Privacy Manager Plug-in for GAX

eServices Digital Administration 9.0.0

4/4/2023

Table of Contents

Privacy Manager Plug-in for GAX

Privacy Manager is a Genesys Administrator Extension (GAX) plugin that allows the administrator to assign rules and actions to incoming text-based messages to protect private information.

For instance, the administrator can configure a rule that looks for an alphanumeric pattern used in a Social Security Number and mask the data completely or partially. The same can also be done for an account number, credit card number, phone number, and or any other custom alphanumeric string that follows a defined pattern.

This guide describes how to deploy, configure, and use Privacy Manager.

Deploying Privacy Manager

Prerequisites

- Genesys Administrator Extension (GAX) 9.0.100.xx or later.
- UCS 8.5.200.19 or later
- UCS Proxy 8.5.100.04 or later.
- Classification Server 8.5.200.05 or later
- If a previous version of Privacy Manager was installed on the current host, uninstall that version.

> **Important**
>
> One UCS instance can support multiple tenants in Privacy Manager, and each tenant's data is stored separately from other tenants. The graphic below shows two tenants (*Environment* and *Tenant_extra*) using one UCS instance (*ContactServer_852_2*). The *Environment* tenant has data that is not accessible in the *Tenant_extra* tenant.

Create a Zip File

1. Create a temporary folder on your desktop.
2. Copy the IP folder from the original location (typically, the product CD) into the temporary folder.
3. Copy the templates folder from the original location into the temporary folder.
4. Zip the temporary folder.

Alternatively, contact Genesys Customer Care to obtain the required Zip file containing the installation package and associated templates.

Installing the plugin

Local Control Agent 8.5.100.31 or higher

1. Upload the Privacy Manager installation package (IP) to GAX. Refer to the GAX documentation for more information.
2. Extract the IP to any folder.
3. Navigate to the **ip** folder in the extracted folder.
4. Do one of the following:
   - On Windows, run **setup.exe**.
   - On Linux, run **install.sh**.
5. Restart GAX.

Local Control Agent 8.5.100.29 or lower

1. Add the following option to the **Application Options** tab of the Genesys Administrator Extension (GAX) Application object on the host:
   Section name: **[asd]**

   Option name: **plugin_ip_list**

   Option value: `PrivacyMng64` (for the Windows host) or `PrivacyMng` (for the Linux host)

   > **Tip**
   > This option is also used by eServices Manager Plug-in for GAX and Content Analyzer Plug-in for GAX. If you are running both eServices Manager and Content Analyzer with Privacy Manager, the value can be a comma-separated list; for example, `eSMngrPlgnAdm,PrivacyMng,CntAnlzPlgnAdm`.

2. Restart GAX.
3. Upload the Privacy Manager installation package (IP). If you previously uploaded the IP, do so again.
4. Carry out the plug-in installation process. To uninstall Privacy Manager, use the general procedure for removing a plug-in.
5. Restart GAX.

Uninstalling the plugin

On Linux

1. Stop GAX.
2. Go to **<GAX_HOME>/webapp/WEB-INF/lib** on the file system (where **<GAX_HOME>** is your home folder for the GAX application).
3. Delete the **gax-privacy-manager-<$version$>.jar** file (where **<$version$>** is the version of the plugin).
4. Go to **<GAX_HOME>/plug-ins** on the file system.
5. Delete the **gax-privacy-manager-<$version$>.jar** file.
6. Go to **<GAX_HOME>/webapp/plugins** on the file system.
7. Delete the **gax-privacy-manager** folder.
8. Start GAX.

On Windows

1. Stop GAX.
2. Go to **Programs and Features**.
3. Find and run **Genesys Privacy Manager Plug-in for GAX <$version$>** (where **<$version$>** is the version of the plugin).
4. Select the **Remove** check box.
5. Click **Next**.
6. Click **Yes** in **Confirm Windows**.
7. Click **Finish**.
8. Go to **<GAX_HOME>/webapp/plugins** on the file system (where **<GAX_HOME>** is your home folder for the GAX application).
9. Delete the **gax-privacy-manager** folder.
10. Go to **<GAX_HOME>/plug-ins** on the file system.
11. Delete the **gax-privacy-manager-<$version$>.jar** file.
12. Start GAX.

