CONFIDENTIALITY NOTICE

THIS DOCUMENT IS CONFIDENTIAL AND PROPRIETARY TO NEXTLABS, INC. AND MAY NOT BE REPRODUCED, PUBLISHED OR DISCLOSED TO OTHERS WITHOUT COMPANY AUTHORIZATION.

© 2009-2016 NextLabs, Inc. All rights reserved. The information in this document is subject to change without notice.

To provide feedback on this document, email NextLabs, Inc. at techpubs@nextlabs.com.

TRADEMARKS

NextLabs®, ACPL®, Enterprise Data Protection™, and the Enterprise Data Protection logo are registered trademarks of NextLabs, Inc. All other brands or product names used herein are trademarks or registered trademarks of their respective owners.

LICENSE AGREEMENT

This documentation and the software described in this document are furnished under a license agreement or nondisclosure agreement. The documentation and software may be used or copied only in accordance with the terms of those agreements. No part of this document may be reproduced, stored in a retrieval system or transmitted in any form or by any means electronic or mechanical, including photocopying and recording for any purpose other than the purchaser's use, without the prior written permission of NextLabs, Inc.

The content of this document is provided for informational and instructional use only. It is subject to change without notice, and should not be construed as a commitment by NextLabs, Inc.

NextLabs, Inc. assumes no responsibility or liability for any inaccuracies or technical errors that may appear in the content of this document.

Published in San Mateo, CA, by NextLabs, Inc.
www.nextlabs.com
info@nextlabs.com
support@nextlabs.com
650.577.9101

Document Revision Number: EDRM-SAP 7.7
Contents

1 Introduction ............................................................................................................. 7
   About NextLabs Enterprise Digital Rights Management for SAP ............................. 7
   Features ............................................................................................................. 8
      Read Tags .................................................................................................. 8
      User-Driven Rights Management ............................................................... 8
      Protect Data Leaving the SAP System (Outbound) ........................................ 8
      Automated Digital Rights Management .................................................... 9
   Classifying objects ......................................................................................... 9
   The Document Workflow ................................................................................ 9
      Classification of Business Objects .............................................................. 10
      Document Control .................................................................................... 10
      Enterprise Digital Rights Management .................................................... 10
      Auditing .................................................................................................... 11
   Overview of the Components .......................................................................... 11
      About the Policy Controller ..................................................................... 12
      Policy Evaluation ..................................................................................... 12
      Monitoring and Auditing .......................................................................... 13
      Policy Controller Functional Components .............................................. 13
      About the Policy Controller Communication Interface ............................. 14
      Policy Controller Communication Interface .......................................... 14
      SAP Policy Model .................................................................................... 14
   Contacting Technical Support ......................................................................... 15

2 Installation and Set Up ..................................................................................... 17
   Explanation of User Roles .............................................................................. 17
   Enterprise Digital Rights Management Installation Procedures ...................... 18
   Before You Begin ........................................................................................... 18
      Supported Platforms and Products ............................................................ 18
      Firewall and Port Requirements .................................................................. 19
      Configuring Firewall Settings .................................................................... 19
      Installation Procedures ............................................................................. 21
   Installing the Policy Controller ....................................................................... 21
   Installing Enterprise Digital Rights Management for SAP .............................. 21
Contents

Making the NextLabs Namespace Modifiable .................................................. 23
Configuring the Policy Controller Communication Interface: Java Connectors .... 24
  Before You Install .................................................................................... 25
  Install C++ Runtime Environment on the Policy Controller Host ................. 25
  Define an SAP Gateway Host and Port ..................................................... 25
  Create an RFC User .................................................................................. 25
  Locate Required Values for Configuration .............................................. 26
Installing the Java Connector for the Policy Controller ................................. 27
  Script Installation of the Java Connector for the Server Policy Controller ...... 27
  Manual Installation of the Java Connector for the Server Policy Controller .... 30
Installing the Java Connector for EDRM ....................................................... 31
Configuring the RFC Connection for the Policy Controller ......................... 31
Configuring the RFC Connection for EDRM .............................................. 34
Configuring the Properties and Services Files Manually ............................. 36
  Configuring the SAP JavaSDK Properties File (Manual installation option only) . 36
  Configuring the SAP JavaSDK Properties File (Required for all installation types) . 37
  Configuring the SAPJCo-EDRM Properties File ..................................... 38
  Configuring the Windows System Services File (Manual and Script Installation) 40
Testing the Java Connector Configuration ................................................... 40
  Testing the Connection from the Policy Controller Side ............................ 41
3 Configuration ......................................................................................... 43
  Configuring NextLabs Control Center for EDRM .................................... 43
    Enrolling Users from SAP into the Policy Server ................................... 43
    Extracting User Data from SAP .............................................................. 43
    Uploading SAP User Data into Control Center ...................................... 46
    Configuring SAP Actions ..................................................................... 47
    Configuring SAP Obligations ............................................................... 49
  Configuring NextLabs EDRM ................................................................. 55
Using the NextLabs Configuration Tool ...................................................... 56
  Checking Configuration Status ............................................................... 57
Activating the Entitlement Pack .................................................................. 59
Adding Composite Keys and Classification Values .................................... 60
  Security Identifiers ................................................................................. 60
  Composite Keys ..................................................................................... 60
  Classification Values ............................................................................ 60
  Adding Composite Keys ....................................................................... 61
  Adding New Classification Values ........................................................ 63
Linking Composite Keys (SECENH) ............................................................ 65
Mapping Security Fields (SECMPG) ........................................................... 66
Configuring Security Identifier/Composite Key Value Tables (EPVAL) ......... 68
Mapping Transaction Codes and UI Functions to Actions (ACTIONS) ........... 69
## Contents

- Example Policy: Read Tags ................................................................. 122
- Example Policy: Classify ................................................................. 124
- Example Policy: Protect ................................................................. 126
- Example Policy: Protect and Classify ............................................. 128
- Example Policy: Remove Protection ............................................... 130
- Applying Security Classifications ................................................. 132
- About Classification Data ............................................................... 132
- About Compound Classification Keys ............................................ 132
- Best Practices for Retrieving Security Classification Records .......... 133
- Viewing Security Classification Records ........................................ 133
- Applying Security Classifications Manually ................................... 135

### 6 Administration and Maintenance ............................................. 137

- Maintenance for Entitlement Manager for SAP ECC ....................... 137
- Viewing NextLabs Log Information in SAP .................................... 138
- Configuration and Management ..................................................... 140
  - Configuration Tools ................................................................. 140
  - Management Activities ............................................................ 140
    - Stopping and Starting Entitlement Managers ........................... 141
    - Monitoring Enforcers .......................................................... 141
    - Uninstalling, Repairing, or Modifying Policy Controllers and Enforcers .................................................. 142
- About Service Account Permissions ............................................. 143
- About Bundle Encryption ............................................................ 143
  - Authentication Failure .............................................................. 144
  - Decrypting the Bundle: Policy Controller .................................. 144
  - Decrypting the Bundle: Java Policy Controller .......................... 145
- Managing Enforcer Policies ......................................................... 146
- Managing Event Logging ............................................................. 146
  - Logging Settings ........................................................................ 147
  - Changing Logging Levels ......................................................... 147
- Load Balancing the Policy Controller .......................................... 148
  - Example: Load Balancing Configuration ..................................... 149
- Deleting Decrypted Temporary Documents .................................... 152

### 7 Custom Enhancements ............................................................. 153

- Dynamic User and Resource Attributes ....................................... 153
- Configuring Enhancement Activations for Dynamic Attributes ........ 154
- Referencing Dynamic Attributes in Policies ................................ 156
- BADI Implementation for Dynamic User or Resource Attribute ....... 156
This section introduces NextLabs Enterprise Digital Rights Management (EDRM) 7.7 for SAP.

Topics include:

- About NextLabs Enterprise Digital Rights Management for SAP
- Features
- Classifying objects
- Overview of the Components
- Contacting Technical Support

### About NextLabs Enterprise Digital Rights Management for SAP

NextLabs Enterprise Digital Rights Management (EDRM) is a platform-independent rights management solution that applies data protection and data classification to the documents in SAP systems, based on policies. This way, organizations can securely share their critical information internally and throughout the extended enterprise.

The NextLabs EDRM solution can read the pre-existing classifications from native or NextLabs-protected documents when they are uploaded into SAP and also provide protection against unauthorized use of SAP data even after the data leaves the SAP system. The documents remain protected even after they are downloaded from the SAP system because classification and protection are persistent with the documents.

Authorized users can use the User-Driven Rights Management (UDRM) features to classify and protect the documents. The documents are classified and protected based on policies.

Administrators can perform the same UDRM operations (Classify, Protect, or Classify and Protect) in an automated way by using Automated Digital Rights Management (ADRM). ADRM enables administrators to classify and/or protect multiple files in a single operation.
Chapter 1: Introduction

Features

NextLabs Enterprise Digital Rights Management (EDRM) for SAP provides these features:

Read Tags

The Read Tags feature, based on the policy decision, reads the pre-existing classifications from native or NextLabs-protected documents when they are uploaded into SAP and then inserts those classifications into the corresponding columns of the Security Classification Maintenance table. For more information, see Reading Tags.

User-Driven Rights Management

The User-Driven Rights Management (UDRM) context menu options (Classify, Protect, Protect and Classify, and Remove Protection) are available from the SAP context menu.

Classify

The Classify option enables authorized users to classify native or NextLabs-protected documents in the SAP system, based on the policy decision. For more information, see Applying Classification to Documents.

Protect

The Protect option enables authorized users to encrypt native documents in the SAP system into NextLabs-protected documents, based on the policy decision. An already encrypted document cannot be further protected. For more information, see Protecting Documents.

Protect and Classify

The Protect and Classify option enables authorized users to protect (encrypt) native documents and simultaneously classify the documents, based on the policy decision. If the Protect and Classify option is run on any NextLabs-protected document, the system can only classify the document. For more information, see Applying Protection and Classification to Documents Simultaneously.

Remove Protection

The Remove Protection option, based on the policy decision, enables authorized users to view or edit NextLabs-protected documents in their native format by downloading the documents to a temporary location. The documents remain protected within SAP, ensuring that unauthorized users are unable to access them. For more information, see Removing Document Protection.

Protect Data Leaving the SAP System (Outbound)

NextLabs EDRM can automatically protect and classify the sensitive business data leaving (outbound) the SAP ERP system. This includes master data and transactional data that are accessed in SAP through transactions, reports, data dictionary, table browsers, smart forms, and queries. The system automatically protects, and optionally classifies, the data being downloaded and
saved as a local file or being sent to the SAP inbox. The documents remain protected even after they are downloaded from the SAP system because classification and protection are persistent with the documents. For more information, see Protecting Data Leaving the SAP System (Out-bound).

**Automated Digital Rights Management**

Administrators can use ADRM to protect and classify multiple documents, based on policies, in a single operation. The feature can be run in the manual or auto run mode and the reports can be run in the foreground or background processing mode. The documents remain protected even after they are downloaded from the SAP system because classification and protection are persistent with the documents. For more information, see Applying Protection and Classification Using Automated Digital Rights Management.

**Classifying objects**

You can manually create classifications in the Security Classification Maintenance table or use the Read Tags feature that reads the pre-existing external classifications when you upload originals into SAP. You can write classification policies that can be enforced by either using ADRM or UDRM (Classify or Protect and Classify).

**The Document Workflow**

There are four information-control objectives for managing the creation, access, and distribution of data, also known as the document workflow:

- **Classification of Business Objects**
- **Document Control**
- **Enterprise Digital Rights Management**
- **Auditing**
Classification of Business Objects

The Security Classification module manages data classifications and simplifies the process of identifying and maintaining classification values. Enterprise Digital Rights Management for SAP supports both manual, as well as automatic policy-based classification.

Document Control

Document control policies authorize or block user access to specific business data within SAP, based on both the Classification of Business Objects and selected user identity attributes. User alerts can be customized to provide authorized users with guidance and education on compliance restrictions and requirements.

Enterprise Digital Rights Management

Using Enterprise Digital Rights Management for SAP, originals uploaded into SAP can be classified automatically. Therefore, classifications applied to documents persist even when documents leave the SAP system. Using other NextLabs products, you can then design complementary policies that prevent controlled data from being used or distributed in unauthorized ways. A typical example is a policy blocking users from emailing documents that have been downloaded out of SAP to unauthorized users. In addition, EDRM enables users to design policies that automatically apply NextLabs encryption to documents, which adds an additional layer of persistent protection. For more information on how to construct EDRM policies that classify and apply NextLabs protection to files, see Designing EDRM Policies.
Auditing

Administrators need insight into the actions users are performing in SAP, including how users are creating and downloading data. Audit policies enable administrators to log these details.

Overview of the Components

NextLabs Enterprise Digital Rights Management for SAP includes the components shown in Figure 1-2.

**Figure 1-2: Overview of the Components**

**Policy Controller**

The Policy Controller provides the interface to the Control Center, and functions as the Policy Decision Point, or PDP. It comprises a set of software modules that are delivered ready-to-install and requires no customization.

**Policy Controller Communication Interface**

The Policy Controller Communication Interface provides the connection between the Policy Controller (PDP) and EDRM, which resides in SAP and serves as the Policy Enforcement Point, or PEP. The following sections provide more detail on these components.
Control Center

NextLabs Control Center is an Information Control Platform (ICP) that turns business policy into automated information controls for data access, use and sharing across server, client, and cloud applications. For more information, see NextLabs Control Center Administrator’s Guide.

Policy Studio

NextLabs Policy Studio is an application that you can use to define, deploy, monitor, and manage your data loss prevention policies. For more information, see NextLabs Policy Studio User’s Guide.

Rights Management Server

NextLabs EDRM for SAP is integrated with NextLabs Rights Management Server (RMS) and enables you to embed classifications into originals as they are uploaded into SAP. This way, classifications are present when documents are downloaded out of SAP. The Rights Management Server can apply NextLabs protection to documents. For more information, see NextLabs Rights Management Server Administrator’s Guide.

Rights Management Client

NextLabs Rights Management Client (RMC) is a separate NextLabs product that can be integrated with NextLabs EDRM. RMC enables authorized users to remove protection (decrypt) on the downloaded local copies of documents. For more information, see NextLabs Rights Management Client User’s Guide.

About the Policy Controller

The functions that the Policy Controller provides can be grouped into the categories:

- Policy Evaluation on page 12
- Monitoring and Auditing on page 13
- Policy Controller Functional Components on page 13

Policy Evaluation

Whenever a policy enforcement point (PEP) detects an event that may be covered by a currently deployed policy, it sends a request to the Policy Controller which acts as the policy decision point (PDP). The Policy Controller applies all context information to the events, and makes decisions on what policy applies and how. It then relays the effects of any relevant policy back to the PEP, which contains system-specific logic to apply the enforcement.

The PEP then instructs the application or file system to respond to the user’s action as the policy requires—either allow or deny. If the policy evaluation results in the action being denied, the Policy Controller returns a message indicating that access is denied or that the requested action cannot be performed. These may take the form of standard system errors, or the customized text balloon that was defined with the policy being enforced, or both. Depending on the policy, it may also be accompanied by an obligation being triggered.
Monitoring and Auditing

Every Policy Controller can monitor how users are accessing documents on that host, so that even if it never blocks any actions by policy subjects, it is still providing a valuable service of capturing who is using what information, how, and when. To do this, create and deploy Allow policies.

Policy Controller Functional Components

The functional components of the Policy Controller are represented in Figure 1-3.

- **The Policy Evaluation Engine** evaluates whether or not every user action is covered by any of the policies currently cached at that enforcement point. It bases this on multiple criteria such as who the user is, what host he is using, how he is connected to the network, what action is being attempted, and on what resource; the date, the time, and so on. It does this in real time, and operates continuously whether the host is connected to the network or not.

- **The Context Manager** keeps constant track of the environmental context of all events, and provides it to the Policy Engine and Policy Adapter. The context includes user identity, computer host name, network connection type, and date and time.

- For any policy that evaluates to True, the **Obligation Manager** initiates an obligation by sending a request to a policy adapter’s obligation services or executing built-in obligations. It contains three sub-components:
  - **Policy Logger** - Collects and logs all activity details and policy decision results.
  - **Messaging Services** - Sends message to recipients or targets listed in a policy.
  - **Application Extender** - Launches an application or custom executable that performs some custom obligation.

  **Note:** Although it is logically a part of the Policy Controller, the Obligation Manager runs as a separate process, visible in the Processes tab in Windows Task Manager.

- **The Control Manager** records non-policy activities, updates the configuration, and secures the controller. Components include:
  - **Activity Recorder** - Records activities tracked by the Policy Adapter in real time.
  - **Configuration Manager** - Applies profile and system configuration changes, in real time.
- **Policy Authentication** - Authenticates the policy set from the Control Center and encrypts it on the local file system.
- The **ICENet Client** provides the interface for all communication with the Control Center. It is used for deploying new or changed policies, periodically sending activity logs from each control point, and providing controller health status.

**About the Policy Controller Communication Interface**

The Policy Controller Communication Interface sends policy check event information from the PEP to the PDP, where policy evaluation occurs, and then sends the PDP policy decision and result back to the PEP, where enforcement occurs. This communication interface uses a NextLabs Java Connector (JCo) plug-in that integrates with the SAP Remote Function Call (RFC) interface.

**Policy Controller Communication Interface**

The Entitlement Manager for SAP implementation must include an interface that enables communication between the ABAP-based SAP back-end system (SAP ECC) and the NextLabs Policy Controller.

When a transaction occurs, transaction information is routed from the SAP Agent to the Policy Controller Communication Interface. The Interface routes that information to the Policy Controller, which checks the transaction and classification information against any pertinent policies. The Policy Controller sends the allow or deny result back through the communication interface to the SAP Agent. This communication interface is configured using the NextLabs Java Connector (JCo) plug-in. The NextLabs Java Connector is a middleware component that enables development of SAP-compatible components and applications in Java. It is configured with SAP ECC using the Remote Function Call (RFC) interface. For more information about configuring the Java Connector, see **Configuring the Policy Controller Communication Interface: Java Connectors** on page 24.

**SAP Policy Model**

A few SAP-specific components must be configured in NextLabs Control Center and Policy Studio. These components enable the integration of SAP Security Classification settings with policies created in NextLabs Policy Studio.

**SAP Obligations**

Included in a Policy Controller response might be a custom SAP Obligation. There are several options for custom obligations. One obligation displays an SAP message informing users that access is denied or allowed (with educational warnings on how sensitive data should be handled). Another custom obligation applies file tagging to an SAP document so that Security Classification settings persist even outside of SAP. In addition, custom obligations can be designed in ABAP code, so that any program created in ABAP an be triggered upon a policy evaluation.

SAP Obligations must be registered with the Control Center, as is discussed in **Configuring SAP Obligations**.
SAP Action

When creating SAP policies in Policy Studio, you might need to select SAP-specific actions, such as “Copy from SAP.” SAP actions must be registered with the Control Center, as discussed in Configuring SAP Actions.

Contacting Technical Support

For help with NextLabs products, email Technical Support at support@nextlabs.com.
This section describes the basic installation and setup procedures required for the EDRM for SAP.

Topics:
- Explanation of User Roles
- Enterprise Digital Rights Management Installation Procedures
- Before You Begin
- Installing the Policy Controller
- Installing Enterprise Digital Rights Management for SAP
- Making the NextLabs Namespace Modifiable
- Configuring the Policy Controller Communication Interface: Java Connectors

Note: After installation and setup is complete, perform the configuration steps listed in the Configuration section.

### Explanation of User Roles

Table 2-1 lists users and associated permissions that are required to install and configure EDRM for SAP. These roles are referenced in the tables that follow. The roles at your organization may vary. For example, an SAP Consultant might be responsible for the SAP ABAP Developer functions. However, the permission requirements are still relevant.

<table>
<thead>
<tr>
<th>Role</th>
<th>Description/Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>NextLabs Administrator</td>
<td>Must have Administrator privileges on the host where the NextLabs Control Center and Policy Controller are installed; must have Administrator Profile password and NextLabs Administrator password</td>
</tr>
<tr>
<td>Basis Administrator</td>
<td>Must have permissions to perform Basis administrator functions, including installing .car files and configuring enterprise services within SAP</td>
</tr>
<tr>
<td>SAP Functional Consultant</td>
<td>Must have access to all tables, transactions, and programs within the /NEXTLABS/ and /NXLEDRM/ namespace</td>
</tr>
<tr>
<td>SAP ABAP Developer</td>
<td>Must have a developer key</td>
</tr>
</tbody>
</table>
Enterprise Digital Rights Management Installation Procedures

Table 2-2 provides links to the procedures for installing and configuring EDRM for SAP and then performing the configuration.

