security Rule Set

The SAPAppRuleSet security rule set receives SQL requests from the SAPAppConnRule connection rule if the request comes from an SAP client. You can mask data based on the type of user that accesses the database.

The following table describes the rule folders in the SAPAppRuleSet security rule set:

Rule Folder	Description
BlackList	Contains the BlackListDefinition and ExecuteUnion rules.
CustomerMasking	Contains masking rules for customers.
EmployeeMasking	Contains masking rules for employees.
UserHandling	Contains masking rules based on the application user that accesses the data.
VendorMasking	Contains masking rules for vendors.
WhiteList	Contains the WhiteListDefinition and ExecuteUnion rules.

The following table describes the rules in the SAPAppRuleSet security rule set:

Rule	Description
BlackListDefinition	Defines the users that receive masked data. Separate users with a comma.
ExecuteUnion	Rewrites the SQL request.
RULE_EMAIL_MASK	Masks email addresses.
RULE_NAMEMASK	Masks names.
RULE_STREET_MASK	Masks addresses.
RULE_TELENO	Masks telephone numbers.
SaveOrig	Saves the original SQL request.
WhiteListDefinition	Defines the users that receive unmasked data. Separate users with a comma.

BlackList Rule Folder

The BlackList rule folder is a security rule folder that defines the users that receive masked data.

You can define the list of BlackList users in the BlackListDefinition security rule. The security rule uses the Define Symbol action to create a list of users that view masked data. Separate user names with a comma.

If you want to mask data based on the users in the BlackList folder, enable the BlackList folder and disable the WhiteList folder.

For example, you have 100 users. You want to mask data when 20 of the users access the database. Enable the BlackList folder and disable the WhiteList folder. List the 20 users in the BlackListDefinition rule. Dynamic Data Masking will mask data when a request is sent from one of the users in the BlackListDefinition rule.

WhiteList Rule Folder

The WhiteList rule folder is a security rule folder that defines the users that receive unmasked data.

You can define the list of WhiteList users in the WhiteListDefinition security rule. The security rule uses the Define Symbol action to create a list of users that view unmasked data. Separate user names with a coma.

If you want to mask data based on the users in the WhiteList folder, enable the WhiteList folder and disable the BlackList folder.

For example, you want the database administrator to view unmasked data. You want everyone else to see masked data. Enable the WhiteList folder and disable the BlackList folder. List the database administrator in the WhiteListDefinition rule. Dynamic Data Masking will mask data for everyone that is not the database administrator. The database administrator will view unmasked data.

INDEX