Configuring Privacy Manager

Local Control Agent 8.5.100.31 or higher

In Genesys Administrator Extension,

1. Create a Privacy Manager Application with the type Application Cluster.
2. Add tenants to Privacy Manager Application.
3. Specify the Host and Port. The Application Cluster application object and the GAX application must be configured with the same host in order to enable mutual TLS connections.
4. Connect the Privacy Manager Application to UCS and Classification Server.
5. Configure these connections: TLS, ADDP, and so on.
6. Navigate to the GAX Application Options for the Privacy Manager Application.
7. Create a section with the name `gax-privacy-manager`.
8. Create the following options in the **gax-privacy-manager** section:
   - `app-name` = The Privacy Manager Application name.
   - `file-storage-path` = Full path to the folder in which Privacy Manager creates its temporary file. You must ensure that Privacy Manager has permission to write into that directory.
   - `connection-timeout` = 3
9. If you need multi-language support, set the UTF-8 JVM parameter for all Java components (UCS, GAX Server, Browser) in the corresponding .ini file (or .bat file if you start the component from command line):
   - ini file: `[JavaArgs] -Dfile.encoding=UTF-8 ...`
   - bat file: `set JAVA_OPTS=%JAVA_OPTS% -Dfile.encoding=UTF-8 ...`
10. Restart GAX.

Local Control Agent 8.5.100.29 or lower

In Genesys Administrator Extension,

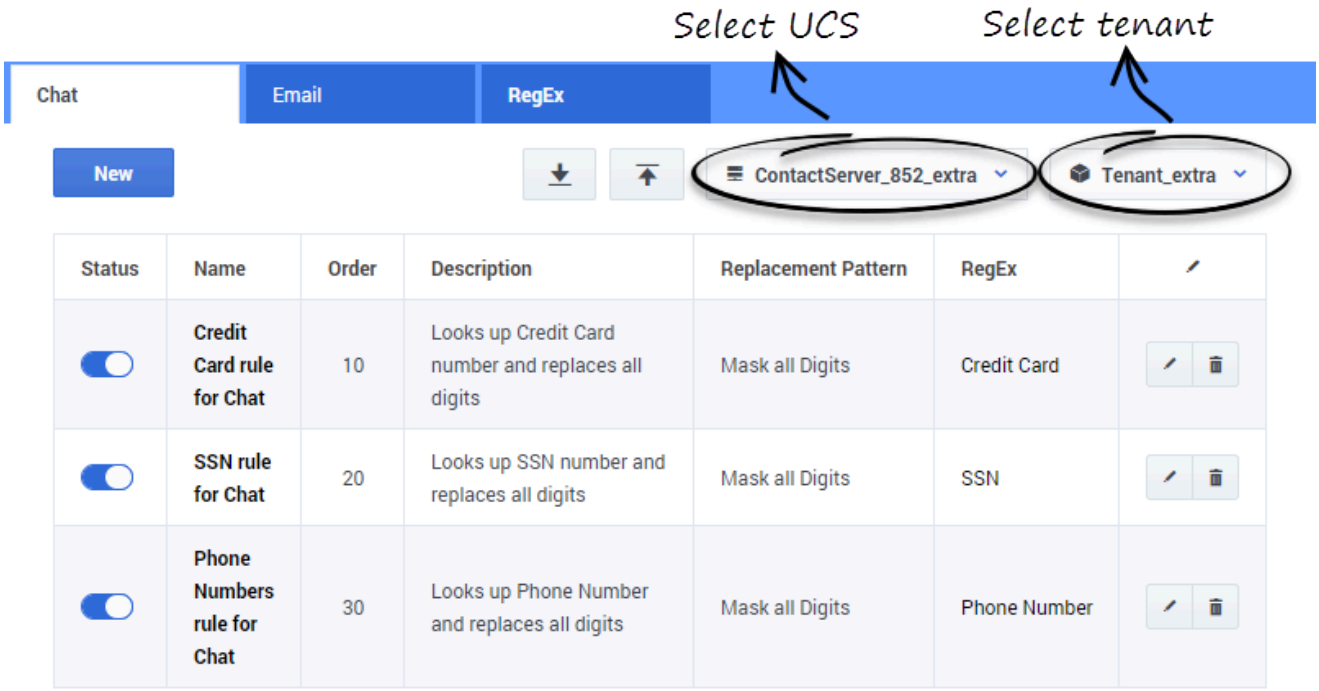
1. Create a Privacy Manager Application with the type Application Cluster.
2. Add tenants to Privacy Manager Application.
3. Specify the Host and Port. The Application Cluster application object and the GAX application must be configured with the same host in order to enable mutual TLS connections.
4. Connect the Privacy Manager Application to UCS and Classification Server.