Begin with the procedures in Table 2-2, then proceed to the procedures required for EDRM.

Note: These procedures do not include NextLabs Control Center installation, which is covered in the NextLabs Control Center Installation Guide.

Table 2-2: Procedures for Installing and Configuring EDRM for SAP

<table>
<thead>
<tr>
<th>Procedure(s)</th>
<th>System Access</th>
<th>Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>Installing the Policy Controller</td>
<td>Policy Controller Administrator Password</td>
<td>NextLabs Administrator</td>
</tr>
<tr>
<td>Configuring the Policy Controller Communication Interface: Java Connectors</td>
<td>IIS configuration</td>
<td>NextLabs Administrator and SAP Basis Administrator</td>
</tr>
<tr>
<td>Enrolling Users from SAP into the Policy Server</td>
<td>In SAP: /NEXTLABS/USER_EXTRACT and On Control Center: NextLabs Enrollment Manager</td>
<td>SAP Functional Consultant and NextLabs Administrator</td>
</tr>
<tr>
<td>Installing Enterprise Digital Rights Management for SAP</td>
<td>SAINT</td>
<td>SAP Basis Administrator</td>
</tr>
<tr>
<td>Configuring SAP Actions</td>
<td>Control Center Policy Server</td>
<td>NextLabs Administrator</td>
</tr>
<tr>
<td>Configuring SAP Obligations</td>
<td>Control Center Policy Server</td>
<td>NextLabs Administrator</td>
</tr>
</tbody>
</table>

Before You Begin

Before you begin, ensure that your environment meets the following requirements:

- Supported Platforms and Products
- Firewall and Port Requirements
- Installation Procedures

Supported Platforms and Products

The following platforms are supported:

- SAP ERP 6.0, EHP 6
- For the Policy Controller for Microsoft File Servers: Windows Server 2008, 64-bit; with Application Server Role, with ASP.net enabled
- NextLabs Control Center 7.7
Firewall and Port Requirements

Each Windows PC where the NextLabs Enforcer is being installed must be configured to enable communication between the Control Center over the required ports. These port settings are required to enable the Enforcer to send heartbeat messages to the Control Center and receive policy bundles in return. The ports are also used by the administrative applications that connect to the Enforcers for status updates and other information.

The following are the default port assignments:

- Push Deployment port: 2000
- Policy Controller port: 8443
- RMI Registry port: 1099. If your implementation uses the Java Policy Controller, an exception for the RMI Registry port. To change this port assignment, you need to modify the JavaSDKService.properties file.

Note: The port numbers listed here are defaults. If your Control Center uses different port assignments, the exception ports on each Windows PC must match the Control Center port assignments. The Push Deployment port in particular is special in that it can be set differently for each Enforcer profile, in the Enforcer Profile Configuration tab in the Control Center Administrator interface. For details, see the Control Center Administrator’s Guide.

Configuring Firewall Settings

Configure firewall settings on each Windows PC where the enforcer is being installed to allow connections to the appropriate ports.

Procedure

1 In the Windows Control Panel, select System and Security, then select Windows Firewall.
2 Select Advanced settings.
3 Select Inbound Rules, then select New Rule, as shown in Figure 2-1.
4 Select *Port* as the type of rule to create, then click *Next*.

5 Select *TCP*, then select *Specific local ports* and type the port number or numbers, as shown in Figure 2-2, then click *Next*.

![Figure 2-1: Creating a new inbound rule](image1)

![Figure 2-2: Specifying the ports](image2)
Installing the Policy Controller

You must install the Server Policy Controller. For more information about installing Policy Controllers, see the Installing Policy Controllers section in the NextLabs Control Center 7.7 Installation Guide.

Next steps
Installing Enterprise Digital Rights Management for SAP

Installing Enterprise Digital Rights Management for SAP

This section provides information on a new installation of NextLabs Enterprise Digital Rights Management for SAP. You must first install the NextLabs_Base_770_001.CAR file and then install the NextLabs_EDRM_770_001.CAR file. You can use the same procedure to install both the .CAR files.

Before you Begin

Make sure that you have obtained the necessary installation files and passwords from NextLabs Support.

NextLabs Enterprise Digital Rights Management for SAP is installed on SAP ECC using a standard .CAR file procedure.

Procedure

1. In the SAP interface, log in to client 000.
2. Enter transaction SAINT. The Add-on Installation Tool window displays a list of all preconfigured add-ons and systems.
3. Click Start to begin a new Add-on Installation.
4 Locate an installation package from **Installation Package > Load Package > From Front End**.

5 **Navigate to the** NextLabs_Base_770_001.CAR file, **then click Open**.

6 When prompted, click **Decompress** to extract the installer files. The installation screen appears.

7 Click **Start**.

8 Select the installation package from the installation queue.

9 On the screen that appears, click **Continue**. You do not need to enter any information.

10 When prompted to add Modification Adjustment Transports to the queue, click **No**.

11 When prompted, enter the appropriate password for the OCS package, and then click **OK**. The installation begins.

   **Note:** Contact NextLabs Support to obtain the necessary installation passwords.

12 When the installation is complete, do any of the following:

   - Click **Logs** to view the installation logs and confirm the installation was successful.
   - Click **Finish** to exit the installer. When you click **Finish**, the final screen displays informing you that the installation was successfully completed.

   **Note:** You can verify in the Status/Remarks that the installation was successful. You can also scroll through the components to locate the newly installed NextLabs Add-on.

13 Log off client 000 and log in to client 100.

   **Note:** If you encounter any installation errors due to conflicts with other NextLabs products that have been installed previously, perform the following steps:

   a Click the back arrow to exit the SAINT transaction.

   b Delete the decompressed queue from the SAINT transaction that received the error.

   c Enter transaction **/NEXTLABS/CRT**.

   d After the **CRT** transaction has completed successfully, go to step 1 and repeat the procedure to install the NextLabs_EDRM_770_001.CAR file.

**Next Steps**

*Making the NextLabs Namespace Modifiable*
Making the NextLabs Namespace Modifiable

Upon initial system setup, you must configure the NextLabs Namespace to be modifiable. This step must be performed prior to configuring the NextLabs Number Range. The NextLabs Namespace should be made unmodifiable after configuration is complete.

Note: The only modifications supported for the NextLabs namespace are officially released NextLabs product code. Customers should not store other modifications to code in the NextLabs namespace because it can result in installation and upgrade issues.

Procedure

1. In the SAP interface, enter transaction SE03. The Transport Organizer Tools screen appears.
2. Select Set System Change Option.
Chapter 2: Installation and Set Up

NextLabs Enterprise Digital Rights Management 7.7 for SAP • User’s Guide

In the System Change Option screen, scroll down to the NextLabs NameSpace.

In the Modifiable field, change Not Modifiable to Modifiable.

Click the Save button.

Next steps
Configuring the Policy Controller Communication Interface: Java Connectors

Configuring the Policy Controller Communication Interface: Java Connectors

Setting up the communication interface between the Policy Controller and SAP Agent involves the following procedures:
• Verify that all prerequisites are met (see Before You Install on page 25)
• Installing the Java Connector for the Policy Controller on page 27
• Installing the Java Connector for EDRM on page 31
• Configuring the RFC Connection for the Policy Controller on page 31
• Configuring the RFC Connection for EDRM on page 34
• Configuring the SAP JavaSDK Properties File (Manual installation option only) on page 36
• Configuring the SAPJCo-EDRM Properties File on page 38
• Configuring the Windows System Services File (Manual and Script Installation) on page 40
• Testing the Java Connector Configuration on page 40

Before You Install

The procedures discussed in this section must be performed before you set up the NextLabs Java Connectors.

Install C++ Runtime Environment on the Policy Controller Host

Before installing the Java Connectors on Windows, you must install runtime components of Visual C++ libraries.

For 64 bit machines:

• Install the 64-bit version of Microsoft Visual C++ 2005 SP1 Redistributable Package. This download is available for free from the Microsoft Download Center.
• Install the patch for Microsoft Visual C++ 2005 SP1. This download is available for free from the Microsoft Download Center.

For 32 bit machines:

• Install the 32-bit version Microsoft Visual C++ 2005 SP1 Redistributable Package. This download is available for free from the Microsoft Download Center.

Define an SAP Gateway Host and Port

The SAP Gateway Host name and port must be defined within SAP ECC. In the steps below, the host name and port must be inserted into a properties file used by the Java Policy Connectors.

Create an RFC User

An RFC user must already be configured within the SAP system (of type Communication, with the assigned Communication role), before you configure the Java Policy Connectors. In addition, the RFC User must have the following authorizations defined as RFC Name Values:

• /NEXTLABS/*
• RFC1
• SDIFRUNTIME
• SYST

In the procedures that follow, the RFC user must be inserted into a properties file used by the Java Policy Connectors.
Chapter 2: Installation and Set Up

Locate Required Values for Configuration

Before you install and configure the Java Connector, make sure you have all the values that you need to enter as part of set up. Required parameters are listed in Table 2-3, along with procedures for how to locate the values (where applicable). You need these values whether you are installing manually or using the install script.

Table 2-3: Required Parameters for Configuring the JavaSDK.properties File (for both Script Install and Manual Procedure)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Explanation</th>
<th>How to Retrieve</th>
</tr>
</thead>
<tbody>
<tr>
<td>pc_hostname</td>
<td>FQDN for host where the Policy Controller is installed</td>
<td>Contact IT</td>
</tr>
<tr>
<td>ashost</td>
<td>Host where SAP ECC is installed</td>
<td>Contact IT</td>
</tr>
<tr>
<td>user</td>
<td>RFC user</td>
<td>Contact IT</td>
</tr>
<tr>
<td>passwd</td>
<td>Password for the RFC User</td>
<td>If you are performing a manual install, you must encrypt the RFC User password (the script installation does this automatically, so you can just enter the plain text version of the password in the installation menu. For a manual installation, this password should be encrypted using the mkpassword.bat utility supplied with the installation of the Control Center: • On the host where the Control Center is installed, open a command prompt and browse to &lt;install dir&gt;Control Center/tools/crypt. • Run mkpasswd.bat -password &lt;plain text password&gt; • The encrypted version of the plain text password appears. Copy it for use when you configure the SAP JavaSDK Properties File manually.</td>
</tr>
<tr>
<td>gwhost</td>
<td>FQDN of host where SAP ECC is installed</td>
<td>Contact IT</td>
</tr>
<tr>
<td>gwserv and port</td>
<td>The Gateway Service Name and port defined within SAP ECC</td>
<td>This gateway service name is configured in the following location: • Enter transaction SMGW. • Navigate to Goto &gt; Expert functions &gt; Host Name Buffer. • At the bottom of the screen, Gateway Service Ports and Names are defined.</td>
</tr>
<tr>
<td>progrid</td>
<td>The program ID for the Remote Function Call defined for the Java Connector communication interface</td>
<td>This program ID was defined in the procedures discussed in the section Configuring the RFC Connection for the Policy Controller.</td>
</tr>
<tr>
<td>&lt;system&gt;</td>
<td>The SAP system name</td>
<td>Contact the Basis Administrator</td>
</tr>
</tbody>
</table>
Installing the Java Connector for the Policy Controller

You can install the Java Connector for the Policy Controller using a script or manually.

- Script Installation of the Java Connector for the Server Policy Controller
- Manual Installation of the Java Connector for the Server Policy Controller

Script Installation of the Java Connector for the Server Policy Controller

If you are installing the Java Connector for Policy Controller with the Server Policy Controller, a script is available that automatically installs and uninstalls the Java Connector for Policy Controller files and adds the required values to the properties file. This script is only available for Server Policy Controllers running on Windows Server 2008 and 2012, and it must be run from the command line.

Note: Although the script automatically installs and uninstalls files, you still need to enter values in the Properties file when running the script.

Before You Begin

- Obtain the Java Connector for Policy Controller installation file from NextLabs. For more information, contact NextLabs Technical Support at support@nextlabs.com.
- To stop the Policy Controller, you must have the profile password defined in the Control Center Administrator interface.

Procedure

1. Stop the Policy Controller, if it is running.
2. Locate the Java Connector for Policy Controller installation file from NextLabs support (SAPJCo-EntitlementManager-<version number>.zip).
3. Open a command prompt as Administrator and navigate to the folder where the SAP Java Connector zip file is extracted.
4. Run deployManager.bat. The install menu appears in the console.
5 In the first screen, enter 1 and specify the root location of the Policy Controller. The default is C:\Program Files\NextLabs\Policy Controller.
6 Enter 8 to access the install menu. A screen appears for each of the configuration values you need to define.

![Figure 2-6: Installation Menu Options](image)

7 Using the numbered options 1 through 8 in the Installation Menu, enter the values for the Properties file. The instructions for retrieving these values are provided in Locate Required Values for Configuration.

Note: When you install the Java Connector using the Deployment Manager script, the RFC user password can be supplied in plain text form. You do not need to use the `mkpassword.bat` utility to encrypt the password (as you do for the manual process).

Uninstall deletes all the jars, dlls, and configured values in the SAPJavaSDK-service.properties file.

It is recommended that you make a backup of the SAPJavaSDKService.properties file.

8 When you have entered all the Properties values, select 9 to proceed with installation.

9 There is one installation file for the Java Connector that the script cannot install because it is specific to the host where the Java Connector is being installed. From the extracted contents of the .zip file, open the `xlib` folder. Open the folder that corresponds with the operating system of the host on which you are installing the Java Connector:

- "Windows 64-bit, Intel-Arch: Open the NTia64 folder
- "Windows 64-bit, AMD: open the NTamd64 folder
10 In the appropriate folder, select sapjco3.jar and sapjco3.dll. Copy the files to <install dir>/Policy Controller/jre/lib/ext.

**Manual Installation of the Java Connector for the Server Policy Controller**

If your implementation uses the Server Policy Controller, you can manually install the Java Connector for Policy Controller as described in this section.

**Before You Begin**
- The Policy Controller must be installed.
- Obtain the Java Connector for Policy Controller installation file from NextLabs. For more information, contact NextLabs Technical Support at support@nextlabs.com.
- To stop the Policy Controller, you must have the profile password defined in the Control Center Administrator interface.

**Procedure**

1 Stop the Policy Controller, if it is running.

2 Create the following folders in the Policy Controller install directory:

   <install dir>/Policy Controller/jservice/config
   <install dir>/Policy Controller/jservice/jar/sap
   <install dir>/Policy Controller/jre/lib/ext

3 Locate the Java Connector for Policy Controller installation file from NextLabs support (SAPJCo-EntitlementManager-<version number>.zip).

4 In the zip file, locate the SAPJavaSDKService.Properties file at SAPJCo-EntitlementManager-<version number>.zip/SAPJCo-EntitlementManager/config.

5 Copy the SAPJavaSDKService.Properties file to <install dir>/Policy Controller/jservice/config.

6 In the zip file, locate the SAPJco-EntitlementManager.jar file at SAPJCo-EntitlementManager-<version number>.zip/SAPJCo-EntitlementManager.

7 Copy the SAPJco-EntitlementManager.jar file to <install dir>/Policy Controller/jservice/jar/sap.

8 From the extracted contents of the .zip file, open the xlib folder that matches the host on which you are installing the Java Connector:
   - Windows 64-bit, Intel-Arch: open the NTia64 folder
   - Windows 64-bit, AMD: open the NTamd64 folder

9 In the appropriate folder, select sapjco3.jar and sapjco3.dll. Copy the files to <install dir>/Policy Controller/jre/lib/ext.

**Next Steps**

*Installing the Java Connector for EDRM*
Installing the Java Connector for EDRM

Use this procedure to install the Java Connector for NextLabs Enterprise Digital Rights Management.

Before you Begin

- Obtain the Java Connector installation file from NextLabs. For more information, contact NextLabs Technical Support at support@nextlabs.com.
- To stop the Policy Controller, you must have the profile password defined in the Control Center Administrator interface.
- Ensure that NextLabs Rights Management Server has been installed and configured correctly.
- Ensure the Rights Management Server Key Management settings are configured correctly.

Procedure

1. Stop the Policy Controller, if it is running.
2. Locate the EDRM Java Connector installation file (SAPJCo-EDRM-<version number>.zip) received from NextLabs support.
3. Extract the SAPJCo-EDRM zip file to a temporary location.
4. Copy the SAPJCo-EDRM.jar file to the following location: 
   <install_dir>/Policy Controller/jservice/jar/sap/
5. Copy the SAPJCo-EDRM.properties file to the following location: 
   <install_dir>/jservice/config/

Next Steps

Configuring the RFC Connection for the Policy Controller

After installing the Java Connector, the next step is to configure a Remote Function Call for the Policy Controller.

Procedure

1. In the SAP interface, enter transaction SM59. The RFC connection configuration screen appears.
2. Select the TCP/IP connections folder, then click Create.
Chapter 2: Installation and Set Up

Figure 2-7: Creating a TCP/IP RFC Connection

3 Set the Connection type to T.

4 Enter an RFC Destination name, for example, NEXTLABS_PC.
   
   Note: Make note of the RFC Destination name. You need to enter this name in the
   /NEXTLABS/CONCFG table in a subsequent step.

5 In the Technical Settings tab, select Registered Server Program.

6 Enter a Program ID, for example, NXL_CONNECT_TO_PC.
   
   Note: Make note of the Program ID. You need to enter this ID in a properties file in a
   subsequent procedure.
7 Click the MDMP & Unicode tab.

8 Select Unicode.
Figure 2-9: Selecting Unicode

9 Save the configuration.

Next Steps
Configuring the RFC Connection for EDRM

Configuring the RFC Connection for EDRM
Use this procedure to configure an RFC connection for the Protect and Classify features to work.

1 In the SAP interface, enter the SM59 transaction code. The Configuration of RFC Connections screen appears.

2 Select the TCP/IP connections folder and then click Create.
3 Set the Connection type to T.

4 Enter an RFC Destination name, for example, NEXTLABS_EDRM. Make note of the RFC Destination name, because you will need to enter the RFC destination name into the /NEXTLABS/CONCFG table in a subsequent step.

5 In the Technical Settings tab, select Registered Server Program.

6 Enter a Program ID, for example, NXL_CONNECT_TO_EDRM. Make note of the Program ID, as you will need it when you configure the properties file in a subsequent step.

7 Click the MDMP & Unicode tab.

8 Select Unicode.

9 Save the configuration.

Next Steps

Configuring the Properties and Services Files Manually
Configuring the Properties and Services Files Manually

Two additional files must be configured for the Java Connector: the SAPJavaSDKService.properties file and the Windows services file.

If you used the script method of installation, then the SAPJavaSDKService.properties file is already configured (the script provides a menu where you can enter the required values, which are written to the properties file automatically). However, you must still configure the Windows Services file manually.

Configuring the SAP JavaSDK Properties File (Manual installation option only)

If the Java Connector files are installed manually, the SAPJavaSDK properties file must be configured to communicate with the SAP Server and the Policy Controller.

Note: If you use the installation script, the properties file is configured automatically. Configuring the SAP JavaSDK properties file when using the script is necessary only if you want to change information after the initial installation, or add additional SAP Servers or Policy Controllers to your implementation. For more information on how to configure multiple Policy Controllers for load balancing, see Load Balancing the Policy Controller on page 148.

Procedure

1. Stop the Policy Controller, if it is running.
2. Navigate to and open the SAPJavaSDKService.properties file in the following location:
   
   <tomcat-home>/nextlabs/dpc/jservice/config

3. Verify that jar-path is set to the following:

   <install dir>/Policy Controller/jservice/jar/sap/SAPJCo-EntitlementManager.jar

4. Locate the section for server destination details prefix.

5. If necessary, add a new set of server and destination parameters to reflect the number of servers in your implementation.

For each server instance, you must append prefix values to the dest_name and server_name property in the following format: DEST<number>_; and SERV<number>_; where <number> refers to the server instance.

For example, below is the configuration for three server instances.

   #server destination details prefix
   dest_prefix=DEST1_;DEST2_;DEST3_; server_prefix=SERV1_;SERV2_;SERV3_;

   If our implementation included four server instances, we would add DEST4_; to dest_prefix and SERV4_; to server_prefix.
6 Locate the destination data provider connection details section. Enter the following information:

- `ashost` = FQDN for the host where SAP ECC is installed
- `sysnr` = System number
- `client` = client
- `user` = RFC user name
- `passwd` = RFC user password. This password must be encrypted using the `mkpassword.bat` utility installed with the Control Center. These instructions are provided in Locate Required Values for Configuration.

**Note:** For more information on how to use the `mkpassword.bat` utility, see Control Center Installation Guide.

- `lang` = language code (en for English)

7 Locate the server data provider connection details section. Enter the following information:

- `gwhost` = Gateway host name
- `gwserver` = Gateway service name
- `progid` = Program ID that was defined in the RFC configuration step (in our example, `NEXTLABS_CONNECT_TO_PC`)

8 After you have made all the changes, save and close the properties file.

**Configuring the SAP JavaSDK Properties File (Required for all installation types)**

Use this procedure to configure the SAP JavaSDK properties file.

**Procedure**

1 Stop the Policy Controller, if it is running.

2 Open the `SAPJavaSDKService.properties` file using a text editor.

3 Locate the line `server_prefix=SERV1_;` and add `SERVRMI_;` at the end of the line as shown in the following example:

   `server_prefix=SERV1_;SERVRMI_;`

4 Add the following code below the `server_prefix=SERV1_;SERVRMI_;` line:

   ```
   # START of EDRM Settings
   #destination data provider Connection details
   SERVRMI_jco.client.ashost=[!CLIENT_HOST!]
   SERVRMI_jco.client.sysnr=[!CLIENT_SYSNR!]
   SERVRMI_jco.client.client=[!CLIENT_ID!]
   SERVRMI_jco.client.user=[!CLIENT_USER!]
   SERVRMI_jco.client.passwd=[!CLIENT_PASSWD!]
   ```
SERVRMI_jco.client.lang=en
#comment below two lines if connection pool is not required
SERVRMI_jco.destination.peak_limit=3
SERVRMI_jco.destination.pool_capacity=3
#server data provider Connection details
SERVRMI_jco.server.gwhost=[!GATEWAY_HOST!]
SERVRMI_jco.server.gwserv=[!GATEWAY_SERV!]
SERVRMI_jco.server.progid=[!GATEWAY_PRGID!]
SERVRMI_jco.server.connection_count=02
# NOTE: To add more destination server settings; add SERVRMI1_ or
SERVRMI2_

etc to server_prefix and respective block of SERVRMI_ or SERVRMI
properties.

# END of SERVRMI Settings

5 Replace the following variables, including the brackets and the exclamation points, with
the appropriate values for your system:
- [!CLIENT_HOST!]: The hostname of your server, that is, the FQDN for the host where SAP
  ECC is installed.
- [!CLIENT_SYSNR!]: The system to use, that is, the system number.
- [!CLIENT_ID!]: The client ID to use. For example: 100
- [!CLIENT_USER!]: The user name of the account. For example: developer
- [!CLIENT_PASSWD!]: The password of the account. You must encrypt the password by
  using the mkpassword.bat utility that is available when you install NextLabs Control
  Center. For more information, see NextLabs Control Center Installation Guide.
- [!GATEWAY_HOST!]: The host name of the gateway server.
- [!GATEWAY_SERV!]: The gateway service name.
- [!GATEWAY_PRGID!]: The Program ID that was defined when Configuring the RFC
  Connection for EDRM. That is, NXL_CONNECT_TO_EDRM.

6 Save the SAPJavaSDKService.properties file.

Next Steps
Configuring the SAPJCo-EDRM Properties File

Configuring the SAPJCo-EDRM Properties File

Use this procedure to configure the SAPJCo-EDRM properties file.

Procedure

1 Stop the Policy Controller, if it is running.

2 Open the SAPJCo-EDRM.properties file using a text editor.
The following is an example of the SAPJCo-EDRM.properties file.

```
name = SAPJCo-RMAPI
jar-path = <install_dir>/Policy Controller/jservice/jar/sap/SAPJCo-EDRM.jar
friendly_name = SAP Java EDRM API
description = SAP Java EDRM API
category = EDRM SERVICE

#SAP module names
rmapi_handler=/NXLEDRM/IRM_API_AGENT

#server destination details prefix
server_prefix=SERVRMI_;
SERVRMI_jco.client.trace=1

#Key Management Service properties
KEYSTORE_NAME=<install_dir>/NextLabs/Policy Controller/jservice/jar/KeyManagement/rmskmc-keystore.jks
KEYSTORE_PASSWORD=[!KEYSTORE_PASSWORD!]
TRUSTSTORE_NAME=<install_dir>/NextLabs/Policy Controller/jservice/jar/KeyManagement/rmskmc-truststore.jks
TRUSTSTORE_PASSWORD=[!TRUSTSTORE_PASSWORD!]
PC_HOST_NAME=localhost
RMI_PORT_NUM=1499

#Delimiter
delimiter = |
```

3 Ensure that jar-path contains the correct location of the SAPJCo-EDRM.jar file. That is, the SAPJCo-EDRM.jar file must be pointing to the following location:

```
<install_dir>/Policy Controller/jservice/jar/sap/
```

4 Save the file and start the Policy Controller.
Next Steps

Configuring the Windows System Services File (Manual and Script Installation)

You must enter the SAP Gateway service and port number in the Windows Services file on the host where the Policy Controller is installed.

Procedure

1. Browse to `<Windows Home>\System 32\drivers\etc`.
2. Open the services file using Wordpad or Notepad.
3. Add a new entry in the file for the Gateway Service name (the `gwserv` parameter value) and port. For example:
   ```plaintext
   sapgw00 3300/tcp #sap connection for NextLabs
   ```
4. Save the file.

Next Steps

After configuring the Properties and Services files, the next step is to test the connection. See Testing the Java Connector Configuration.

There is an additional step to configure the Java Connector: you must define the `AGENT_COMMUNICATION_OPTION` and `AGENT_RFC_NAME` in `/NEXTLABS/CONCFG`. This step is discussed in Configuring SAP Data Handling and Connection Settings on page 70.

Testing the Java Connector Configuration

After the configuration is complete, the final step is to test the Java Connector configuration.

Procedure

1. In SAP ECC, enter transaction `SMGW`. The Gateway Monitor appears.
2. In the Go to menu, select Logged in Clients.
3 Check for the Program IDs that were defined for the RFC connections in SM59.

![Gateway Monitor for NQ7.qapf2.qalab01.nextlabs.com / Connections to CI](image)

*Figure 2-10: Testing the Java Connections*

**Testing the Connection from the Policy Controller Side**

You can test the connection from the Policy Controller side provided that you have performed the following tasks:

- Installed EDRM for SAP.
- In `/NEXTLABS/CONCFG` table (accessed through SM30), made the following configuration changes:
  - `AGENT_COMMUNICATION_OPTION` should be set to `RFC`.
  - `AGENT_RFC_NAME` should be set to the **RFC Destination Name** that was defined in SM59.
  - `EDRM_AGENT_RFC_NAME` should be set to the **RFC Destination Name** that was defined in SM59.

After you have performed these tasks and started the Policy Controller, you can test the connection from the Policy Controller side.

**Procedure**

1. Browse to the location of the Agent log at `<InstallDir>\PolicyController\Agentlog`.

2. Search the log for the following message, which indicates the starting of the SAPJavaSDK, used in the Communication interface.

   INFO: SAPJAVASDK init() started.
   INFO: SAPJavaRMAPI init() started.

   The installation is complete.

**Next steps**

The next step is to configure EDRM for SAP. See Configuring NextLabs Control Center for EDRM on page 43.
This section describes the configuration steps for NextLabs Enterprise Digital Rights Management for SAP.

Topics:

- Configuring NextLabs Control Center for EDRM
- Configuring NextLabs EDRM
- Using the NextLabs Configuration Tool
- Configuring Encryption Keys
- Configuring the Check Table for Classification Objects
- Configuring Mapping Table for Classification Values
- Configure Rights Management Client Installer Path
- Defining Implementations

### Configuring NextLabs Control Center for EDRM

This section explains how to configure NextLabs Control Center for EDRM. The configuration procedures in this section are required regardless of the features installed on your implementation.

- Enrolling Users from SAP into the Policy Server
- Configuring SAP Actions
- Configuring SAP Obligations

### Enrolling Users from SAP into the Policy Server

You can enroll SAP users into NextLabs Policy Server so that they can be added to User Components in Policy Studio. This is a two-stage process: first extract user data from SAP; then upload user data into Policy Server.

### Extracting User Data from SAP

You can download user data from SAP by scheduling a batch job to run the extract program: `/NEXTLABS/USER_EXTRACT`. 

---

3 Configuration
**Procedure**

1. In the SAP interface, enter transaction SM36. The *Define Background Jobs* screen appears. See Figure 3-1.

**Define Background Job**

![Define Background Job Screen](image)

Figure 3-1: Define Background Job Screen

2. Enter the Job name and click *Step* on the application tool bar. A pop-up window appears.

**Program Values Pop-up Window**

![Program Values Pop-up Window](image)

Figure 3-2: Program Values Pop-up Window
3 Click **ABAP program**, and enter the name of the ABAP program: `/NEXTLABS/USER_EXTRACT`.

![Figure 3-3: Enter the ABAP program name](image)

4 Click **Check** and **Save** at the bottom of the window, then, return to the **Define Background Job** screen by clicking **Back**.

5 Click **Start Condition** in the tool bar. A pop-up window appears, prompting you to schedule the job.

![Figure 3-4: Scheduling the background job](image)

6 Schedule the start Date and Time conditions, or click **Immediate** to start the job at once. You can also schedule the job to run repeatedly by clicking on the **Date/Time** button and specifying the frequency. Click **Check** and **Save**.

7 After the pop-up disappears, be sure to click **Save** to save the job definition in the original screen. This is very important. If the job definition is not saved in the **Define Background Job** screen, the job is not scheduled. After you save the job, a message appears at the bottom of the screen in the status bar displaying the job saved or released message.

8 To verify the job definition, enter transaction **SM37** and click **Execute**. Make sure the User name in the selection screen is the same as the one used to define the job. The jobs scheduled or released by this user are displayed as in **Figure 3-5**.

**Note:** The download path for the user extract file can be configured using the **FILE** transaction.
Uploading SAP User Data into Control Center

After you have extracted user information from SAP, you need to upload it into NextLabs Control Center.

Before You Begin

Obtain the Enrollment Adapter file from NextLabs. For more information, contact NextLabs Technical Support at support@nextlabs.com.

Procedure

1. Locate the Enrollment Adapter file from NextLabs Technical Support.

2. Extract the Enrollment Adapter zip file to \C:\Program Files\NextLabs\PolicyServer\tools\enrollment. This creates a folder named SAPEnrollmentAdapter, which contains the following files:
   - Mapping.properties (maps SAP export columns to the ldif template)
   - Sap_nextlabs_com.def (ldif enrollment definition file that maps ldif to the dictionary)
   - Sap2ldif.bat (script that calls the ldif converter)
   - SAPDataLdifConverter.jar (converter program)

3. Change the mapping properties enrollment domain in two places to reflect the SAP system and company, for example, “EC2.nextlabs.com.”
   - dn: cn=<UserID>,dc=sap,dc=domain_name,dc=com
   - userPrincipalName: <<UserID>>@sap.domain_name.com

4. Set two file paths in Sap2ldif.bat. The first points to the folder on the SAP ECC server where the user extract is being placed. You should share this folder on the ECC system so that the enrollment process can reach it. The second is the output location of the ldif file.
   - “C:\Program Files\NextLabs\PolicyServer\java\bin\java” -jar SAPDataLdifConverter.jar mapping.properties “\\DEMO20-SAP02\src\userdata.txt"
   - “C:\Users\user.name\Documents\sapldif\sap_nextlabs_com.ldif”

5. In Sap_nextlabs_com.def, change the location of the ldif filename to
   “C:\Users\user.name\Documents\sap ldif\sap_nextlabs_com.ldif” (This is the same path as the output file.)
To run the conversion, run sap2ldif.

Set up enrollment for the ldif using the enroll and sync commands of the enrollmgr utility.

```
enrollmgr -u Administrator -enroll -t LDIF -n sap.domain_name.com -d "C:\Program Files\NextLabs\PolicyServer\tools\enrollment\SAPEnrollmentAdapter\sap_nextlabs_com.def"

enrollmgr -u Administrator -sync -t LDIF -n sap.domain_name.com -d "C:\Program Files\NextLabs\PolicyServer\tools\enrollment\SAPEnrollmentAdapter\sap_nextlabs_com.def"
```

Next steps

Configuring SAP Actions

Before you can write SAP policies, you must add SAP actions to Policy Studio. Actions include SAP Copy-From, Execute Transaction, Check-in, and others.

Procedure

1. Use Notepad or WordPad to open one of the following XML files supplied by NextLabs Technical Support:
   - Open SAP Actions_FullBuild.xml. This file contains all the SAP actions.
   - Open SAP Actions_Upgrade.xml. This file contains only the new SAP actions added in the current release.
2. Open the main configuration file, configuration.xml, on the Control Center host. By default, this file is located in
   ```
   <installDirectory>\PolicyServer\server\configuration
   ```
3. Copy the actions from SAP Actions_FullBuild.xml or SAP Actions_Upgrade.xml.
4. Paste the actions into configuration.xml in the <ActionList> </ActionList> section.

The following are examples of actions in the <ActionList> section of configuration.xml.

```xml
<ActionList>
  <!-- [EDRM Action] -->
  <Action>
    <Name>UPLOAD</Name>
    <DisplayName>Upload</DisplayName>
    <ShortName>UP</ShortName>
    <Category>Transform</Category>
  </Action>
</ActionList>
```
<Name>GETLOCATION</Name>
<DisplayName>GetLocation</DisplayName>
<ShortName>GL</ShortName>
<Category>Transform</Category>
</Action>

<Action>
<Name>DOWNLOAD</Name>
<DisplayName>Download</DisplayName>
<ShortName>DL</ShortName>
<Category>Transform</Category>
</Action>

<Action>
<Name>SAVE</Name>
<DisplayName>SAVE</DisplayName>
<ShortName>SV</ShortName>
<Category>Access</Category>
</Action>

<!-- RMC Actions -->
<Action>
<Name>RIGHT_VIEW</Name>
<DisplayName>Right View</DisplayName>
<ShortName>R0</ShortName>
<Category>Access</Category>
</Action>

<Action>
<Name>RIGHT_EDIT</Name>
<DisplayName>Right Edit</DisplayName>
<ShortName>R1</ShortName>
<Category>Access</Category>
</Action>

<Action>
<Name>RIGHT_PRINT</Name>
<DisplayName>Right Print</DisplayName>
<ShortName>R2</ShortName>
<Category>Access</Category>
</Action>

<Action>
<Name>RIGHT_CLIPBOARD</Name>
<DisplayName>Right ClipBoard</DisplayName>
<ShortName>R3</ShortName>
<Category>Access</Category>
</Action>

<Action>
<Name>RIGHT_SAVEAS</Name>
<DisplayName>Right SaveAs</DisplayName>
<ShortName>R4</ShortName>
<Category>Access</Category>
5 Save the changes and restart Control Center.

6 After the system restarts, test the configuration by opening Policy Studio and searching for the new actions in the Actions component pane.

Next steps

The next step is Configuring SAP Obligations on page 49 for Policy Studio.

Configuring SAP Obligations

You also must insert obligations for Enterprise Digital Rights Management for SAP in the Control Center configuration.xml file. The following obligations may be required, depending on your configuration:

- **SAP User Alert**: Display messages defined in Policy Studio to users in SAP
- **SAP Message Class Display**: Display messages from SAP Message classes as part of a NextLabs policy notification
- **Maintain Classification from Tags**: Run Read Tags upon document upload
- **Temporary File Location for Processing Originals**: Define a temporary location to be used by EDRM for processing originals
- **EDRM Rights Management**: Classify, protect, or protect and classify documents
• EDRM Remove Protection: Provide the user with an unprotected copy of the document

  Note: Another obligation pertinent to Rights Management Server may be required to implement Integrated Rights Management policies.

Procedure

1. Use a text editor to open one of the following XML files, supplied by NextLabs Professional Services:
   - Open SAP Obligations_FullBuild.xml. This file contains all the SAP obligations.
   - Open SAP Obligations_Upgrade.xml. This file contains only the new SAP obligations added in the current release.

2. Open the main configuration file, configuration.xml, on the Control Center host. This file is located at:

   `<installDirectory>\NextLabs\PolicyServer\server\configuration`

3. Copy the obligations you require from SAP Obligations_FullBuild.xml or SAP Obligations_Upgrade.xml.

4. Paste the obligations into configuration.xml file in the `<Obligations>` section.

The following are examples of obligations:

```
<Obligations>
<!-- [SAP USER MESSAGE] -->
<Obligation>
<DisplayName>SAP User Message</DisplayName>
<RunAt>PEP</RunAt>
<Name>SAPMSG</Name>
<Arguments>
  <Argument>
    <Name>Language Code</Name>
    <Value>EN:English</Value>
    <Value>DE:German</Value>
    <Value>ZH:Chinese</Value>
    <Value>ZF:Chinese trad.</Value>
    <Value>SR:Serbian</Value>
    <Value JA:Japanese</Value>
    <Value>KO:Korean</Value>
    <Value>TH:Thai</Value>
    <Value>RO:Romanian</Value>
    <Value>SL:Slovenian</Value>
    <Value>HR:Croatian</Value>
```
<Obligation>
<DisplayName>SAP–EDRM–Maintain Classification from Tags</DisplayName>
<RunAt>PEP</RunAt>
<Name>SAPMSG</Name>
<Arguments>
<Argument>
<Name>Read Tags</Name>
<Value>Read Tags from Original</Value>
</Argument>
<Argument usereditable="false" hidden="true">
<Name>logid</Name>
<Value default="true">$CELogUid</Value>
</Argument>
</Arguments>
</Obligation>

<Obligation>
<DisplayName>SAP–EDRM–Temporary File Location for Processing Originals</DisplayName>
<RunAt>PEP</RunAt>
<Name>SAPMSG</Name>
<Arguments>
<Argument>
<Name>Storage Category</Name>
<Value>DMS_C1_ST</Value>
<Value>DMS_H1_ST</Value>
<Value>DVS1CONT</Value>
<Value>ZDMS_CNI</Value>
<Value>ZDMS_UK1</Value>
<Value>ZDMS_US1</Value>
<Value>ZIND_IND</Value>
<Value>ZUS_US_DB</Value>
<Value>ZUK_UK_DB</Value>
</Argument>
<Argument>
<Name>File Location</Name>
</Argument>
<Argument usereditable="false" hidden="true">
</Argument>
</Arguments>
</Obligation>
<?xml version="1.0" encoding="utf-8"?>

<Name>logid</Name>
<Value default="true">$CELogUid</Value>
</Argument>
</Arguments>
</Obligation>

<Obligation>
<DisplayName>SAP–EDRM Rights Management</DisplayName>
<RunAt>PEP</RunAt>
>Name>SAPMSG</Name>
<Arguments>
<Argument usereditable="false">
<Name>Protect</Name>
<Value>Yes</Value>
<Value>No</Value>
</Argument>
<Argument usereditable="false">
<Name>Classify</Name>
<Value>User-Defined</Value>
<Value>All-Column</Value>
<Value>Specific-Column</Value>
<Value>No-Classification</Value>
</Argument>
<Argument usereditable="true">
<Name>Field</Name>
<Value>EXPORT SECURITY</Value>
<Value>EXPORT LICENSE</Value>
<Value>COMPANY SECURITY</Value>
<Value>ECCN</Value>
<Value>MUNITIONS</Value>
<Value>IP SECURITY</Value>
</Argument>
<Argument usereditable="true">
<Name>Value</Name>
</Argument>
<Argument usereditable="false">
<Name>Mode</Name>
</Argument>
<Value default="true">Append</Value>  
<Value default="false">Overwrite</Value>  
</Argument>  
<Argument usereditable="false" hidden="true">  
{Name>logid</Name>  
<Value default="true">$CELogUid</Value>  
</Argument>  
</Arguments>  
</Obligation>  

<Obligation>  
<DisplayName>SAP-EDRM Remove Protection</DisplayName>  
<RunAt>PEP</RunAt>  
<Name>SAPMSG</Name>  
<Arguments>  
<Argument usereditable="false">  
{Name>Remove Protection</Name>  
<Value>Yes</Value>  
<Value>No</Value>  
</Argument>  
<Argument usereditable="false" hidden="true">  
{Name>logid</Name>  
<Value default="true">$CELogUid</Value>  
</Argument>  
</Arguments>  
</Obligation>  

<!-- [SAP MESSAGE CLASS] -->  
<Obligation>  
<DisplayName>SAP Message Class Display</DisplayName>  
<RunAt>PEP</RunAt>  
<Name>SAPMCL</Name>  
<Arguments>  
<Argument>  
{Name>Message Class ID</Name>  
</Argument>  
<Argument>  
{Name>Message Number</Name>  
</Argument>  
</Arguments>  
</Obligation>
5 Save your changes and restart Control Center.