5. Configure these connections: TLS, ADDP, and so on.
6. Navigate to the GAX Application object's **gax-privacy-manager** section and configure the following options:
   - **app-name** = Specify the Privacy Manager Application name.
   - **file-storage-path** = Specify the full path to the folder in which Privacy Manager creates its temporary file. You must ensure that Privacy Manager has permission to write into that directory.
7. If you need multi-language support, set the UTF-8 JVM parameter for all Java components (UCS, GAX Server, Browser) in the corresponding .ini file (or .bat file if you start the component from command line):
   - ini file: `[JavaArgs] -Dfile.encoding=UTF-8 ...`
   - bat file: `set JAVA_OPTS=%JAVA_OPTS% -Dfile.encoding=UTF-8 ...`
8. Restart GAX.

Using Privacy Manager

Starting

1. In GAX Configuration Manager, go to **eServices** and select **Privacy Manager**.
2. Do one of the following steps:
   a. For 8.5.1 releases and lower, select a UCS. The main Privacy Manager window displays.
   b. For 8.5.3 releases and higher, you can select a UCS and tenant to use with Privacy Manager.



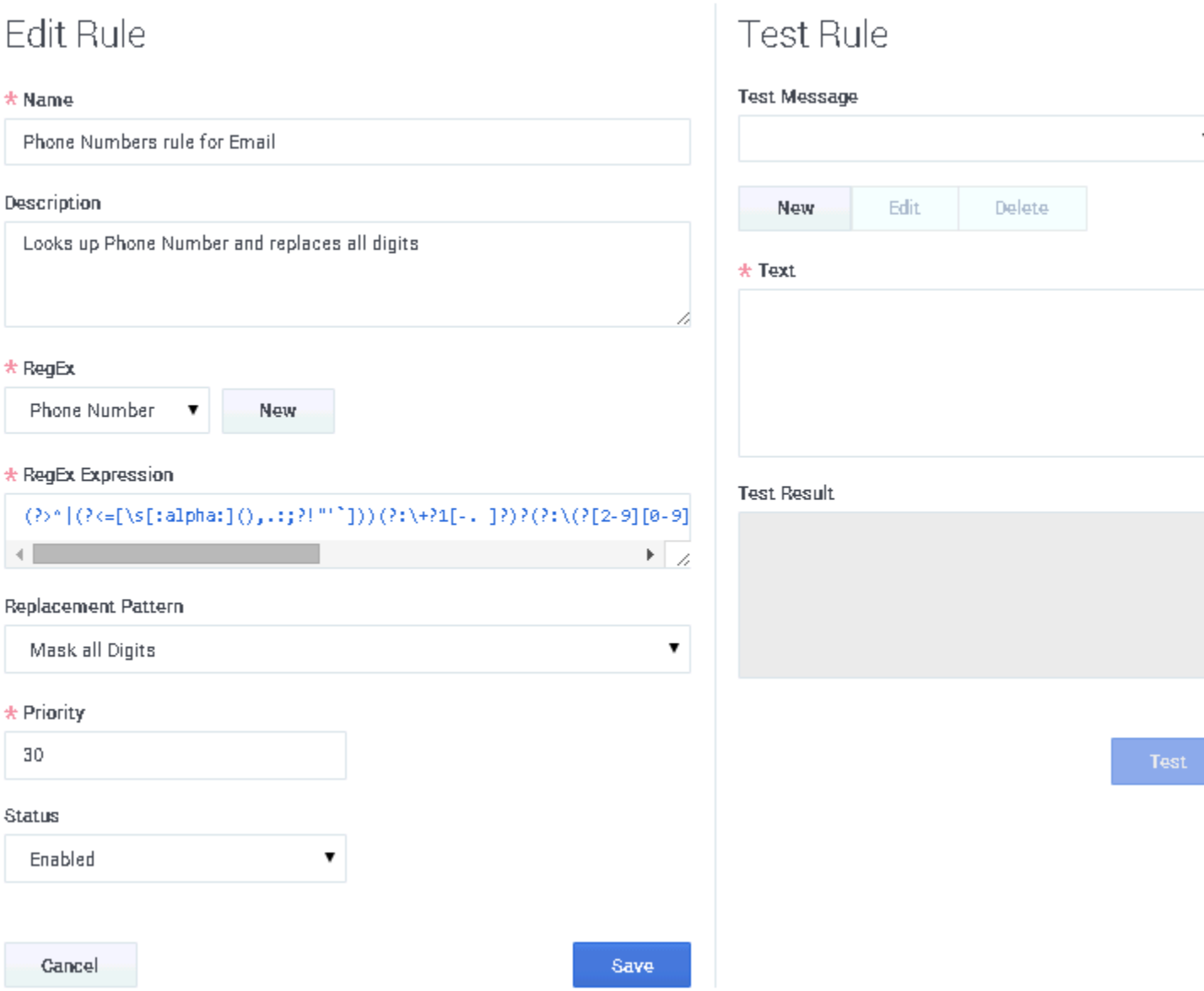
Tip

Privacy Manager is designed to be viewed at a minimum screen resolution of 1280x1024, or a full HD resolution of 1920x1080.