6 After the system restarts, test the configuration by opening Policy Studio and creating a new policy. The new obligations should appear in the drop-down list of Custom Obligations.

This completes the configuration of NextLabs Control Center to support EDRM for SAP.

**Next steps**

*Configuring NextLabs EDRM*

**Configuring NextLabs EDRM**

Use the procedures in this section to configure NextLabs EDRM.

- Activating the Entitlement Pack
- Adding Composite Keys and Classification Values
  - Security Identifiers
  - Composite Keys
  - Classification Values
- Linking Composite Keys (SECENH)
- Mapping Security Fields (SECMPG)
- Configuring Security Identifier/Composite Key Value Tables (EPVAL) on page 68
- Configuring SAP Data Handling and Connection Settings on page 70
Chapter 3: Configuration

- Setting Default Values Automatically
- Changing Connection Configuration Settings
- Recommended Configuration for Implementations with Many Classifications (more than 40,000 rows)
- Configuring Number Range Intervals
- Configuring the NextLabs Number Range
- Configuring Policy Checks Based on Transaction/UI Function
- Configuring Special Fields for the Security Classification Maintenance Table
- Assign Security Identifiers to Transactions

Using the NextLabs Configuration Tool

Use the NextLabs configuration tool to configure EDRM using SPRO (SAP Project Reference Object).

Note: The configuration tool covers only the configuration procedures performed in SAP. You must already have installed the software components as described in Installation and Set Up on page 17, and performed the configuration procedures in the NextLabs Control Center as described in Configuring NextLabs Control Center for EDRM on page 43.

Procedure

1. In the SAP interface, do one of the following:

   - Enter transaction /NEXTLABS/CONFIG. This transaction shows the installed NextLabs components only.
   - Enter transaction SPRO, then do the following:
     a. Click SAP Reference IMG, then expand the Cross-Application Components node.
     b. Expand NextLabs: Entitlement (Manager/Packs) Configuration. As shown in Figure 3-6, the configuration tool organizes procedures by Entitlement Manager and Entitlement Packs. You see only the Entitlement Packs that you have installed.
Using the NextLabs Configuration Tool

3 Configure the NextLabs Entitlement Manager, which is the base product.

4 Configure each Entitlement Pack that you want to implement. See Configuring NextLabs EDRM on page 55.

5 For more information about procedures, click the Documentation icon shown in Table 3-1.

Checking Configuration Status

As you finish configuring the Entitlement Manager and each Entitlement Pack, you can run a utility to check if each component has been configured correctly. To run this utility, either run transaction /NEXTLABS/UTIL_CNFG, or execute the Utility Program to Check Configuration activity, as shown in Figure 3-7.
The utility displays a report showing the status of each configuration activity, as shown in Figure 3-8. The Activity Message column provides a description of the status. The Activity Result column provides a graphical status. A green symbol indicates that the configuration meets the minimum requirements, a yellow symbol is advisory (for example, an optional item is inactive), and a red symbol indicates an error that needs to be fixed.

The utility's primary function is to report which configuration activities have been completed and which have not. Even if all activities are complete and have the green or yellow status, this does not necessarily mean that the configuration is appropriate for your system or that policies will run as intended. The utility is comparable to a programming tool that can detect syntax errors, but not errors in logic.
Activating the Entitlement Pack

The first configuration step is to activate the Entitlement Pack for the SAP back-end systems in the `/NEXTLABS/EPCONF` table.

Procedure

1. In the SAP interface, enter transaction `SM30`. The Maintain Table Views screen appears.
2 In Table/View, enter /NXLABS/EPCONFIG to access the NextLabs Entitlement Pack Configuration screen.

3 Click Maintain.

4 Enter the NextLabs Entitlement Pack you are installing. For EDRM, enter NXLEDRM.

5 Select Flag for Activation of EP.

6 Enter the Identifier Append Structure used for the Entitlement Pack. For EDRM, enter /NXLABS/SECIDT_ECC.

7 Click Save.

Next steps
Linking Composite Keys (SECENH) on page 65 for Entitlement Pack for ECC.

Adding Composite Keys and Classification Values

The first step of configuring data for the Security Classification Maintenance table is identifying the data to be displayed. The following are the types of data that appear in the Security Classification Maintenance table:

- Security Identifiers
- Composite Keys
- Classification Values

Security Identifiers

A Security Identifier is the primary business object that classifications are applied to. Several Security Identifiers are configured in the Security Classification Maintenance screen upon installation, including Material and Document. Custom Security Identifiers can also be configured. However, this addition is only relevant if you are adding custom enhancements for new transactions.

Composite Keys

The Security Classification Maintenance table also contains Composite Keys. Composite keys enable you to create a unique version of a Security Identifier that can be classified independently. For example, if the Identifier is Document and a Composite Key is Document Version, you can maintain a unique classification for the combination of Document and Document Version. A Security Identifier can have any number of Composite Keys associated with it.

Classification Values

The Security Classification Maintenance table also contains Classification Values. These are classifications customers want to associate with business objects. Some common examples are
export control, license, and IP control category. Classification values should be configured to reflect requirements addressed in a customer implementation.

Security Identifiers Which fields should display as classification values?

![Figure 3-9: Customizing the SAP Security Classification Screen](image)

Adding Composite Keys

Composite Keys are not required, but are commonly used in SAP systems to track different states of business objects. For example, the Composite Keys for Document are included in the Nextlabs installation because they are so common; these include DOKTL, DOKAR, and DOKVR. You can add composite keys as needed.

**Note:** To prevent custom enhancements from being overwritten by NextLabs upgrades, create Append Structures in your customer namespace. Do not you add Append Structures to the NextLabs namespace.

**Procedure**

1. In the SAP interface, enter transaction SE11.
2. Select **Data Type**, and in the field, enter the structure name `/NEXTLABS/SECENH_CLS`, then click **Display**. The pre-configured composite keys display.
3. Click **Append Structure** on the toolbar.
If the message **No append defined** appears, click **OK**.

5 Click **Create Append**, enter an Append Structure name, then click **Continue**.

6 In the **Append Structure** screen, enter information for the Component you want to add. In our example, we add a **WERKS** Component for Plant.

**Note:** To add additional Composite Keys, you can add them to the same Append Structure.
Adding a Composite Key as a New Component

7 Save and Activate the new component.

8 Save and Activate the new Append Structure.

Adding New Classification Values

Custom Classification values can be added to the Security Classification Maintenance table.

Note: To prevent custom enhancements from being overwritten by NextLabs upgrades, create Append Structures in your customer namespace. Do not add them to the NextLabs namespace.

Procedure

1 In the SAP interface, enter transaction SE11.

2 Select Data Type, and in the field, enter the structure name /NEXTLABS/CLS_APPEND. The default Classification values that default on installation display.
3 Click **Append Structure** on the toolbar.

![Append Structure](image)

**Figure 3-12: Append Structure**

4 If the message **No append defined** appears, click **OK**.

5 Click **Create Append**, enter an Append Structure name, then click **OK**.

6 In the **Append Structure** screen, enter information for a new Component(s).
   a Enter the Component name.
   
   b If you want to select a predefined Component type, enter it in the **Component type** screen (the characteristics of the Component type default in the Data Type, Length, and other fields).
   
   c If you do not want to select a pre-defined Component type, define the component characteristics:
      - Enter the component Data Type (for example, CHAR).
      - Enter the Length of the field.
      - Enter a Short Description of the component.
4 **Save and Activate** New Component.

5 **Save and Activate** the new Append Structure.

**Next steps**

The next step for Composite Keys is to associate them with a Security Identifier (see **Linking Composite Keys (SECENH)** on page 65). This step is not necessary for Classification Values. If you are configuring Classification Values only, the next step is **Mapping Security Fields (SECMPG)** on page 66.

---

**Linking Composite Keys (SECENH)**

Composite keys (such as Document Version and Part) must be configured in the Security Identifier Composite Key window so that they are displayed in the Security Classification Maintenance window and are available for use in policies. After adding Composite Keys, you must link them to a Security Identifier. This is done in SECENH.

**Procedure**

1 In the SAP interface, enter transaction SM30. The Maintain Table Views screen appears.

2 In **Table/View**, enter /NEXTLABS/SECENH. Click **Display**. The *Maintain Security Identifier Composite Key* screen appears.

3 Change from Display to Change view.

4 Click **New Entries**.

5 For the existing Security Identifier **DOCNUM**, enter the following three Key Fields (as shown in Figure 3-13):
   - **DOKAR**: For Document Type
   - **DOKTL**: For Document Part
   - **DOKVR**: For Document Version

6 Associate each new Composite Key configured in the previous step with a Security Identifier. In Figure 3-13, the Security Identifier for Material (MATNR) is linked with a Composite Key for Plant (WERKS).

**Note:** You can use the Search Help to see the list of valid Security Identifiers and Key Field (Composite Key) Names.
Figure 3-13: Linking Composite Keys to Security Identifiers

7 Save the changes.

Next steps
Mapping Security Fields (SECMPG)

Mapping Security Fields (SECMPG)

For classification fields configured in the Security Classification Maintenance table, you must perform the step of mapping values to the Policy Controller. This procedure is required for fields to be passed to the Policy Controller.

Procedure

1 In the SAP interface, enter transaction SM30. The Maintain Table Views screen appears.

2 In Table/View, enter /NEXTLABS/SECMPG.
Using the NextLabs Configuration Tool

**Display View "NextLabs: Security Fields Mapping": Overview**

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Property Name</th>
<th>Cardinality</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>FCCN</td>
<td>FCCN</td>
<td>Single</td>
<td>Classification Data</td>
</tr>
<tr>
<td>EXPORTLIC</td>
<td>EXPORT LICENSE</td>
<td>Single</td>
<td>Classification Data</td>
</tr>
<tr>
<td>EXPSECURITY</td>
<td>EXPORT SECURITY</td>
<td>Single</td>
<td>Classification Data</td>
</tr>
<tr>
<td>IP_SECURITY</td>
<td>IP SECURITY</td>
<td>Multiple</td>
<td>Classification Data</td>
</tr>
<tr>
<td>TCODE</td>
<td>TRANSACTION</td>
<td>Single</td>
<td>Transaction Data</td>
</tr>
<tr>
<td>USML</td>
<td>NATIONS</td>
<td>Single</td>
<td>Classification Data</td>
</tr>
</tbody>
</table>

**Figure 3-14: NextLabs Security Fields Mapping**

3 For each classification field to configure:

a Enter the **Field Name** that was added to the Append Structure for Classification Values.

b Enter the **Property Name**. This is the name of the value that should be entered in Policy Studio when the classification value is added to policy in a Resource component.

c Select **Multiple** or **Single** Cardinality.

d Enter the source of the classification data in the Source column:

- If the value should be passed to the Policy Controller from the Security Classification table, select **Classification Data**.
- If the value should be passed from a transaction, select **Transaction Data**.

5 Save the changes.

**Next steps**

Configuring Security Identifier/Composite Key Value Tables (EPVAL)
Configuring Security Identifier/Composite Key Value Tables (EPVAL)

For each composite key, default security identifier, and custom security identifier in your implementation, designate the master data table and field from which data should be drawn, and select options for restricting the data and records to be validated.

This procedure must be performed for each SAP back-end system.

Procedure

1. In the SAP interface, enter transaction SM30. The Maintain Table Views screen appears.

2. In Table/View, enter /NEXTLABS/EPVAL to access the Security Identifier/Composite Key Value screen.

3. Click New Entries. For each entry, specify the following information:
   - In Security Identifier/Composite Key, enter the security identifier or composite key added as part of your implementation.
   - In Value Table, enter the master data table to use to supply values for the security identifier or composite key.
   - In Field Name, enter the name of the field in the master data table that supplies the values for the security identifier (skip this step if you do not require a value table and select Inactive for Validation).
   - In Validation Attributes, select one of the following options:
     - Active for Validation: Select this option to validate field entries against a designated value table. If this option is selected, the field can come from any value table maintained in SAP. During validation, the table is checked to validate whether the user is entering a valid value in the Security Classification Maintenance table.
     - Combine Validation with other fields: Select this option to require a Security Identifier and/or its Composite key(s) to come from the same value table. If this option is selected, the field must come from the same record in the value table as other fields that both (1) come from the same table and (2) have Combined Validation with other fields selected. In other words, you can have one Security Identifier (DOCNUM) and two composite keys (DOKAR and DOKVR) mapped to the same table (DRAW). If Combined Validation with other fields is selected for DOCNUM, DOKAR, and DOKVR, a validation check ensures that the values come from the same record in the same table. In this case, the Security Identifier and Composite Keys are always bound as a data set.
     - Inactive for Validation: Select this option if you do not want to require validation against a value table in SAP. In this case, the values do not need to come from a table maintained within the SAP system. Any data can be entered in the Security Classification Maintenance table by a user.

4. Click Save.
Table 3-1 lists the security identifiers and composite keys that are required for EDRM. The required identifiers must be configured for each installed Entitlement Pack.

**Table 3-1: Security identifiers, value tables and fields**

<table>
<thead>
<tr>
<th>Identifier/Composite Key</th>
<th>Value Table</th>
<th>Field</th>
<th>Validation Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>DOCNUM</td>
<td>DRAW</td>
<td>DOKNR</td>
<td>Combine validation with other fields</td>
</tr>
<tr>
<td>DOKAR</td>
<td>DRAW</td>
<td>DOKAR</td>
<td>Combine validation with other fields</td>
</tr>
<tr>
<td>DOKTL</td>
<td>DRAW</td>
<td>DOKTL</td>
<td>Combine validation with other fields</td>
</tr>
<tr>
<td>DOKVR</td>
<td>DRAW</td>
<td>DOKVR</td>
<td>Combine validation with other fields</td>
</tr>
<tr>
<td>NXL_OBJECT</td>
<td>/NEXTLABS/OBJECT</td>
<td>NXL_OBJECT</td>
<td>Active for validation</td>
</tr>
</tbody>
</table>

**Next steps**

**Mapping Transaction Codes and UI Functions to Actions (ACTIONS)**

By default, all SAP actions are mapped to the Run action in Policy Studio. If required, you can change the mapping based on your business requirements.

*Note:* For more information on defining the Copy from SAP action, see Configuring SAP Actions on page 47.

**Procedure**

1. In the SAP interface, enter transaction **SM30**. The Maintain Table Views screen appears.

2. In **Table/View**, enter **/NEXTLABS/ACTION**, then Click **Display**. All the entered action mappings display.

3. Click the toggle button to change from Display to Change view.

4. Click **New Entries** to add a new Action mapping.

5. Enter a valid UI Function or Transaction Code, or click the Search Help icon on the right side of the field for the Search Help, where you can search for valid UI Functions or Tcodes, and select one from the list.

6. Enter the exact name of the action to which you are mapping the Transaction Code or UI Function.

*Note:* There is no Search Help option for this field. Be sure to enter the Action name exactly as it appears in Policy Studio.
You can map any valid Transaction and UI Functions to any valid policy action to match your business processes and needs.

7 When you have finished entering new Policy Action mappings, click Save.

Next steps

Configuring SAP Data Handling and Connection Settings

Configuring SAP Data Handling and Connection Settings

The Default Configuration Maintenance table (/NEXTLABS/CONCFG) enables you to define numerous data handling and connection settings. This section describes each of these settings, including what values are available. This section also describes how to use a configuration program to make this process more efficient.

Setting Default Values Automatically

Rather than entering values for the settings in the Default Configuration Maintenance table manually, you can use the /NEXTLABS/CONCFG_MAINTAIN program to automatically populate the table with default settings. After you enter these default settings, you can change them as needed.

Procedure

1 In the SAP interface, enter transaction SE38 or SA38. The Program Execution screen appears.

2 Enter /NEXTLABS/CONCFG_MAINTAIN then click Execute.
When prompted to select whether you want to both create the activity entries and Update Entries with Default settings (you can change the settings manually afterward), select or deselect this option, and click Execute.
4 Optional: To change the settings later, go to the /NEXTLABS/CONCFG table. See Changing Connection Configuration Settings on page 72.

Changing Connection Configuration Settings

Use this procedure to change the default settings in the /NEXTLABS/CONCFG table, based on your business requirements.

Procedure

1 In the SAP interface, enter transaction SM30. The Maintain Table Views screen appears.

2 In Table/View, enter /NEXTLABS/CONCFG. Click Display. The Activity Maintenance Table appears.

3 Toggle the table from Display to Change view.

4 Change the default settings, based on your business requirements. Table 3-2 describes each setting and lists the valid options.

Table 3-2: Connection Configuration Settings

<table>
<thead>
<tr>
<th>Activity Name</th>
<th>Description</th>
<th>Valid Settings</th>
</tr>
</thead>
<tbody>
<tr>
<td>AGENT_COMMUNICATION_OPTION</td>
<td>Method for Policy Controller Communication Interface</td>
<td>RFC</td>
</tr>
<tr>
<td>AGENT_DEFAULT_DENY_OBLIGATION</td>
<td>If a policy definition does not include a default Deny message, you can define whether the default message that appears should come from a default NextLabs user alert (SAPMSG) or a default SAP Message Class (SAPMCL). The default SAPMSG is hard-coded and cannot be changed. The default SAPMCL message class (/NEXTLABS/CA000) can be modified.</td>
<td>SAPMSG or SAPMCL</td>
</tr>
<tr>
<td>AGENT_DEFAULT_IP_INCASE_OF_ERR</td>
<td>If your SAP implementation includes users for which there may not be an available IP addresses (for example, a service account user for running a background job), you can define a default IP address here that will be supplied for the user accounts.</td>
<td>IP address</td>
</tr>
<tr>
<td>AGENT_LOGICAL_PORT_NAME</td>
<td>Deprecated</td>
<td>Deprecated</td>
</tr>
<tr>
<td>AGENT_RFC_NAME</td>
<td>Remote Function Call Destination name</td>
<td>The RFC Destination name created using the SM59 transaction (see Configuring the RFC Connection for the Policy Controller).</td>
</tr>
<tr>
<td>EDRM_AGENT_RFC_NAME</td>
<td>The RFC destination name.</td>
<td>The RFC Destination name created using the SM59 transaction (see Configuring the RFC Connection for EDRM). For example, NEXTLABS_EDRM.</td>
</tr>
<tr>
<td>EDRM_MAX_RECORD_PER_REQUEST</td>
<td>The number of records per request.</td>
<td>The recommended value is 5.</td>
</tr>
<tr>
<td>BYPASS_TCODE_WHITELIST_CHECK</td>
<td>For internal configuration only. Do not change the default value without consulting NextLabs Support.</td>
<td>Blank: Disabled (Default)</td>
</tr>
</tbody>
</table>
### Table 3-2: Connection Configuration Settings (Continued)

<table>
<thead>
<tr>
<th>Activity Name</th>
<th>Description</th>
<th>Valid Settings</th>
</tr>
</thead>
</table>
| IRM_POLICY_CHECK       | This activity is deprecated. This setting enables a policy check to be performed whenever users execute /NEXTLABS/IRM programs for ECC. The benefit of the policy check would be to prevent unauthorized classification and/or encryption of originals. (This setting does not automatically create the policy. The policy would need to be defined in Policy Studio.) | (Deprecated)  
Yes: Enabled (Default)  
Space: Disabled                                                                                                                                  |
| MULTI_QUERY_TIMEOUT_IN_MILLISC | The SAP Agent collects transaction information from multiple policy queries and routes it to the Policy Controller. In the event that communication fails, because the Policy Controller is down or for some other reason, SAP needs to be configured for when and how it should respond. Each multi-query request sent to the Policy Controller contains groups of objects. A timeout occurs when any one group exceeds the timeout value you set. | Activity Type: value in milliseconds, must be greater than 0. The default is 40000.  
Activity Handler: in the event of a timeout, SAP takes the action that is specified in the Activity Handler for TIME_PERIOD_IN_MILLISECOND. |
| PBSC_POLICY_CHECK      | This activity is deprecated. This setting is used by other NextLabs products and any change to the setting is ignored. This setting enables a policy check to be performed whenever users execute /NEXTLABS/PBSC programs for ECC. The benefit of the policy check would be to prevent unauthorized classification of business objects. (This setting does not automatically create the policy. You must define the policy in Policy Studio.) | (Deprecated)  
Yes: Enabled (Default)  
Space: Disabled                                                                                                                                  |
| PBSC_UPLOAD_WAIT_TIME  | This activity is deprecated. This setting is used by other NextLabs products and any change to the setting is ignored. This setting enables a policy check to be performed whenever users execute the /NEXTLABS/SEC_CLS transaction. The benefit of the policy check would be to prevent unauthorized classification of business objects. (This setting does not automatically create the policy. You must define the policy in Policy Studio.) | (Deprecated)  
Numeric Values in seconds (1 to 180 seconds)  
Default is 60                                                                                                                                      |
| SEC_CLS_POLICY_CHECK   | This setting enables a policy check to be performed whenever users execute the /NEXTLABS/SEC_CLS transaction. The benefit of the policy check would be to prevent unauthorized classification of business objects. (This setting does not automatically create the policy. You must define the policy in Policy Studio.) | Yes: Enabled (Default)  
Space: Disabled                                                                                                                                  |
| SEC_CLS_SCREEN_CHANGE_HISTORY | This setting determines whether or not Change History appears by default in the NextLabs Security Classification Selection screen. This setting is the default. Users can change it if needed. | X: Display Change History as a selection option.  
Blank: Do not display change history as a selection option.                                                                                           |
| SEC_CLS_SCREEN_CLASSIFICATION | This setting determines whether or not classification fields display by default in the NextLabs Security Classification Selection screen. This setting is the default. Users can change it if needed. | X: Display classification fields as a selection option.  
Blank: Do not display classification fields as a selection option.                                                                                   |
### Table 3-2: Connection Configuration Settings (Continued)

<table>
<thead>
<tr>
<th>Activity Name</th>
<th>Description</th>
<th>Valid Settings</th>
</tr>
</thead>
<tbody>
<tr>
<td>SEC_CLS_SCREEN_COMPOSITE_KEY</td>
<td>This setting determines whether or not Composite keys display by default in the NextLabs Security Classification Selection screen. This setting is the default. Users can change it if needed.</td>
<td>X: Display Composite keys as a selection option. Blank: Do not display Composite keys as a selection option.</td>
</tr>
<tr>
<td>SEC_CLS_SCREEN_IDENTIFIER</td>
<td>This setting determines whether or not Identifiers display by default in the NextLabs Security Classification Selection screen. This setting is the default. Users can change it if needed.</td>
<td>X: Display Identifier as a selection option. Blank: Do not display Identifier as a selection option.</td>
</tr>
<tr>
<td>SEC_CLS_SCREEN_WARNING_MESSAGE</td>
<td>When more than a configured number of records are fetched, based on criteria entered in the NextLabs Security Classification Selection screen, a warning appears. Users are asked whether or not they wish to continue. This setting determines the number of records that triggers this warning. If a large warning number is supplied here, it can impact the amount of time users have to wait for fetched records. Also, this warning message does not prevent users from entering large numbers, but it does enable them to cancel the request.</td>
<td>Integer for the number of records before a warning is displayed (default 200).</td>
</tr>
<tr>
<td>SEC_CLS_UPLOAD</td>
<td>This setting determines the behavior of importing security classification information (from an Excel or .csv file) into the Security Classification Maintenance table. The options are whether you want to update new values only, or overwrite (meaning, delete all values and write new ones to the table).</td>
<td>Overwrite: overwrite values (default). Blank: update only</td>
</tr>
<tr>
<td>SEC_CLS_UPLOAD_HAS_HEADER</td>
<td>This setting determines behavior for importing security classification data (from an Excel or .csv file) into the Security Classification Maintenance table. If your source files have a header you can configure the update process to ignore the header when the values are imported.</td>
<td>X: Ignore first row (it is a header). Blank: Do not ignore first row (it is not a header).</td>
</tr>
<tr>
<td>SEC_CLS_UPLOAD_NO_OF_COLS</td>
<td>This setting determines behavior for importing security classification data (from an Excel or .csv file) into the Security Classification Maintenance table. You can set the number of columns to be imported from the source file. If the number you set is less than the number of columns in a source file, the excess columns are not imported.</td>
<td>Numeric Values (1 to 30) Default is 20</td>
</tr>
<tr>
<td>SEC_CLS_UPLOAD_NO_OF_ROWS</td>
<td>This setting determines behavior for importing security classification data (from an Excel or .csv file) into the Security Classification Maintenance table. You can set the number of rows to be imported from the source file.</td>
<td>Numeric values (1 to 64000) Default is 50000.</td>
</tr>
</tbody>
</table>
5 Verify that your changes are associated with the appropriate transport request by following these steps:

a Select **Menu > Table View > Transport**. A pop-up requests that you specify whether you want to create a new request or use an existing configuration request.

b Click the **green check mark**.

c Click the **Select all icon** (or F7).

d Click **Include in Request**.

---

### Table 3-2: Connection Configuration Settings (Continued)

<table>
<thead>
<tr>
<th>Activity Name</th>
<th>Description</th>
<th>Valid Settings</th>
</tr>
</thead>
<tbody>
<tr>
<td>SEC_CLS_WILDCARD_COMPOSITE_KEY</td>
<td>This setting determines whether to allow wildcard entries for Composite key fields in the Security Classification Maintenance table. There are important best practice recommendations associated with this setting. For more information, see <a href="more%3E">Recommended Configuration for Implementations with Many Classifications (more than 40,000 rows)</a> on page 77.</td>
<td>X: Allow wildcards for Composite keys. Blank: Do not allow wildcards for Composite keys.</td>
</tr>
<tr>
<td>SEC_CLS_WILDCARD_IDENTIFIER</td>
<td>Allow wildcard entries to Identifier value fields in the Security Classification Maintenance table. There are important best practice recommendations associated with this setting. For more information, see <a href="more%3E">Recommended Configuration for Implementations with Many Classifications (more than 40,000 rows)</a> on page 77.</td>
<td>X: Allow wildcards for Identifier values. Blank: Do not allow wildcards for Identifier values.</td>
</tr>
</tbody>
</table>
| TIME_PERIOD_IN_MILLISECONDS                | The SAP Agent collects transaction information from a single policy query and routes it to the Policy Controller. In the event that communication fails, because the Policy Controller is down or some other reason, SAP needs to be configured for how it should respond. | Activity Type: value in milliseconds, must be greater than 0. The default is 25000. Activity Handler: enter what should happen in the event of a timeout. The option you select here also applies to **MULTI_QUERY_TIMEOUT_IN_MILLISC**.  
  • Error: SAP displays an error message to the user and prevents the transaction.  
  • Information: SAP displays an Informational message to the user and allows the transaction.  
  • Blank: If the field is left blank, SAP allows the transaction to proceed without displaying a message to the user. |
5 Click **Save** to save the changes you made to /NEXTLABS/CONCFG.
Recommended Configuration for Implementations with Many Classifications (more than 40,000 rows)

If you anticipate having a large number of classification records (more than 40,000 rows) in the Security Classification Maintenance table, NextLabs recommends that you do one of the following to improve system performance:

- Do not use wildcards for Identifiers and Composite Keys. These settings are configured in the `SEC_CLS_WILDCARD_COMPOSITE_KEY` and `SEC_CLS_WILDCARD_IDENTIFIER` fields (see Connection Configuration Settings on page 72)
- If you must use wildcards for Identifiers and Composite keys, your BASIS Administrator should create an index in the `/NEXTLABS/SECIDT` table for the `MANDT` field and the relevant Security Identifiers and/or Composite keys for which wildcards are being used.

Next steps

Configuring Number Range Intervals

Configuring Number Range Intervals

Number range intervals are used for IDs for NextLabs internal operations, for instance, for security classification records and for PBSC logs and queues. Number Range Intervals are configured in the `/NEXTLABS/NRCONF` table.

After you have configured Number Ranges here, you can embed them as Sub objects of the NextLabs Number Range (which is done as a later configuration step; see Configuring the NextLabs Number Range).

Procedure

1. In the SAP interface, enter transaction `SM30`. The Maintain Table Views screen appears.
2. In Table/View, enter `/NEXTLABS/NRCONF` and click Display. The Activity Maintenance Table appears.
3. Toggle from View to Change mode.
4. Define the Security Classification Number Range Sub Object:
Chapter 3: Configuration

Configuring the NextLabs Number Range

After you have assigned Number Ranges for NextLabs processes (see Configuring Number Range Intervals), you are ready to associate them as sub objects of the NextLabs object. If SAP ECC is installed on separate systems, this procedure must be performed on each system.

Procedure

1. In the SAP interface, enter transaction SNRO. The Number Range Object Maintenance screen appears.

2. Enter or select the /NEXTLABS/ object and click Change to edit the object.
3 In the *Interval Change* screen, select the **Number Range** option.

4 In the Sub Object field, you can click the Search Help button to retrieve a list of sub objects that have been defined for `/NEXTLABS/` (as part of a prior configuration step).
5 Select a Sub Object by double-clicking it. (You need to perform this procedure for all Sub Objects.)

6 With the sub object selected, click Change Intervals.

7 In the Maintain Number Range Intervals screen, click Insert Interval.
Using the NextLabs Configuration Tool

Maintain Number Range Intervals

NR Object: NextLabs Objects
Subobject: 000001

Figure 3-23: Inserting an Interval

8 Enter the number as From 1 and To 9999999999 and click Insert Interval.

Figure 3-24: Inserting an Interval

9 Click Save to save your configuration changes.

Make the NextLabs Namespace Unmodifiable

After configuring the NextLabs Number Range, you should make the NextLabs namespace unmodifiable. This prevents users from making changes in the NextLabs namespace, which run the risk of being overwritten on a subsequent installation or upgrade of a NextLabs product.

Note: The only modifications supported for the NextLabs namespace are officially released NextLabs product code. Customers should not store other modifications to code in the NextLabs namespace because it can result in installation and upgrade issues.

Next steps
Configuring Policy Checks Based on Transaction/UI Function

Configuring Policy Checks Based on Transaction/UI Function

You can configure one of two options for how policy checks should occur for business objects: to occur only when there is a classification applied to the object, or to occur for all business objects
regardless of whether they are classified. You can set this configuration for each transaction and/or UI function. In other words, you may want policy checks for every business object for certain transactions/UI functions, but policy checks should only occur for classified business objects for other transactions/UI functions.

**Procedure**

1. In the SAP interface, enter transaction SM30. The Maintain Table Views screen appears.

2. In Table/View, enter /NEXTLABS/CHKCLS and click Display. The Activity Maintenance Table appears. The default setting (*) specifies that policy checks are enabled for all transactions/UI functions.

   You can make exceptions to the default setting by adding different settings for specific transaction codes or functions.

3. Toggle from View to Change mode.

4. Click New Entities to add a new record to the table.

5. Enter a transaction and/or UI function and select the desired security classification check behavior. The options are:
   - **Check Classified Only**: enable policy checks only when there is classification data for a business object.
   - **Check All**: enables policy checks for classified and unclassified business objects.

6. When you have entered all the required information, click Save.

**Next steps**

**Configuring Special Fields for the Security Classification Maintenance Table**

(Optional) In the /NEXTLABS/EPCLS table, you can change the properties of fields that exist in the Security Classification Maintenance table. You can mark the fields in the Security Classification Maintenance table as follows:

- **Read Only**—If the fields should be read-only.
- **Do Not Display**—If the fields should not be displayed at all.

**Procedure**

1. In the SAP interface, enter transaction SM30. The Maintain Table Views screen appears.

2. In Table/View, enter /NEXTLABS/EPCLS and click Display. The NextLabs: SECCLS: Entitlement Pack Classification Field Maintenance screen appears.
3 Toggle from Display to Change mode.

4 Click **New Entities** to add new values to the table.

   a For the first entry, enter the following values:
   - In NextLabs Entitlement Pack, enter **NXLEDRM**.
   - In Classification Field, enter **EXPSECRTY**.
   - In Field Option in SECCLS UI, select **Read Only**.

   b For the second entry, enter the following values:
   - In NextLabs Entitlement Pack, enter **NXLEDRM**.
   - In Classification Field, enter **SOURCE_SYS**.
   - In Field Option in SECCLS UI, select **Do Not Display**.

3 Click **Save**.

**Next steps**

Assign Security Identifiers to Transactions

---

**Assign Security Identifiers to Transactions**

After setting up all the identifiers (columns in the Security Classification table), you must specify the transactions for which you want each identifier to be checked. You should also decide how classifications should be applied when multiple values are present on a single transaction. It is good practice to create a default activation that applies to all transactions for an identifier and then make determinations for how classifications should be activated for individual transactions as exceptions to the default rule. All transactions that are not explicitly activated receive the default activation setting.

*Important:* It is recommended that you add the **DOCNUM**, **NXL_OBJECT**, and **MATNR** Security Identifiers.
Chapter 3: Configuration

Procedure

1. In the SAP interface, enter transaction SM30. The Maintain Table Views appears.

2. In Table/View, enter `/NEXTLABS/OPTCFG`, then click Display.

   The NextLabs Security Classification Selection screen displays all records that have been defined.

3. Toggle the view from Display to Change.

4. Click New Entries. A blank screen appears where you can enter new activation settings for Transaction code/UI Function and Security Identifier combinations.

5. Enter a valid Transaction Code or UI Function by entering the name or by clicking the Search Help icon on the right side of the field. A wildcard (*) can be used to specify all transactions.

6. Enter a valid Security Identifier by entering the name or by clicking the Search Help icon on the right side of the field.

7. To activate the Security Identifier for the Transaction code, select the Flag for Activation of Identifier check box.

8. When you finish entering activation settings, click Save and return to the NextLabs Security Classification Selection screen. Each new activation record you created appears as a new row.

   Note: After a row has been added for a Transaction Code/Security Identifier pair, you can change the Activation setting anytime by selecting or deselecting the Flag for Activation of Identifier check box.

   ![Figure 3-26: NextLabs: Security Identifier Selection Config: Overview](image)

9. When you finish configuring these settings, click Save.

Next steps

Configuring Encryption Keys
Configuring Encryption Keys

If you use EDRM to apply NextLabs protection (encryption) to files, a Shared Key must be created using Key Management Server (usually installed with the Control Center, or on the same host as the ICENet server). After this key is created, it is automatically sent to endpoints on the next heartbeat.

**Note:** The default key ring name is NL_SHARE_DEFAULT. If you create a Key Ring with any other name, it must be registered with all endpoints.

If you have distributed ICENet servers, you only need to generate Shared Key Rings and Keys on one host. Shared Keys and Key Rings are automatically distributed across all ICENet server locations where Key Management Server is installed.

For more information on Key Management and NextLabs Encryption, consult the Rights Management User’s Guide.

**Procedure**

1. In the Command prompt on the device where Key Management is installed, change directory to `<Install Dir>\Nextlabs\Policy Server\tools\keymanagement`.

2. Run the following command to create a shared key ring, where `keyRingName` is NL_SHARE_DEFAULT:

```
keymanagement.bat -u <username> -w <password> -createKeyRing -keyRingName NL_SHARE_DEFAULT
```

**Note:** The Shared Key Ring can be no larger than 16 characters. If you create a key ring name that is longer than this, no error is displayed, but the key ring is not created.

3. Run the following command to create a shared key on the newly created key ring:

```
keymanagement.bat -u <username> -w <password> -generateKey -keyRingName NL_SHARE_DEFAULT
```

**Next steps**

*Configuring the Check Table for Classification Objects*

**Configuring the Check Table for Classification Objects**

The classification objects are used for the Outbound feature. Use this table to maintain custom values that will be available in the Security Classification Maintenance table and in the /NXLE-DRM/MAPOBJ table. The Table 3-3 provides example values. If you require a default value, use the **DEFAULT** keyword. The **DEFAULT** classification object is used to apply classification values to those transactions that are not maintained in the /NXLED/RM/MAPOBJ table. For more informa-
tion about configuring the /NXLEDRM/MAPOBJ table, see Configuring Mapping Table for Classification Values.

1 Enter transaction SM30. The Maintain Table Views screen appears.

2 In the Table/View field, enter /NEXTLABS/OBJECT.

3 Click Maintain. The Change View “Text Table for Classification Object”: Overview screen appears.

4 Click New Entries to create new entries for the classification object.

5 In Classification Object field, enter the object name as required.

6 In the Description field, enter the object description.

Table 3-3: Check Table

<table>
<thead>
<tr>
<th>Classification Object</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CUSTOMER</td>
<td>CUSTOMER DATA</td>
</tr>
<tr>
<td>DEFAULT</td>
<td>DEFAULT</td>
</tr>
<tr>
<td>MATERIAL</td>
<td>MATERIAL DATA</td>
</tr>
<tr>
<td>OFFICE</td>
<td>OFFICE INBOX</td>
</tr>
<tr>
<td>PO</td>
<td>PO DATA</td>
</tr>
<tr>
<td>SALES</td>
<td>SALES DATA</td>
</tr>
<tr>
<td>VENDOR</td>
<td>VENDOR DATA</td>
</tr>
</tbody>
</table>

7 Click Save.

Next steps

Configuring Mapping Table for Classification Values

Configuring Mapping Table for Classification Values

Use this procedure to configure classifications that are inserted into documents upon download (Outbound). This is achieved by associating the transaction code with an object that is already classified in the Security Classification Maintenance table. For more information about using the Outbound feature, including the use of the /NXLEDRM/MAPOBJ table, see Protecting Data Leaving the SAP System (Outbound).

Before You Begin

Ensure that any required Security Classification records are already created using the transaction /NEXTLABS/SEC_CLS for Customer data, Material data, Vendor data, PO data, Sales data,
Configuring Mapping Table for Classification Values

Default, and Office Inbox with the corresponding identifier in the Object for Tags Classification field.

Procedure

1. Enter transaction SM30. The Maintain Table Views screen appears.

![Figure 3-27: Maintain Table Views](image)

2. In the Table/View field, enter /NXLEDRM/MAPOBJ and click Display. The NextLabs: Mapping Table for Object Classification table appears.

3. Toggle from View to Change mode.

4. Enter the transaction code for the desired Object mapping.

5. Enter Source table information if the transaction code is SE11/SE16/SE16N or similar, and if a custom mapping, specific to the table data source, is required.

6. Select the desired Object and then click Save to save the record.
Configure Rights Management Client Installer Path

When transactional data is leaving the SAP system (outbound), EDRM requires a default path for Rights Management Client (RMC) for the endpoints. A logical path pointing to this location must also be configured within SAP.

After the logical path is configured, whenever a user tries to download data from SAP, the logical path information is retrieved and the data is sent to RMC. Subsequently, RMC performs the rights protection on the downloaded file.

In this activity, the path to the RMC directory is configured using the following steps:

1. In SAP, enter the FILE transaction. The Logical File Path Definition screen appears.
2. In the list of logical paths, locate /NEXTLABS/RIGHTS_MANAGEMENT_PATH.
3. Select the path and click Assignment of Physical Path to Logical Path.
4. In the Physical path field, specify the path to the conversion directory:
During processing `<FILENAME>` is replaced with the file being downloaded from SAP.

5 Click Save.

**Next steps**

Defining Implementations

---

### Defining Implementations

The last configuration step is to define enhancement implementations for NextLabs EDRM. The enhancements implement the interception points for policy checks for transactions and business objects.

Enhancement implementations require a developer license and knowledge of your SAP environment. The latter is important because the instructions provided in this guide cannot anticipate or address certain variables in your landscape that might impact how you should configure enhancements. For instance, one unknown variable is what enhancement implementations are already in use in your system, and how NextLabs code should be integrated with existing implementations.