Creating or Editing a Rule

- To create a rule, click **New**.
- To edit an existing rule, click the rule name or the Edit icon.

The window that opens is titled either **New Rule/Test Rule** or **Edit Rule/Test Rule**, but in either case the fields and functionality are identical.



New Rule, Edit Rule, Test Rule

Note the following:

- For **RegEx**, you can either select one of the ready-made regular expressions or open the RegEx Create/Edit window to create a new one.
- For **Priority**, 1 is the first (highest) priority, 2 is next, and so on.

> **Important**
>
> Although it is possible to assign the same priority to multiple rules in the same group, Genesys recommends against doing so.

Testing a Rule

The right-hand pane of the edit/create window allows you to test a rule.

- **Test message** is text that you will try applying the rule to. You can select a ready-made test message or write your own.
- Click **Test** to run the rule on the test message. The results appear in **Test Result**.

Creating, Editing, and Testing RegExes

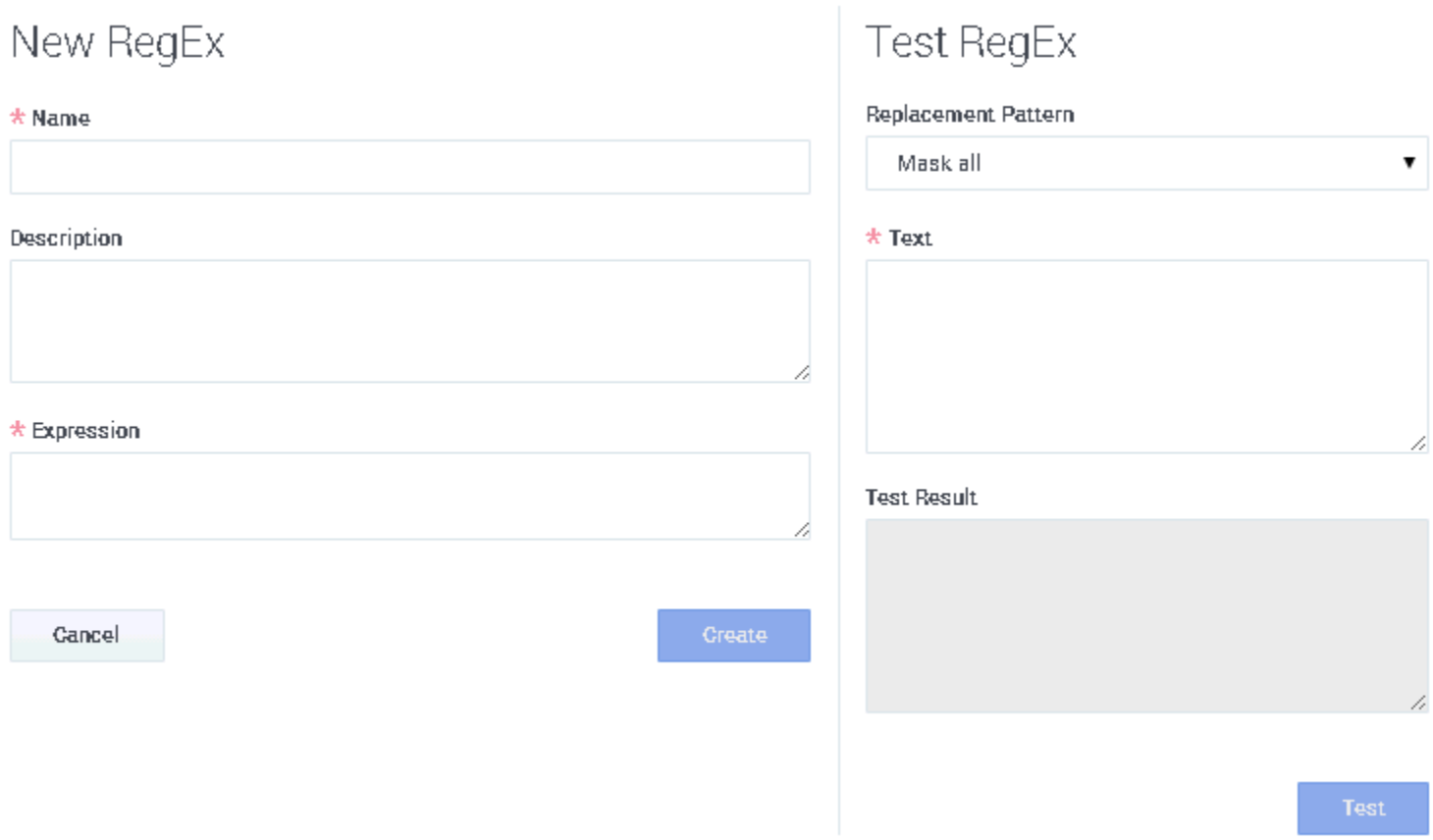
You can use Privacy Manager to write your own rules and test them, but Genesys also provides hard-coded rules that use the following regular expressions:

Name	Regular Expression
Credit Card (Visa and MasterCard only)	`(?:^|(?<=[\D;a-zA-Z(),.:;?!"'`>]))(?:4\d{3}|5[1-5]\d{2}|6011|622[1-9]|64[4-9]\d|65\d{2})[-.=\n\r]{0,10}\d{4}[-.=\n\r]{0,10}\d{4}[-.=\n\r]{0,10}\d{4}(?:$|(?=[\Da-zA-Z(),.:;?!"'`<= ]))`
Phone Number (North America)	`(?:^|(?<=[\D;a-zA-Z(),.:;?!"'`>]))(?:\+?1[-.=\n\r]{0,10})?(?:\(?[2-9][0-9]{2}\)?[-.=\n\r]{0,10})?[2-9][0-9]{2}[-.=\n\r]{0,10}[0-9]{4}(?:$|(?=[\Da-zA-Z(),.:;?!"'`<= ]))`
SSN (Social Security Number - U.S. only)	`(?:^|(?<=[\D;a-zA-Z(),.:;?!"'`>]))(?!000|666|9)\d{3}[-.=\n\r]{0,10}(?!00)\d{2}[-.=\n\r]{0,10}(?!0000)\d{4}(?:$|(?=[\Da-zA-Z(),.:;?!"'`<= ]))`

To create a RegEx,

- Click **New** on the RegEx tab, or
- Click **New** next to the RegEx field in the **New Rule/Edit Rule** window.

To edit a RegEx, click its Edit icon on the **RegEx** tab.



New RegEx/Test RegEx

Testing RegExes works in the same way as testing rules.

Importing and exporting rules

Important

- This feature is only available in 8.5.3 releases and higher.
- Your account must have sufficient role privileges to use this feature. Refer to the Role-Based Access Control section for more information.

You can export and import rule sets as JSON files. This is useful if you want to copy a rule set to another instance of Privacy Manager.

To export a rule set, click  . Privacy Manager prepares the JSON file for export. Click **Download** to save the JSON file to your computer.

To import a rule set, click  . Select a JSON file and click **Import**.

Privacy Manager does not overwrite existing rules when you import a rule set. For example, if you import a rule set and a rule name matches one that already exists, Privacy Manager appends **Copy_1_of_** to the imported rule name (for example, **Credit Card rule for Chat** becomes **Copy_1_of_Credit Card rule for Chat**).

Limitations

Genesys recommends that you observe the following limits:

- RegEx: no more than 400 per Tenant
- Test Messages per RegEx: no more than 20
- Rules per Group: no more than 200

Role-Based Access Control

Privacy Manager uses role privileges to control user access to various functions.