---

### Implementation Details

**Table 3-4** lists the required enhancement implementations for systems using the Entitlement Pack for EDRM.

#### Table 3-4: Required Enhancement Implementations for EDRM

<table>
<thead>
<tr>
<th>Feature</th>
<th>Type</th>
<th>T-Code</th>
<th>Program/Function Module/BADI Name</th>
<th>Form/Enhancement Point/Method</th>
<th>Location</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>UDRM</td>
<td>Implicit Enh.</td>
<td>SE38</td>
<td>LCV110LC1</td>
<td>HANDLE_NODE_CTMENU</td>
<td>End of method</td>
<td>INCLUDE /NXLEDRM/GET_MENU_FCODE.</td>
</tr>
<tr>
<td>UDRM</td>
<td>BADI</td>
<td>SE19</td>
<td>DOCUMENT/Main02</td>
<td>D101_BEFORE PAI</td>
<td>INCLUDE /NXLEDRM/IRM_FUNC_DMS.</td>
<td></td>
</tr>
<tr>
<td>READ TAGS</td>
<td>BADI</td>
<td>SE19</td>
<td>DOCUMENT/Main01</td>
<td>AFTER_SAVE</td>
<td>INCLUDE /NXLEDRM/EDRM_READ_TAGS.</td>
<td></td>
</tr>
<tr>
<td>Outbound</td>
<td>Implicit Enh.</td>
<td>SE37</td>
<td>ITS_DOWNLOAD</td>
<td></td>
<td>End of function module</td>
<td>INCLUDE /NXLEDRM/PROTECT_IT'S_DOWNLOAD.</td>
</tr>
<tr>
<td>Outbound</td>
<td>Implicit Enh.</td>
<td>SE37</td>
<td>ITS_DOWNLOAD</td>
<td></td>
<td>Start of function module</td>
<td>INCLUDE /NXLEDRM/CHK_RMC_PATH_ON_DWLD.</td>
</tr>
<tr>
<td>Outbound</td>
<td>Implicit Enh.</td>
<td>SE38</td>
<td>SAPLSFES</td>
<td>Include: LSFESUXX</td>
<td>End of form</td>
<td>INCLUDE /NXLEDRM/PROTECT.GUI_DOWNLOAD.</td>
</tr>
<tr>
<td>Outbound</td>
<td>Implicit Enh.</td>
<td>SE37</td>
<td>GUI_DOWNLOAD</td>
<td></td>
<td>Start of function module</td>
<td>INCLUDE /NXLEDRM/CHK_RMC_PATH_ON_DWLD.</td>
</tr>
</tbody>
</table>
Chapter 3: Configuration

Table 3-4: Required Enhancement Implementations for EDRM (Continued)

<table>
<thead>
<tr>
<th>Feature</th>
<th>Type</th>
<th>T-Code</th>
<th>Program/Function Module/BADI Name</th>
<th>Form/Enhancement Point/Method</th>
<th>Location</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outbound</td>
<td>Implicit Enh.</td>
<td>SE37</td>
<td>POPUP_TO_DECIDE_LIST</td>
<td>Start of function module</td>
<td>INCLUDE /NXLEDRM/CLIPBOARD_BLOCK.</td>
<td></td>
</tr>
<tr>
<td>Outbound</td>
<td>Implicit Enh.</td>
<td>SE38</td>
<td>SAPLSTXW</td>
<td>Include: LSTXWFCC Form: CC_DISPLAY_PDF_SHOW</td>
<td>Start of form</td>
<td>INCLUDE /NXLEDRM/SMARTFM its_pdf_protect.</td>
</tr>
<tr>
<td>Outbound</td>
<td>Implicit Enh.</td>
<td>SE38</td>
<td>SAPLSTXW</td>
<td>Include: LSTXWFCC Form: CC_DISPLAY_PDF_SHOW</td>
<td>Start of form</td>
<td>INCLUDE /NXLEDRM/SMARTFM its_pdf_protect.</td>
</tr>
<tr>
<td>Outbound</td>
<td>Implicit Enh.</td>
<td>SE38</td>
<td>SAPLSTXW</td>
<td>Include: LSTXWFCC Form: GET_CACHED_URL</td>
<td>Start of form</td>
<td>INCLUDE /NXLEDRM/SMARTFORM_PDF_PREVIEW.</td>
</tr>
<tr>
<td>Outbound</td>
<td>Implicit Enh.</td>
<td>SE38</td>
<td>SAPLSTXW</td>
<td>Include: LSTXWFCC Form: GET_CACHED_URL</td>
<td>Start of form</td>
<td>INCLUDE /NXLEDRM/SMARTFORM_PDF_PREVIEW.</td>
</tr>
<tr>
<td>Outbound</td>
<td>Implicit Enh.</td>
<td>SE38</td>
<td>SAPLSPOOL_SP01R</td>
<td>Include: LSPPOOL SP01RUXX Include: LSPPOOL SP01RU19 Form: GET selected3</td>
<td>End of form</td>
<td>INCLUDE /NXLEDRM/SPOOL_EXP AS PDF.</td>
</tr>
<tr>
<td>Outbound</td>
<td>Implicit Enh.</td>
<td>SE38</td>
<td>SAPLSPOOL_SP01R</td>
<td>Include: LSPPOOL SP01RUXX Include: LSPPOOL SP01RU19 Form: DOWNLOAD.Selected_pdf</td>
<td>Start of form</td>
<td>INCLUDE /NXLEDRM/GET_FCODE_EXPORT_PDF.</td>
</tr>
</tbody>
</table>

Configuring Temporary File Location for Processing Originals

EDRM can convert documents with supported file formats into the NextLabs-protected format (NXL) and/or insert classification values into the documents. Also, when a user tries to open a file by using a DIR transaction code (CV01N, CV02N or CV03N), the EDRM solution checks the user credentials against the properties of the NextLabs-protected file and provides the required authentication.

To protect or classify originals in a DIR, the originals must be stored in a temporary location for processing. The temporary location can be, for instance, on a content server or a shared folder path accessible from SAP. The files are temporarily placed in this location while being processed (protected, classified, or to remove protection). After the processing is complete, the files are deleted from the temporary location and they will be placed back into the original location.

Policies can be designed and implemented to get the temporary server location path based on the document storage category. For example, a storage category can point to the content server temporary storage location. Otherwise, if a policy is not implemented, the BAdI /NXLEDRM/ENH_IRM_BADI can be used to implement custom logic to get the temporary storage location.

Use this procedure to implement a custom logic to configure temporary storage location for processing originals.

**Procedure**

1. In the SAP interface, enter transaction SE19. The Business Add-Ins screen appears.

2. In the Create Implementation section, select New BADI-Enhancement Spot and enter /NXLEDRM/ENH_IRM_BADI.

3. Click Create Impl.
4 Enter an implementation name.

5 In the Implementation Short Text field, enter a description, and then click OK.

6 Enter the BADI Implementation name, Implementation Class and BADI Definition `/NXLEDRM/ENH_IRM_BADI`.

7 Click Continue.

8 Click the implementing class. The `GET_DYN_SERVER_LOC` method is available for inserting code to get the server temporary location for the document storage category.

9 Double-click the method and write the code for the method.

10 Save and activate the method.

11 Save and activate the BADI implementation.

Implementations

BADI enhancement implementations are based on object-oriented concepts of interfaces, classes, and methods. You add custom code to methods in an interface. The following is the procedure for creating a BADI enhancement implementation.

There are four types of enhancement implementations:

- BADI Enhancements
- Explicit Enhancements
- Implicit Enhancements
- User Exit

The step-by-step procedure is provided once for each type of implementation. Only the pertinent details are provided for each Implementation ID, for example, the transaction to run, the program or function module to modify, the enhancement point, and the code to insert.

**BADI Enhancements**

The following is the procedure for creating a BADI enhancement.

**Procedure**

1. In the SAP interface, enter transaction SE18. The BADI Builder appears.
2. Enter the BADI name, and click Display.
3. Select Implementation > Create.
4. In Implementation Name, enter a name for the implementation, and click Continue.
5. In Implementation Short Text, provide a description of the implementation.
6. Click the Interface tab.
7 Double-click the method in which to add the custom code.
8 Save the implementation.
9 Insert the code in the method.
10 Save and activate the method.
11 Click the Back button and activate the implementation.

Explicit Enhancements
Explicit enhancements are predefined enhancements sections provided explicitly by SAP. The following is the procedure for creating an explicit enhancement.

Procedure
1 In the SAP interface, enter transaction SE38. The ABAP Editor appears.
2 In the next screen, in Program, enter the name of the program. Click Display.
3 Click the Enhance button on the tool bar.
4 Locate the enhancement point in which to insert your code. Right-click the line that contains the text "ENHANCEMENT-POINT" or "ENHANCEMENT-SECTION," and select Enhancement Operations > Create Implementation.
5 Select an existing enhancement implementation or create a new one. To create a new enhancement implementation:
   a Click the Create Enhancement Implementation button.
   b Enter a name for the enhancement implementation and a description.
   c Save the enhancement implementation, then select it, if necessary.
6 Insert code in the Enhancement block.
7 Activate the implementation.

Implicit Enhancements
Implicit enhancement implementations are added at the beginning and end of a program, method, function module, or subroutine. The following is the procedure for creating an implicit enhancement implementation.

Procedure
1 In the SAP interface, enter one of the following transactions:
   • SE37: Displays the Function Builder
   • SE38: Displays the ABAP Editor
   • SE24: Displays the Class Builder
2 In the next screen, in Function Module or Program or Object, enter the name of the function module or program or class. Click Display.

If you specified a class, double-click the method to which to add an enhancement implementation.

3 Click the Enhance button on the tool bar.

4 In the main menu, select Edit > Enhancement Operations > Show Implicit Enhancement Options.

Enhancement points are indicated by an arrow on the left of the line number and by """" in the line, as shown in the following example.

5 Locate the specific enhancement point in which to insert your code, for example, at the beginning of a particular form, or at the end of a particular function module. Right-click the line and select Enhancement Operations > Create Implementation.

6 Select Code as the enhancement type.

7 Select an existing enhancement implementation or create a new one. To create a new enhancement implementation:

   a Click the Create Enhancement Implementation button.
   b Enter a name for the enhancement implementation and a description.
   c Save the enhancement implementation, then select it, if necessary.

8 Insert code in the Enhancement block.

9 Activate the implementation.

User Exit

A user exit is a location where you can access SAP program components and include your program enhancements. The following is the procedure for implementing a user exit enhancement.

Procedure

1 In the SAP interface, enter transaction CMOD. The Project Management of SAP Enhancements screen appears.

2 In Project, enter a name for a new project, and click Create. Alternatively, enter the name of an existing project, and click Display.

3 In Short text, enter a project description. Click Save.

4 Click Enhancement assignments to assign an enhancement point to the project.
5 In the **Enhancement** column, enter the name of the enhancement to implement, then press Enter.

6 Click **Save**.

7 Click **Components**. A list of enhancements in the project appears. Each enhancement includes a function exit.

8 Double-click the function exit. The Function Builder appears.

9 Find the line **INCLUDE <include program>**. Double-click the name of the include program. A message appears if this include has not been previously implemented.

10 If the include has not been implemented, press Enter. A message prompts you to create an object. Click **Yes**.

11 In the include program, insert your code.

12 Save and activate the include program.

13 Navigate back to the screen that displays the enhancements in the project.

14 Activate the project.
This section describes how to use the features in NextLabs Enterprise Digital Rights Management for SAP.

Topics:

- Reading Tags
- Applying Classification to Documents
- Protecting Documents
- Applying Protection and Classification to Documents Simultaneously
- Removing Document Protection
- Applying Protection and Classification Using Automated Digital Rights Management
- Protecting Data Leaving the SAP System (Outbound)

**Reading Tags**

The Read Tags feature reads the pre-existing classifications when you upload a document into SAP and then inserts those classifications into the corresponding columns of the Security Classification Maintenance table. Only after you check-in and save the document, and based on policy, the Read Tags feature automatically inserts the classifications into the corresponding columns of the Security Classification Maintenance table. For more information about the Read Tags policy, see Example Policy: Read Tags.

This example procedure lists the steps to use the Read Tags feature.

**Procedure**

1. In the SAP interface, enter the CV01N transaction code. The Create Document screen appears.
3. In the Description field, enter a description for the DIR.
4. Click Open Original.
5. Locate the document (with pre-existing tags) you want to upload and then click Open. The file is imported into the SAP system.
6. In the Originals section, right-click the document that you imported and then click Check In.

A pop-up window appears.

7. Select a storage category and press ENTER.

8. Click Save. A message appears stating that the document (with document number) is created.

9. To verify if the pre-existing tags are inserted into the Security Classification Maintenance table, perform the following steps:

   a. In the SAP interface, enter the /nextlabs/sec_cls transaction code. The NextLabs: Security Classification Maintenance screen appears.
b In the Document field, enter the document number and click Modify/Display. The Security Classification screen appears.

Applying Classification to Documents

The Classify feature enables you to classify native or NextLabs-protected documents, based on the policy decision. If you run the Classify feature on a native document and then protect the document, the system removes the existing document classification. If you want to protect and also classify a document, you can either choose to use the Protect and Classify feature or use the Protect feature first and then use the Classify feature. For more information about the Classify policy, Example Policy: Classify.

This example procedure lists the steps to use the Classify feature.

Procedure

1 In the SAP interface, enter the CV02N transaction code. The Change Document screen appears.
2 In the Originals section, right-click the document you want to classify and then click Classify.
If the document classification is successful, based on the policy decision, the system inserts the new classification values to the existing document properties.

In the above example, the `export license` and `export security` classifications are inserted to the document properties according to the policy decision.

### Protecting Documents

The Protect feature enables you to encrypt native documents in the SAP system into NextLabs-protected documents, based on the policy decision. For more information about the Protect policy, see Example Policy: Protect.

**Note:** You can view the NextLabs-protected documents using the NextLabs Rights Management Client.

This example procedure lists the steps to use the Protect feature.

**Procedure**

1. In the SAP interface, enter the transaction code `CV02N`. The Change Document screen appears.
2 In the Originals section, right-click the document that you want to encrypt and then click **Protect**.
3 If the document protection is successful, the system converts the native document into a NextLabs-protected file.

In the above example, the Design Document 3.doc document is converted into a NextLabs-protected document.

Applying Protection and Classification to Documents Simultaneously

The Protect and Classify feature enables you to encrypt native documents and simultaneously classify the documents, based on the policy decision. For more information about the Protect and Classify policy, see Example Policy: Protect and Classify.
Note: You can view the NextLabs-protected documents using the NextLabs Rights Management Client.

This example procedure lists the steps to use the Protect and Classify option.

Procedure

1. In the SAP interface, enter the transaction code CV02N. The Change Document screen appears.

2. In the Originals section, right-click the file that you want to encrypt and tag, then click Protect and Classify.

3. If the document protection and classification is successful, the system converts the native document into a NextLabs-protected document and then applies the classification to the
Removing Document Protection

The Remove Protection feature, based on the policy decision, enables authorized users to view or edit NextLabs-protected documents in their native format by downloading the documents to a temporary location. The documents remain protected within SAP, ensuring that unauthorized users are unable to access them. For more information about the Remove Protection policy, see Example Policy: Remove Protection.

This example procedure lists the steps to use the Remove Protection feature.
Procedure

1. In the SAP interface, enter the transaction code **CV02N**. The Change Document screen appears.

2. In the Originals section, right-click the document that you want to decrypt (file with the `.nxl` extension) and then click **Remove Protection**.

If decryption is successful, the decrypted document is downloaded to a temporary location and displayed in its native application. In the above example, the `Design Document 3.doc.nxl` file is decrypted and the system opens the file in Microsoft Word application.
Applying Protection and Classification Using Automated Digital Rights Management

Administrators can use Automated Digital Rights Management (ADRM) to protect and classify multiple documents, based on policies, in a single operation. The feature can be run in the manual or auto run mode and the reports can be run in the foreground or background processing mode. The documents remain protected even after they are downloaded from the SAP system because the classification and protection are persistent with the documents. Authorized users can access the protected documents by using NextLabs Rights Management Client (RMC).

This example procedure lists the steps to run the Automated Digital Rights Management feature.
Chapter 4: Using EDRM Features

Procedure

1. In the SAP interface, enter the transaction code /nxledrm/rm. The NextLabs EDRM Rights Management for DIR (DMS) screen appears.

2. **Optional** To run a batch job only on document available in classified DIRs (that is, DIRs in the Security Classification Maintenance table), select the **Select only Classified DIRs** check-box.

3. **Optional** To run a batch job only on document available in storage categories, either specify a DIR or a range of DIRs in the Storage Category box.

4. Use one of the following options:
   - **Manual RUN**
1 In the DIR Selection-Criteria section, either specify a single value or a range of values for one of following fields:

- Document
- Document Type
- Document Part
- Document Version

2 (Optional) To run a batch job on document linked with material master, in the Material Selection-Criteria section, either specify a single material or a range of materials.

3 Click (Execute).

- Auto RUN

1 In the DIR Selection-Criteria, select one or both of the checkboxes:

- Select DIRs based on Seccls Classification changes, since last run job date/time
- Select DIRs based on changes to DIRs only, since last run job date/time

2 Click Save.

The system displays the Variant Attributes screen.

3 In the Variant Name field, enter a variant name.

4 In the Description field, enter the description for the variant.

5 Select the Only for Background Processing check-box.

6 Click Save.
Define a background job for EDRM Rights Management for DIR by using the \texttt{SM36} transaction code. When defining a Job Step, provide \\
\texttt{/NXLEDRM/RIGHTS_MNGMT_V1} as the ABAP program name and the variant name that you created in step 3.

Protecting Data Leaving the SAP System (Outbound)

NextLabs EDRM automatically protects the sensitive business data leaving (outbound) the SAP ERP system. This includes master data and transactional data that are accessed in SAP through transactions, reports, data dictionary, table browsers, smart forms, and queries. The system automatically protects, and optionally classifies, the data being downloaded and saved as a local file or being sent to the SAP inbox. The documents remain protected even after they are downloaded from the SAP system because classification and protection are persistent with the documents.

To configure classifications that need to be inserted into documents upon download, you must associate the transaction code with an object that is already classified in the Security Classification Maintenance table. For more information, see Configuring Mapping Table for Classification Values.

Whenever a user performs a document download using a configured transaction code, the associated object is fetched from the mapping table. Next, the corresponding object is fetched from the Security Classification Maintenance table, where the associated classifications are defined. Finally, those classifications are inserted into the file when it is downloaded. For example, you can create a record in the \texttt{/NXLEDRM/MAPOBJ} table that contains the transaction code \texttt{ME23N} and Object PO data. This specifies that the EDRM system inserts classifications corresponding to the PO data object into all downloaded files initiated from the \texttt{ME23N} transaction code.

Additionally, for table data downloads (when using transaction code \texttt{SE11}, \texttt{SE16}, or \texttt{SE16N}), there is an additional field (Source) that must be used to associate a specific source table with a corresponding object. This is accomplished by providing a data table name in the Source column, in addition to the transaction code and object information. If you want to classify downloads from specific data source tables, those tables must be listed here. For any source tables not specified, no classifications are applied.

For example, a record could be created in the \texttt{MAPOBJ} table which contains transaction code \texttt{SE16}, Source MARA, and Object Material data. This would specify that when data from the MARA table is accessed in the transaction code \texttt{SE16}, the corresponding classifications for type Material would be fetched from the Security Classification Maintenance table.

\begin{itemize}
  \item \textbf{Note:} The default object should not be mapped to any transaction code. For a transaction code that is not maintained in the \texttt{/NXLEDRM/MAPOBJ} table, the default classification value is
\end{itemize}
fetched from the Security Classification Maintenance table, if maintained there.

Authorized users can view the NextLabs-protected documents in the SAP GUI by either using Rights Management Server (RMS) or Rights Management Client (RMC), provided that RMC is installed on that endpoint. You can identify NextLabs protected documents by the blue Rights Management icon (refer to the image below.)

You can check permissions on the NextLabs protected file by right-clicking the downloaded file, and then selecting Nextlabs > Check Permissions.

![Check Permissions](image)

**Figure 4-1: Check Permissions**

Here are few examples where EDRM protects data by converting the data into NextLabs-protected file, if outbound implementations are active.

**Example 1**

When you use Data Browser in SAP and you choose to download (Figure 4-2) the result list as any of the formats listed in Save list in file pop-up menu (Figure 4-3), the file is automatically NextLabs-protected.
Chapter 4: Using EDRM Features

Figure 4-2: Data Downloaded from SAP Download Context Menu

Note: If outbound implementations are active, the In the clipboard menu item is disabled.
Example 2

If you have mapped the transaction code VCUST to Customer data object (Figure 4-4) and you have defined Customer Object with Classification EXPORT LICENSE as CUSLIC1 (Figure 4-5) in the Security Classification Maintenance table, the CUSLIC1 classification is inserted into downloaded files (Figure 4-6).

Figure 4-4: Map tcode VCUST to Customer Data

Figure 4-5: Define Customer Object in the Security Classification Maintenance Table

Figure 4-6: Check Classification

Example 3
When you use the List Menu to export SAP data in any of the file formats listed in Figure 4-7, the file is automatically NextLabs-protected.

![Figure 4-7: Data Downloaded from SAP List Menu](image)

**Example 4**

When you choose to download SAP data as a local file from the System menu (Figure 4-8) or from the toolbar as a local file (Figure 4-9), Word processing (Figure 4-10), or Spreadsheet (Figure 4-11), the file is automatically NextLabs-protected.

![Figure 4-8: Data Downloaded from SAP System Menu](image)

![Figure 4-9: Data Downloaded from SAP Toolbar](image)
Example 5

When you choose to save an SAP list from the System menu, the downloaded office folders (Figure 4-12) or report tree (Figure 4-13) is automatically Next-Labs-protected.
Example 6

When you use the Export option in the SAP Business Workplace (Figure 4-14) or the Export option available in the Edit menu (Figure 4-15), the file is automatically NextLabs-protected.
### Figure 4-15: Data Downloaded from Business Workplace Export 2
This section explains how to use classifications and policies

Topics:

- About SAP Policies
- Designing EDRM Policies
- Applying Security Classifications

About SAP Policies

This section provides information about some policy characteristics that are unique to the Entitlement Manager for SAP, including:

- About the SAP Resource String
- About SAP Policy Messages
- About Custom Obligations

Note: For more detailed information about NextLabs policies in general, see the Policy Studio User’s Guide.
About the SAP Resource String

When you design SAP policies in Policy Studio, you create an SAP Resource component and define an SAP resource string to identify which transactions/UI functions, and business objects you are trying to target.

The resource string uses the following syntax:

**sap://Server/System/Client/Application/Transaction Code or UI Function Name/Business Object**. Wildcards are supported, so for an example where we want to apply across all Transactions, and Business Objects, the string would be:

**sap://server/system/client/**/*/**

About SAP Policy Messages

User alerts can be displayed when a NextLabs policy is enforced by using these options:

- Defining a NextLabs User Alert in Policy Studio
- Defining an SAP Message Class in SAP and referencing in a Policy Studio policy. To reference a message within an SAP Message Class, specify both the class and the message ID number in Policy Studio.

About Custom Obligations

As with any NextLabs product, you can design policies for Enterprise Digital Rights Management for SAP to automatically trigger a process that occurs upon policy evaluation. With other NextLabs products, this process could be any executable a customer writes.
Designing EDRM Policies

This section provides guidelines for creating EDRM policies and few examples of NextLabs EDRM policies.

Topics include:

- Guidelines for Creating Policies
- Example Policy: Temporary File Location
- Example Policy: Read Tags
- Example Policy: Classify
- Example Policy: Protect
- Example Policy: Protect and Classify
- Example Policy: Remove Protection
- Example Policy: Read Tags
- Example Policy: Classify
- Example Policy: Protect
- Example Policy: Protect and Classify
- Example Policy: Remove Protection

Guidelines for Creating Policies

Use Table 5-1 as a reference to create EDRM policies. For more information about classification types and modes, see:

- Classification Types
- Modes

<table>
<thead>
<tr>
<th>Protect Value</th>
<th>Classification Type</th>
<th>Resulting Policy</th>
</tr>
</thead>
<tbody>
<tr>
<td>No</td>
<td>Any value except No-Classification</td>
<td>Classify</td>
</tr>
<tr>
<td>Yes</td>
<td>No-Classification</td>
<td>Protect</td>
</tr>
<tr>
<td>Yes</td>
<td>Any value except No-Classification</td>
<td>Protect and Classify</td>
</tr>
</tbody>
</table>

Classification Types

The classification types that are available for creating policies are:

- **All-Column**: values from all columns in the Security Classification Maintenance table for the DIR are used to classify the original.
• **Specific-Column**: values from the specified column in the Security Classification Maintenance table for the DIR are used to classify the original. In addition, any value in the Value field within the obligation is also used to classify the original.

• **User-Defined**: no values from the Security Classification Maintenance table are used to classify the original. The original is classified with the Field and Value configured in the obligation only.

**Modes**

The modes that are available for creating policies are:

• **Overwrite**: Overwrite the old values with new values.

• **Append**: New values are appended to old values. However, if there are multiple policies with different modes, append overrides overwrite.

**Example Policy: Temporary File Location**

The Read Tags feature uses this temporary file location to process the documents.

To configure a Temporary File Location policy, shown in Figure 5-2, perform these general steps:

1. Create a Document policy.
2. Set the enforcement type to **Allow**.
3. Select the **Get Location** action component and drag it into the Action field. This is the only valid action.
4. In the Obligations area, choose **Custom Obligation**, and select these values:
   - Name: **SAP—EDRM—Temporary File Location for Processing Originals**
   - Storage Category: `<storage_category_name>`
   - File Location: `<file_location>`
5. Submit and Deploy the policy.
Figure 5-2: Example Policy: Temporary File Location
Example Policy: Read Tags

To create a Read Tags policy, shown in Figure 5-3, perform these general steps:

1. Create a Document policy.
2. Set the enforcement type to Allow.
3. Select the Save action component and drag it into the Action field. This is the only valid action.
4. In the Obligations area, choose Custom Obligation, and select these values:
   - (Mandatory) Name: SAP-EDRM-Maintain Classification from Tags
   - Read Tags: Read Tags from Original
• *(Mandatory)* Read Tags: Read Tags from Original

*Figure 5-3: Example Policy: Read Tags*
Example Policy: Classify

To create a Classify policy, shown in Figure 5-4, perform these general steps:

1. Create a Document policy.
2. Set the enforcement type to Allow.
3. Select the Upload action component and drag it into the Action field. This is the only valid action.
4. In the Obligations area, choose Custom Obligation, and select these values:

   - *(Mandatory)* Name: SAP-EDRM Rights Management
   - *(Mandatory)* Protect: No
   - *(Mandatory)* Classify: Specific-Column
   - Field: Specify a classification type. For example, EXPORT LICENSE.
   - Value: Specify a value for the classification type. For example, ITAR.
   - Mode: Append
Designing EDRM Policies

Document Policy
EDRM-Classify

![Diagram of Document Policy with fields for Enforcement, Subject, Perform the Following, On Resources, Conditions, Subpolicy, and Obligations]

**Figure 5-4: Example Policy: Classify**
Example Policy: Protect

To create a Protect policy, shown in Figure 5-5, perform these general steps:

1. Create a Document policy.

2. Set the enforcement type to Allow.

3. Select the Upload action component and drag it into the Action field. This is the only valid action.

4. In the Obligations area, choose Custom Obligation, and select these values:
   - (Mandatory) Name: SAP-EDRM Rights Management
   - (Mandatory) Protect: Yes
   - Classify: No-Classification
   - Mode: Not Applicable
5 Submit and Deploy the policy.

Figure 5-5: Example Policy: Protect
Example Policy: Protect and Classify

To create a Protect and Classify policy, shown in Figure 5-6, perform these general steps:

1. Create a Document policy.
2. Set the enforcement type to Allow.
3. Select the Upload action component and drag it into the Action field. This is the only valid action.
4. In the Obligations area, choose Custom Obligation, and select these values:
   - (Mandatory) Name: SAP-EDRM Rights Management
   - (Mandatory) Protect: Yes
   - (Mandatory) Classify: Choose one of the following:
     - User-Defined
     - All-Column
     - Specific-Column
   - Field: Specify a classification type. For example, EXPORT LICENSE.
   - Value: Specify a value for the classification type. For example, ITAR.
   - Mode: Append
5. Submit and Deploy the policy.
Figure 5-6: Example Policy: Protect and Classify
Example Policy: Remove Protection

To create a Remove Protection policy, shown in Figure 5-7, perform these general steps:

1. Create a Document policy.
2. Set the enforcement type to Allow.
3. Select the Download action component and drag it into the Action field. This is the only valid action.
4. In the Obligations area, choose Custom Obligation, and select these values:
   - (Mandatory) Name: SAP-EDRM Remove Protection
   - (Mandatory) Remove Protection: Yes
5. Submit and Deploy the policy.
Figure 5-7: Example Policy: Remove Protection

When an authorized user runs the Remove Protection feature on a NextLabs-protected document in the SAP system, the system does the following:

- Downloads the document to a temporary file location.
- Decrypts the document.
- Displays the document in the native format.
Applying Security Classifications

Security Classifications are maintained as rows in the Security Classification Maintenance screen, where you classify business objects by Material, by Document, by the versions (About Compound Classification Keys on page 132) of Materials or Documents, or by another Identifier that you define. The Security Classification Maintenance screen can reside within SAP ECC. This section provides the following information about applying classifications:

- About Classification Data on page 132
- About Compound Classification Keys on page 132
- Applying Security Classifications Manually on page 135

About Classification Data

The identifiers (the column headings) are stored in the table /NEXTLABS/SECIDT and the classification values (the values beneath the headings) are stored in the table /NEXTLABS/SECCLS. The link between the two tables is the Reference Id.

Some columns in the Security Classification screen are standard and others can be customized to address a particular need or business practice.

There are some configuration best practices that may apply depending on how many records you intend to maintain for the Security Classification Maintenance table. For more information about these best practices, see Recommended Configuration for Implementations with Many Classifications (more than 40,000 rows) on page 77.

About Compound Classification Keys

Businesses typically use SAP to manage multiple versions of business objects. As a document evolves into multiple versions, different versions may require different access controls. For instance, in an export control use case, certain versions of a document may not be subject to export regulations, whereas others are. The Entitlement Manager for SAP provides compound classification key support, so you can design policies to reference the version of business objects. In the Figure 5-8, columns in the Security Classification Maintenance table can reference Document Type, Part, and Version, but these values can be configured to represent your business requirements.

When multiple compound classification keys are present, a single record (row) can include classification values for multiple keys. For example, you could configure a classification where for Document DOC-100, Doc Type = DRW, and Doc Version = Internal, Export Compliance should be “ITAR.”

Note: Compound Keys are supported in the back-end data system SAP ECC only.
Best Practices for Retrieving Security Classification Records

If you have a large number of records in the Security Classification Maintenance screen, it is best to follow search techniques that target data you want to retrieve in the most efficient manner. The following tips ensure that your searches are optimized and return results quickly and without error:

- For the best results, always specify a Security Identifier and/or Composite Key for a search, then supply additional search items (such as classification fields). A search based on classification values only may result in too many records or slow search returns.
- If you want to retrieve Classification values for a set of Security Identifiers or Composite Keys, you can use a partial wildcard to specify the set. In other words, you can search on MAT* and Classification Value ITAR to retrieve a list of all material records classified as ITAR.
- If you want to run a search based on classification fields (all ITAR records, regardless of Security Identifier and/or Composite Key), it is recommended that you limit search records using another search criteria, such as date range or number of records.

Viewing Security Classification Records

You can use the Security Classification Maintenance screen to determine which records to view.

Procedure

1. In the SAP interface, enter transaction /nextlabs/sec_cls. The Security Classification screen appears.
2. Do any of the following:
   - Enter selection criteria to specify which records you want to view.
• Create a new security record (in which case, you do not need to define selection criteria).
• Change the selection options you use to specify records you want to view.

Figure 5-9: Security Classification Maintenance Screen: Change Selection Options

3 If desired, click Change Selection Options to view different selection options other than the ones displayed. This determines the options you can use in the Security Classification Maintenance Selection screen.

The choices are currently Security Identifier, Composite Key, Change History, and Security Classification.

Note: The selection options that display by default are configured in the CONCFG table. For more information, see Configuring SAP Data Handling and Connection Settings on page 70.

4 Click Continue.
Applying Security Classifications Manually

In the Security Classification Maintenance screen, you can enter new rows to assign new classifications, or upload multiple rows at once by importing them from an Excel spreadsheet. Note the flexibility of how classifications can be manually assigned: they can be assigned by specific transaction code or for all transactions at once. You can also use partial wildcards to apply classifications to a whole family of materials or documents at once.

Note: There are best practices around using wildcards for customer implementations that include a large number of security classification records. For more information, see Recommended Configuration for Implementations with Many Classifications (more than 40,000 rows) on page 77.

You can also add, delete, and modify classifications. Values in these columns are mapped to NextLabs Policy Controllers. They can be referenced in policies you design in Policy Studio.

Procedure

1. In the SAP interface, enter transaction /nextlabs/sec_cls. The Security Classification Maintenance screen appears.

2. Since we are creating new records, click Create without specifying any selection criteria. The Security Classification screen displays no records.

3. Click the Insert Row Icon in the toolbar to add a new Security Classification record.

4. For each row, you can choose only one identifier. Enter the identifier, or retrieve a valid number by clicking the pop up icon that appears on the right side of the field when it is active.
5 In the Transaction column, enter a Material or Document transaction or UI Function code. For example, you can apply a classification to the viewing of the Material (transaction code MM03), or to the modification of a Document (CV02N). You can enter a wildcard (*) here to apply a classification to all transactions at once.

6 Enter the classification values and pertinent classification information in the remaining columns, as needed. Remember that the columns that display here depend on how you have defined Identifiers during your system’s configuration. When you are finished, click Save.
This section describes the administrative procedures associated with Entitlement Manager for SAP, including maintenance procedures related to Security Classifications in SAP, as well as maintenance procedures related to the NextLabs Policy Server and Control Center.

Topics:

- Maintenance for Entitlement Manager for SAP ECC
- Viewing NextLabs Log Information in SAP
- Configuration and Management
- About Service Account Permissions
- About Bundle Encryption
- Managing Enforcer Policies
- Managing Event Logging
- Load Balancing the Policy Controller
- Deleting Decrypted Temporary Documents

## Maintenance for Entitlement Manager for SAP ECC

The procedures necessary for maintaining SAP Security Classifications are already discussed in different areas of this manual. Table 6-1 provides cross-references to sections where you can find more information on each procedure.

### Table 6-1: Links to Maintenance Procedures Discussed in this Guide

<table>
<thead>
<tr>
<th>Procedure</th>
<th>See section…</th>
</tr>
</thead>
<tbody>
<tr>
<td>You may need to add or edit the Identifiers that display in the Security Classification screen. You add an append structure to the classification table to create a new custom identifier. Then you map the new identifier to the NextLabs Policy Controller.</td>
<td>See Dynamic User and Resource Attributes on page 153 and Mapping Security Fields (SECMPG) on page 66.</td>
</tr>
<tr>
<td>You may need to enter new Security Classification values to Materials and Document, or change the classification settings of existing ones.</td>
<td>See Applying Security Classifications on page 132.</td>
</tr>
</tbody>
</table>
Viewing NextLabs Log Information in SAP

The following NextLabs events are tracked by SAP logging:

- Export of classification data from the Security Classification Maintenance table
- Read Tags
- UDRM features (Protect, Classify, Protect and Classify, Remove Protection)
- ADRM

You can use transaction SLG1 to view these logs for troubleshooting or administrative procedures.

Procedure

1. In the SAP interface, enter transaction SLG1. The Analyze Application Log screen appears.
2. To target NextLabs log information, enter /NXLEDRM/ in the Object field.
3. Specify other filter criteria for the event you want to view in the log.
4 Click **Execute**. The logging events display.
Chapter 6: Administration and Maintenance

Figure 6-2: Display Logs for /NXLEDRM/ Object

Configuration and Management

No configuration changes are required for running enforcers once they have been installed. There are some configuration controls available through Administrator, and there are some minimal management activities that enforcer administrative personnel may need to perform from time to time.

Configuration Tools

The only configuration settings available for enforcers are the enforcer profile settings, and the log file management settings. For details, see Managing Enforcer Policies on page 146 and Managing Event Logging on page 146.

Management Activities

Because enforcers are designed to run continuously with no complications, there are very few management functions required from system administrators. These include stopping enforcers that are currently running, restarting enforcers after they have been stopped, and monitoring the status of currently running enforcers.
Stopping and Starting Entitlement Managers

Because Entitlement Managers are designed to resist tampering, no user can stop them through the standard services manager, the Windows Task Manager, changing registry settings, or any other standard Windows procedure. Entitlement Managers can be stopped only with a special, password-protected utility.

As an additional tamper-resistance measure, no user can even view the contents of an Entitlement Manager’s installation directory while the Entitlement Manager is running. This means that you must stop the Entitlement Manager before you can perform certain tasks, such as examining log files in the logs directory.

Administrators can manually stop each individual Entitlement Manager at the local host using a utility called StopEnforcer.exe, installed on each enforcer host at C:\Program Files\NextLabs\Policy Controller\public_bin. You can run this utility from a link under Start > All Programs > NextLabs. This utility requires the administrative password set for whatever profile is assigned to the Entitlement Manager. This means you need to know which profile is in use, and what its password is, before you can stop an Entitlement Manager.

After you stop an Entitlement Manager with this utility, you can restart it again manually with the standard Services manager in Windows Control Panel, or by rebooting the host.

Monitoring Enforcers

You can monitor the status of all enforcers in the network—on file servers and on PCs—by opening Administrator, going to the Status tab, and clicking the Policy Enforcer Status link. By default, this tab displays the status of all enforcers in the system that are displaying warnings. To show all enforcers with or without warnings, uncheck the Enforcers with Warnings Only checkbox (A). To view only desktop server enforcers, select All Desktop Enforcers from the Show combo-box list at the left (B). If you are interested in the status of enforcers on a specific enforcer host or host group, you can also filter by host name by typing it into the Search By Host field (C) and clicking the Search button.

![Figure 6-3: Monitoring the Status of Enforcers](image-url)
Descriptions of the contents of the enforcer status grid are provided in Table 6-2.

Table 6-2: Information on Policy Enforcer Status

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Status</td>
<td>Indicates the current status of this enforcer, which may be either of the following:  &lt;br&gt;Green light = Clear: the policy enforcer is sending normal heartbeats.  &lt;br&gt;Exclamation point = Warning: the policy enforcer has not sent a heartbeat in the last 24 hours.</td>
</tr>
<tr>
<td>Host</td>
<td>Name of the machine where the policy enforcer is installed.</td>
</tr>
<tr>
<td>Type</td>
<td>Indicates the policy enforcer type: File Server Enforcer or Desktop Enforcer.</td>
</tr>
<tr>
<td>Last Heartbeat</td>
<td>Time stamp of the last heartbeat generated by this policy enforcer. If the policy enforcer is running normally, this time should correspond to the configured heartbeat interval. However, keep in mind that this does not necessarily indicate a problem, since certain policy enforcers—in particular, those on laptop computers used by remote personnel or computers that are turned off when not in use—might not be able to send a heartbeat for an extended period of time even though they are operating normally.</td>
</tr>
<tr>
<td>Last Policy Update</td>
<td>Tells when a new or modified policy or policy component was last deployed to this policy enforcer.</td>
</tr>
<tr>
<td>Policy Up to Date</td>
<td>A check mark appears if the policy enforcer has received the latest version of the policies that are targeted for deployment to it.</td>
</tr>
<tr>
<td>Profile Name</td>
<td>Tells which policy enforcer profile is assigned to this host. This profile determines behavior such as logging and heartbeat frequency.</td>
</tr>
<tr>
<td>Hide</td>
<td>Click to remove this host from the display. This is useful when the policy enforcer software has been uninstalled, and you therefore no longer need to monitor that host. If a policy enforcer is ever reinstalled on this host, the host will reappear on the list. If you click Hide by mistake on a host with an active policy enforcer, it will reappear automatically the next time the policy enforcer sends a heartbeat.</td>
</tr>
</tbody>
</table>

Uninstalling, Repairing, or Modifying Policy Controllers and Enforcers

Policy Controllers and enforcers use the standard Repair/Remove procedure common to all Windows applications. Because of the Policy Controller’s tamper-resistance features, an authorized administrator must first stop it before removing it. It is recommend that the same Administrator remove the Policy Controller as installed it.

Policy Controllers and enforcers are listed separately in the Add/Remove Programs list, and the process for uninstalling, repairing, or modifying is common to all Policy Controllers.

**Important**: You must uninstall the enforcer first, and the Policy Controller second. The sequence must not be reversed.

**Procedure**

1. Open the Windows Control Panel.
2. Launch the Add or Remove Programs utility.
3. Select the enforcer.
4. Click the Change button.
When this wizard runs for an enforcer, it offers the following options:

- **Modify**: Available only for the Policy Controller; enables you to reset any of the options that were selected during the initial installation.
- **Repair**: Runs an automatic diagnostic procedure that detects and restores any missing or damaged system files, for the currently installed version.
- **Remove**: Uninstalls the current Policy Controller. You use this option when you want to remove the enforcer for any reason—for example, to upgrade to a new version.

### About Service Account Permissions

All enforcers run as Windows services, and are assigned a default user account at installation. This account must have read, execute and create permission for the folder where the enforcer is installed. For example, by default the Windows Desktop Enforcer is installed in the C:\Program Files\NextLabs\Desktop Enforcer folder, and is assigned the Local System user account, as shown in Figure 6-4.

![Figure 6-4: Enforcer Service User Accounts](image)

You must take care not to change the permission levels of the install directory, wherever it may be, or of the services user accounts, in such a way that the account does not have read, execute and create permission for the installation folder. If you do this, the account will not be able to access the service, and it will not restart automatically if it ever stops. For example, if you change the Local System user permission to extend only to Program Files without any child directories, the service will not be able to restart.

### About Bundle Encryption

The Policy Server continuously updates enforcers, of all kinds, with any newly-defined or modified policies relevant to them. Each enforcer periodically sends a heartbeat message to the
ICENet Server, which then checks whether any new or changed policies are in queue to be sent to that enforcer. If there are, it sends them, in the form of a file called bundle.bin; this file is referred to as a policy bundle. Each enforcer only retains one bundle at a time, which contains its most recent enforcement instructions. If the Policy Server sends another bundle with updated instructions—additional new policies to enforce, for example—that will overwrite the previous bundle.

All policy bundles sent from the Policy Server to enforcers are encrypted using standard SSL protocols. When they arrive at the enforcer, the enforcer authenticates them with digital certificates to ensure that they were indeed created by the Policy Server, and that they have not been modified by any other processes. This protects against the possibility of anyone deploying spoof policies designed to open security holes in your enterprise.

**Authentication Failure**

Whenever a bundle file arrives at an enforcer client and cannot be authenticated, a Level 3 document activity event is written to the Windows Event Log:

```
policy bundle authentication failed
```

This event will also be displayed by Reporter, if your query includes Level 3 events. The most likely cause of such failure is that the file is corrupted in some way; in such cases you should examine the file contents.

**Decrypting the Bundle: Policy Controller**

Bundle files are encrypted, but administrators can decrypt them for troubleshooting purposes. For this purpose, a special utility called Decrypt.exe is available in the public\_bin directory of each host where an enforcer is installed.

**Procedure**

1. Stop the enforcer on the host where the encrypted bundle file is located. You cannot decrypt any bundles while the enforcer is running.

2. Open a command prompt and run the utility, supplying the arguments shown below.

   ```
   decrypt -b <path\file> -f <OutputFile.txt> -e <InstallPath>
   ```

   In this command:

   - `-b` is the complete path and name of the encrypted bundle file. By default, the path is C:\Program Files\NextLabs\Desktop Enforcer. The file name will always be bundle.bin. This argument is not mandatory; if it is not present, the enforcer will assume the default path and file name.
   - `-f` is the name of the output (decrypted) file. This argument is also not mandatory; if it is not present, the utility will call the output file bundle.out, and place it in the same path as bundle.bin.
   - `-e` is the actual installation directory for the enforcer, which the utility uses to load the security keys from the keystore on the file system. If the enforcer is installed in this default path (C:\Program Files\NextLabs\Desktop Enforcer), and the Decrypt utility is running from its default path, this argument is not needed.
For example:

```
decrypt -b "C:\Program Files\Info Security\Entitlement Manager 7.7 for SAP\Windows Desktop Enforcer\bundle.bin" -f bundle.txt -e "C:\Program Files\Info Security\Entitlement Manager 7.7 for SAP\Windows Desktop Enforcer"
```

If you are using default values for install paths and file names, no arguments are required, and you can use the following command:

```
decrypt
```

If the Decrypt utility has been moved to a non-default location, only the `-e` argument is required.

To display the utility’s help screen, type the following command:

```
decrypt -h
```

3. When the utility starts, it prompts you for the standard utility password, which is the same as the password required to stop the enforcer.

After the utility runs, the output text file is available for analysis.

**Decrypting the Bundle: Java Policy Controller**

As discussed in more detail in NextLabs Enforcer User’s Guides, the Policy Server continuously updates enforcers, of all kinds, with any newly-defined or modified policies relevant to them. Each enforcer periodically sends a heartbeat message to the ICENet Server, which then checks whether any new or changed policies are in queue to be sent to that enforcer. If there are, it sends them, in the form of a file called `bundle.bin`; this file is referred to as a policy bundle. Each enforcer only retains one bundle at a time, which contains its most recent enforcement instructions. If the Policy Server sends another bundle with updated instructions—additional new policies to enforce, for example—that will overwrite the previous bundle.

All policy bundles sent from the Policy Server to enforcers are encrypted using standard SSL protocols. When they arrive at the enforcer, the enforcer authenticates them with digital certificates to ensure that they were indeed created by the Policy Server, and that they have not been modified by any other processes. This protects against the possibility of anyone deploying spoof policies designed to open security holes in your enterprise.

Bundle files are encrypted, but administrators can decrypt them for troubleshooting purposes. For this purpose, a special utility is included with the PolicyControllerJava.zip installation file. `Decrypt.bat` (for Windows) and `decrypt.sh` (for Solaris) are available in `<tomcat-home>\nextlabs\dpc` folder.

**Note:** To run this utility on Solaris, users must have executable permission for the file (`decrypt.sh`).

**Procedure**

1. Run the appropriate utility, depending on your installation environment:
   - For Windows, enter `decrypt` in the command prompt.
• For Solaris, enter "./decrypt.sh" in the command shell.

2. When the utility starts, it prompts you for the standard utility password, which is the same as the password required to stop the enforcer.

3. After the utility runs, the output text file, bundle.out, exists in the same folder as the decrypt utility.

Managing Enforcer Policies

All policy enforcers are governed by a number of configuration settings that control such aspects as logging behavior, heartbeat rates, tamper-prevention passwords, and network configuration. These are assigned default values when you first install an enforcer, but they can be changed manually at any time. To simplify this, Administrator enables you to create named sets of configuration settings, which you can then assign to one or more enforcers in your network. These are referred to as enforcer profiles, and you manage them in Administrator, with the settings on the Policy Enforcer Configuration tab.

The settings controlled by enforcer profiles include the following:

• Which ICENet Server the enforcer will use to communicate with the Policy Server
• How often the enforcer sends heartbeat signals to the Policy Server to indicate it is operating normally

For more details on defining and using enforcer profiles, see the NextLabs Control Center Administrator’s Guide.

Managing Event Logging

All enforcers maintain a set of local log files, which can be useful for troubleshooting or when communicating with NextLabs Technical Support. Both kinds of enforcers save their log files at

<InstallDir>\agentLog

Because of the tamper-resistance features, you must stop any enforcer before you can view or open its log files.
Enforcers maintain a log file called Agent0.log.0 until the file reaches its specified maximum size, the file is saved as Agent0.log.1, and current logging continues in the original file name. Every time the file reaches its maximum size it is closed and saved, and the ending integer in all existing files is incremented. That is, the file ending in 0 will always contain the latest information, and the one ending in the highest integer will contain the oldest information.

Logging Settings

You can configure the limit on the number and size of log files each enforcer maintains by editing the file logging.properties in the following directory:

<InstallDir>\config

The following properties control the number and size of log files maintained by each policy enforcer:

- **java.util.logging.FileHandler.count**: Specifies the maximum number of log files that can be archived at any given time. When this maximum is reached, the oldest file is discarded so that a new file can be started. Default = 10.
- **java.util.logging.FileHandler.limit**: Specifies the maximum size of each log file, in bytes. When this limit is reached, the current file is archived and a new log file is started. Default = 500K.

Changing Logging Levels

You can configure enforcers to the following levels of event logging, in order of increasing verbosity:

- Severe
- Warning
- Info
- Fine
- Finest

By default, the logging level is set to Severe, but if you wish you can change this individually for each policy enforcer. It is controlled by three parameters in the logging.properties file:

- **java.util.logging.ConsoleHandler.level**
- **com.bluejungle.level**
- **.level**
You should always set `java.util.logging.ConsoleHandler.level` and `com.bluejungle.level` to the same value. The `.level` parameter represents a default level that will apply when no level has been defined for a logging component. Figure 6-5 shows an example of this file.

Figure 6-5: Changing Enforcer Logging Levels

Load Balancing the Policy Controller

The Policy Controller is the component of the NextLabs system that receives policy evaluation queries from the SAP Agent, evaluates them, and returns a policy decision. Customers may choose to load balance this component to enhance system performance or implement redundancy in their system.
NextLabs supports load balancing for the following configurations:

- Server Policy Controller on Windows with the Java Connector
- Java Policy Controller on Windows with the Java Connector
- Java Policy Controller on Solaris with the Java Connector

There is no special configuration required for on the Policy Controller side. You only need to follow the standard procedures to install the Policy Controller on multiple hosts, then configure the Java Connector to point to the SAP server, following the procedures appropriate for the operating system.

**Note:** For more information on how to install the Policy Controller, see Installing the Policy Controller.  
For more information on how to install and configure the Java Connector, see Installing the Java Connector for the Policy Controller.

On the SAP server side, you must use the load balancing configuration options provided in the SAP Gateway Monitor (transaction SMGW). There are several options for how to configure load balancing in SMGW. The example instructions below explain one configuration.

**Note:** For more information on other load balancing configurations in SAP, consult SAP documentation.

### Example: Load Balancing Configuration

The following procedure provides an example configuration for load balancing the Policy Controller using properties available in SAP Gateway Monitor.

**Procedure**

1. Install and configure the Policy Controllers and Java Connectors on host to be load balanced. The Java SDK Properties files on both hosts should point to the same SAP server.

2. In the SAP interface, enter transaction **SMGW**. The SAP Gateway Monitor appears.
3 Select **Goto > Parameters > Change**.

![Figure 6-7: Changing Gateway Connection Parameters](image)

4 There are several Gateway parameters that can be used to configure load balancing. In this example, we use the gw/reg_1b_level = 1 parameter for load balancing. In the level parameter, the following values are possible:

- 0: No load balancing; the first free registered program is used.
- 1: The program with the lowest counter is used. Every time a registered program is assigned a request, the counter is increased by 1.
- 2: The program with the lowest load is used. The load is determined as defined by profile parameter: gw/reg_1b_ip.
**Figure 6-8: Using the gw/reg_lb_level in the SAP Gateway Monitor Parameters for Load Balancing**

**Note:** For more information on how to configure other parameters for load balancing in this screen, consult SAP documentation.

5. Save the Parameter change.
Deleting Decrypted Temporary Documents

Administrators can schedule a batch job to run at specified intervals to delete all decrypted temporary documents present in the temporary location.

This example procedure lists the steps to define a background job to delete all decrypted files available in the temporary location.

Procedure

1. In the SAP interface, enter the SM36 transaction code. The Define Background Job screen appears.

2. In the Job name field, enter a name for the job (for example, DECRYPT_FILE). Make sure that the Status is Scheduled.

3. Set a priority for the job, or Job class.
   - A: High priority
   - B: Medium priority
   - C: Low priority

4. Click Step on the application tool bar. The Create Step 1 screen appears.

5. In the ABAP program section, in the Name field, enter: /NXLEDRM/DELETE_DECRYPT_FILE

6. Click Check and then click Save. The Step List Overview screen appears.

7. Click Back to return to the Define Background Job screen and then click Save. A message appears stating that the DECRYPT_FILE job has been saved with status: Scheduled.

8. Click Start Condition in the tool bar. A pop-up window appears, prompting you to schedule the job. Schedule the start Date and Time conditions, or click Immediate to start the job at once. You can also schedule the job to run repeatedly by clicking on the Date/Time button and specifying the frequency.

9. Click Check and Save.

10. After the pop-up disappears, make sure to click Save to save the job definition in the original screen. This is very important. If the job definition is not saved in the Define Background Job screen, the job is not scheduled. After you save the job, a message appears at the bottom of the screen in the status bar displaying the job saved or released message.

11. To verify the job definition, enter the SM37 transaction code and click Execute. Make sure the username in the selection screen is the same as the one used to define the job. The jobs scheduled or released by this user are displayed.
This section explains the custom enhancements available for the NextLabs Digital Rights Management for SAP.

Topics:

- Dynamic User and Resource Attributes
- BADI Implementation for Dynamic User or Resource Attribute

The procedures in this section reference standard ABAP procedures and assume that users have ABAP development expertise. For more information on how to perform ABAP procedures, see the SAP documentation.

**Note:** The only modifications supported for the NextLabs namespace are officially released NextLabs product code. Customers should not store other modifications to code in the NextLabs namespace because it can result in installation and upgrade issues.

### Dynamic User and Resource Attributes

There may be use cases where customers want to enforce controls on user or resource attributes that are not (or cannot be) enrolled into the NextLabs Control Center. For example, a customer may need to enforce access controls based on physical location or another attribute that can vary dynamically.

To address this requirement, EDRM for SAP supports the implementation of dynamic user and resource attributes. ABAP developers can insert dynamic look-ups into policy checks to retrieve user and resource information from a designated store at the point of policy evaluation. The mechanism that collects and stores the user or resource attribute are to be determined and developed by the customer. For example, you could build a program to automatically gather a user attribute (such as physical computer location) and store it as a session variable. Or, you could create a user interface that prompts users to enter attributes.

This section provides information about how to configure the BADI method to collect these attributes from the storage location you designate. It does not provide guidelines for the solution you develop to collect and store those attributes (which are dependent on the particulars of your environment and business requirements).
Configuring Enhancement Activations for Dynamic Attributes

This section provides step-by-step instructions for using the BADI Definition for dynamic user and resource attributes, including creating the implementation, the supported methods, and the structure for parameters for the method.

Procedure

1. In the SAP interface, enter transaction SE19. The Business Add-Ins screen appears.
2. Select Classic BADI and enter the BADI Name /NEXTLABS/ENH_DYNATT. Click Create Implementation.
3. Enter an Implementation Name.
4. In Implementation Short Text, enter a description.
5. Click the Interface tab.

![Business Add-In Builder: Change Implementation ZUSERLOCATIONDYNATT](image)

Figure 7-1: Methods for /NEXTLABS/ENH/DYNATT
6 GET and SET methods are available, as described in table Table 7-1.

Table 7-1: Available Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET_DYN_USER_ATT</td>
<td>Retrieves user attributes written to a storage location (defined by the customer). This is used to pass user attributes to the requesting policy check.</td>
</tr>
<tr>
<td>SET_DYN_USER_ATT</td>
<td>Writes user attributes to a storage location (defined by the customer).</td>
</tr>
<tr>
<td>GET_DYN_RESOURCE_ATT</td>
<td>Retrieves resource attributes written to a storage location (defined by the customer). This is used to pass resource attributes to the requesting policy check.</td>
</tr>
<tr>
<td>SET_DYN_RESOURCE_ATT</td>
<td>Writes resource attributes to a storage location (defined by the customer).</td>
</tr>
</tbody>
</table>

7 Select the method for which to write code, then enter the code. Refer to Table 7-2 for supported parameters.

Table 7-2: Parameters for GET and SET methods

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>I_USER_ID</td>
<td>User ID (Import parameter)</td>
</tr>
<tr>
<td>I_S_GENERIC_ATTR</td>
<td>Generic attributes for transaction interception (Import parameter)</td>
</tr>
<tr>
<td></td>
<td>• IPADD: IP Address</td>
</tr>
<tr>
<td></td>
<td>• ADDON: NextLabs Add on</td>
</tr>
<tr>
<td></td>
<td>• Action: NextLabs Action name</td>
</tr>
<tr>
<td></td>
<td>• ACC_HIER_T: Access Control Context hierarchy</td>
</tr>
<tr>
<td></td>
<td>• PBSC_ATTR: PBSC Attributes</td>
</tr>
<tr>
<td></td>
<td>• SECENH: Composite key information</td>
</tr>
<tr>
<td></td>
<td>• FILTER: Bypass message processing flag</td>
</tr>
<tr>
<td></td>
<td>• USERID: User ID</td>
</tr>
<tr>
<td>I_S_TCODE</td>
<td>Transaction code or UI function code (Import parameter)</td>
</tr>
<tr>
<td>I_S_SECIDT</td>
<td>Security identifiers work area (Import parameter)</td>
</tr>
<tr>
<td>E_USER_ATTR_T</td>
<td>User attribute table (Export parameter)</td>
</tr>
<tr>
<td>E_RESOURCE_ATTR_T</td>
<td>Resource attribute table (Export parameter)</td>
</tr>
</tbody>
</table>

The following sample code in GET_DYN_USER_ATT shows an example implementation.

```plaintext
**Start
**Attribute name - Location
s_attrss-key = 'Location'
Call Function 'ZGet_Location'
Importing
  i_userID = syuname
Exporting
  e_location = s_attrss_value
```
APPEND s_attrs TO e_user_attr_t.

The following example code in GET_DYN_RESOURCE_ATT sends all plant data from the material being accessed by a user as dynamic resource attributes.

**Start
** Attribute name - Plant
s_attr-key = 'Plant'.

** Get Attributes values (sample code)
SELECT werks FROM marc INTO nxl_v_werks
WHERE matnr = i_s_secidt-matnr.
  s_attr-value = nxl_v_werks

APPEND s_attr TO e_resource_attr_t.
ENDSELECT

8 After you have finished entering code, Save and Activate the method.

9 Save and Activate the Implementation.

Referencing Dynamic Attributes in Policies

Dynamic attributes can be referenced the same way as any attribute in a Policy Studio policy. Reference the User Key and Value in the Properties field of the User components.

BADI Implementation for Dynamic User or Resource Attribute

A typical use case to use this implementation would be to enforce control on user or resource attributes that are not enrolled or that cannot be enrolled into the NextLabs Control Center.

For example, you might need to enforce access controls based on physical location or another attribute that can vary dynamically. ABAP developers can insert dynamic look-ups into policy checks to retrieve user and resource information from a designated store at the point of policy evaluation. The mechanisms that collect and store the user or resource attributes must be determined and developed by the customer. For example, you can write a program to automatically gather user or resource attributes (such as physical computer location) and store it as a session variable. Or, you could create a user interface that prompts users to enter attributes.

This section provides information about how to configure the BADI method to collect these attributes from the storage location you designate. It does not provide guidelines for the solution you develop to collect and store those attributes, which are dependent on the particulars of your environment and business requirements.

Note: The only modifications supported for the NextLabs namespace are officially released NextLabs product code. Do not store other modifications to code in the NextLabs namespace because doing so can result in installation and upgrade issues.
Procedure

1. In the SAP interface, enter transaction SE19. The Business Add-Ins screen appears.

2. In the Create Implementation section, select the Classic BAdI option, and enter the BAdI Name /NEXTLABS/ENH_DYNATT.

3. Click Create Impl.

4. Enter an implementation name.

5. In the Implementation Short Text field, enter a description, and then click OK.

6. Click the Interface tab. The following GET and SET methods are available:

   - GET_DYN_USER_ATT: Retrieves user attributes written to a storage location (defined by the customer). Used to pass user attributes to the requesting policy check.
   - SET_DYN_USER_ATT: Writes user attributes to a storage location (defined by the customer).
   - GET_DYN_RESOURCE_ATT: Retrieves resource attributes written to a storage location (defined by the customer). Used to pass resource attributes to the requesting policy check.
   - SET_DYN_RESOURCE_ATT: Writes resource attributes written to a storage location (defined by the customer).

7. Write code to retrieve attributes from a specified location, or to set attributes.

8. Save and Activate the Method.

9. Save and activate the BAdI implementation.