Roles in Privacy Manager

To grant a user access to the GAX menu and Privacy Manager Plug-in for GAX, assign the following privileges to the users in GAX:

- **COM/Access Configmanager**—Allows user to access Configuration Manager in GAX.
- **gax-privacy-manager/Privacy Manager Plug-in for GAX Access**—Allows user to access Privacy Manager.

In Privacy Manager 8.5.3 and later, you can also assign the following **gax-privacy-manager** role privileges to users:

- **Import/Export Access**—Allows user to access the import and export buttons.
- **Regex Access Create**—Allows user to create RegEx rules.
- **Regex Access Delete**—Allows user to delete RegEx rules.
- **Regex Access Edit**—Allows user to edit RegEx rules.
- **Regex Access**—Allows user to access the RegEx tab.
- **Rule Access Create**—Allows user to create chat and email rules.
- **Rule Access Delete**—Allows user to delete chat and email rules.
- **Rule Access Edit**—Allows user to edit chat and email rules.

You might need a combination of role privileges to accomplish certain tasks. See the tables below for examples.

Import and Export

Task	Required role privileges
Export	• **Import/Export Access** • **Regex Access**
Import	• **Import/Export Access** • **Rule Access Create** • **Regex Access**

Task	Required role privileges
	• **Regex Access Create**

Test rules with ability to create new tests

Task	Required role privileges

Test new chat/email rule	• **Rule Access Create**
	• **Regex Access**
	• At least one of the following:
	  • **Regex Access Create**
	  • **Regex Access Edit**

Test existing chat/email rule	• **Rule Access Edit**
	• **Regex Access**
	• At least one of the following:
	  • **Regex Access Create**
	  • **Regex Access Edit**

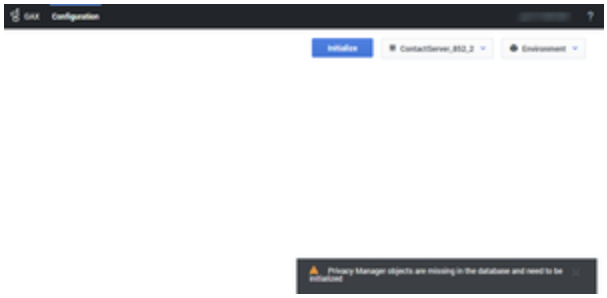
Test new RegEx rule	• **Regex Access** • **Regex Access Create**
Test existing RegEx rule	• **Regex Access** • **Regex Access Edit**

Test chat/email rules without ability to create new tests

Task	Required role privileges
Test new chat/email rule	**Rule Access Create**
Test existing chat/email rule	**Rule Access Edit**

Initialize a UCS database

You must initialize a UCS database if it has not been previously used with Privacy Manager. If the UCS database has not been initialized, Privacy Manager displays a warning message:



You can initialize the database by clicking **Initialize**. However, to access the **Initialize** button, you must have the following role privileges:

- **Rule Access Create**
- **Regex Access**
- **Regex Access Create**

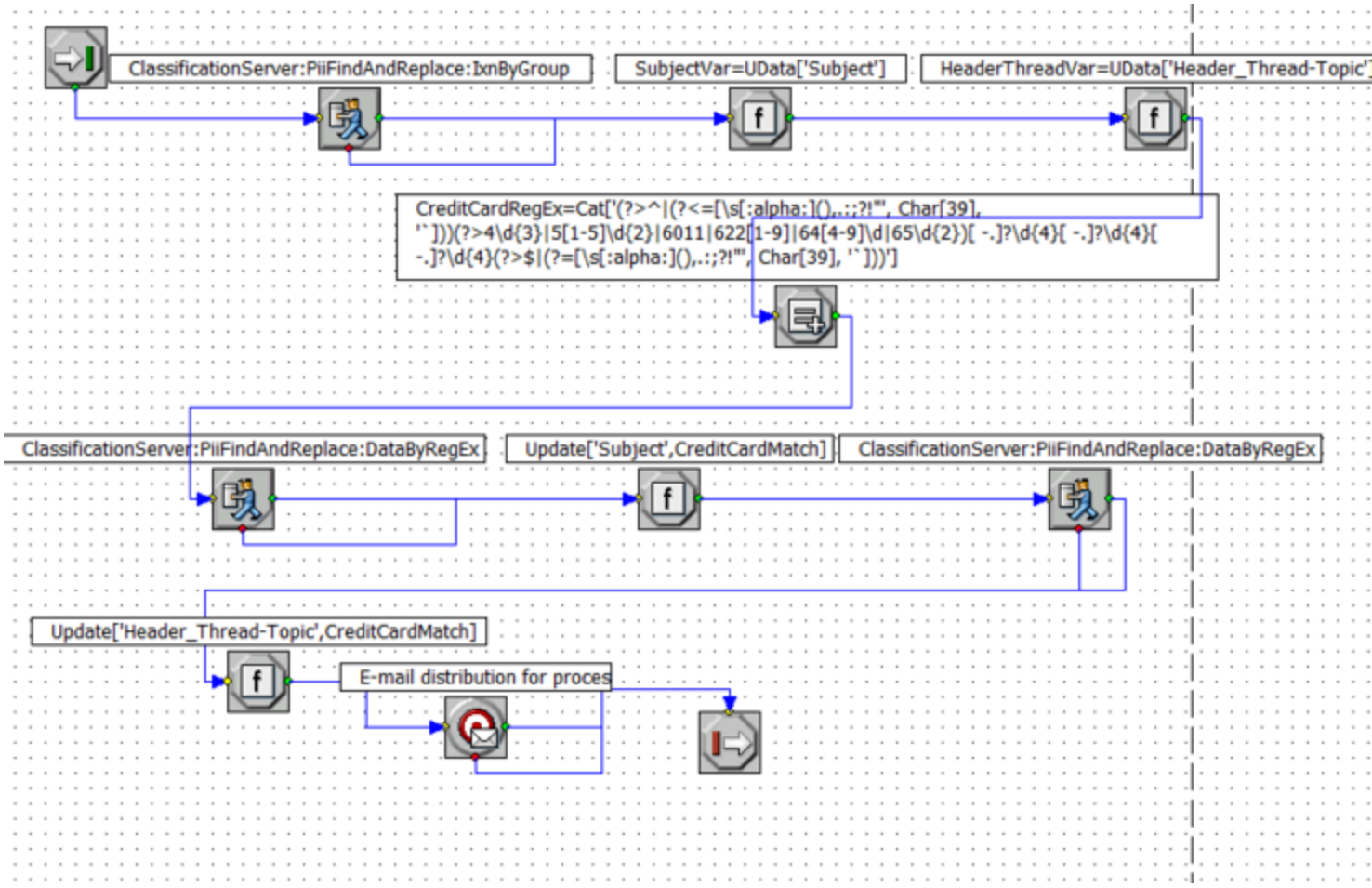
Routing Strategies for Sensitive Data

For channels other than Chat, you must use Composer or Interaction Routing Designer (see the Universal Routing documentation) to create strategies (or modify existing ones) that include an External Service object that calls one of the following methods:

- IxnByGroup—This method specifies an interaction in the UCS database and the group of rules to apply to it. Its parameters are listed below.
- DataByRegex—This method extracts the text to be screened from the interaction as it passes through the strategy and the regular expression to apply to the text. Use it when you do not want to (or cannot) retrieve the interaction from the UCS database. In addition to the External Service object, strategies using this method must include some strategy object that extracts content from the user data and puts it in a variable which it passes to the External Service object. The parameters of this method are listed below.

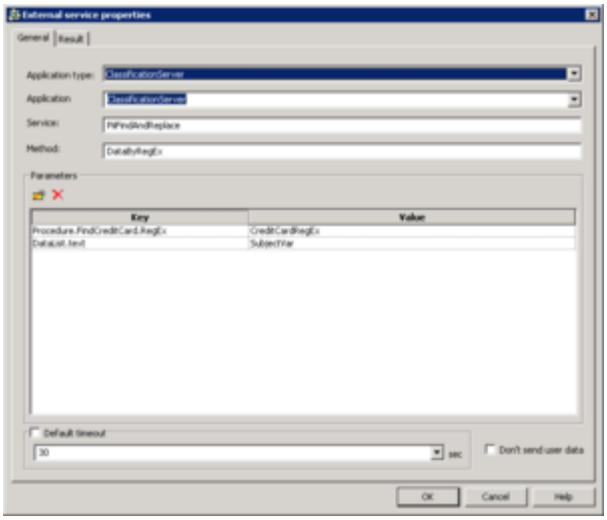
Sample Strategy

The following strategy illustrates the use of both methods on an email interaction:

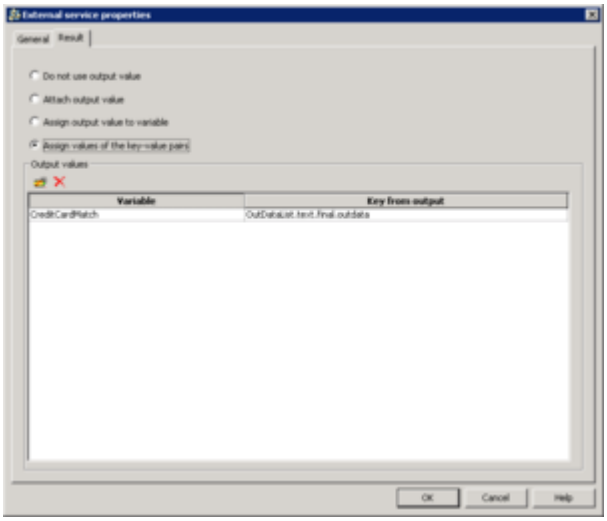


Strategy Using IxnByGroup and DataByRegEx

1. In the first **External Service** object, IxnByGroup looks at the interaction in the UCS database and scans its entire content: Subject, Header, Body. It also updates the content of the interaction as stored in UCS, replacing any sensitive data that it finds with strings of * (asterisks). However, IxnByGroup does not affect the interaction's User Data, which contains attributes, such as Subject and various headers, that might also contain sensitive data. For that we must use DataByRegex.
2. Two Function objects retrieve the content of the Subject and Thread-Topic.
3. A Multi-Assign object creates a variable CreditCardRegEx and assigns it a value consisting of a regular expression that finds credit card numbers.
4. In the second **External Service** object, DataByRegex scans the content of the Subject field.



External Service, General Tab (click to enlarge)



External Service, Result Tab (click to enlarge)

5. The following Function object updates the interaction (in the Interaction Server database), substituting * for the found data.
6. The third **External Service** object does the same for the Thread-Topic field.
7. When the interaction is terminated, the User Data attributes are also updated in the UCS database.

IxnByGroup Parameters

Parameter	Type	Description	Mandatory?	Default Value
Group	String	ID of the rule group to be applied. Either Group or	N	No default value

Parameter	Type	Description	Mandatory?	Default Value
		GroupName may be specified, but not both. If both are specified an Error is generated. If neither is specified, the predefined `Email` group is used.		
GroupName	String	Name of the rule group to be applied. Either Group or GroupName may be specified, but not both. If both are specified an Error is generated. If neither is specified, the predefined `Email` group is used.	N	Email
IxnAccessSpec	List	Specifies which parts of interaction stored in UCS should be processed, and other parameters needed for `Ixn. Ucs Access Provider`. This string is passed to the Provider and is used by the Provider exclusively. The string has the following form: `key:value=<part>:<operation>`, where *part* can be `Subject`, `Header`, `Text` (body), `StructuredText`, `Content` (MIME content), or `_EmailAll` (all fields) *operation* can be `check` (the modified part of the interaction is not written back to UCS) or `update` (the modified part of the interaction is written back to UCS).	N	`key:value=_AllEmail:update`
IxnList	String	List of IDs of interactions stored in UCS, separated	N	No default value

Parameter	Type	Description	Mandatory?	Default Value
		by the pipe character (|). If absent, the Interaction ID is taken from user data.		
ProcedureOpt	String	Sets the output type of the procedure: • `final`—only final processed data is placed in the result • `trace`—full output with results of all intermediate procedure steps, including positions, is placed in the result.	N	`final`

DataByRegex Parameters

Parameter	Type	Description	Mandatory?	Default Value
DataList	List	Specifies the data portions to process: a list of key-value pairs, where the key is the reference ID of this data portion and the value is a string specifying the data portion to process.	Y	No default value
Procedure	List of K-V pairs	Describes find and replace procedure by direct explicit specifications of its steps. • Key (String)—Reference ID of this step of the	Y	No default value

Parameter	Type	Description	Mandatory?	Default Value
		procedure • Value: (List of key-value pairs)—Specification of this step of the procedure, as listed in "Values of **Procedure**" below.		
ProcedureOpt	String	Sets the output type of the procedure: • `final`—only final processed data is placed in the result • `trace`—full output with results of all intermediate procedure steps, including positions, is placed in the result.	N	`final`

Values of **Procedure**

Key	Type	Description	Default Value
TheOrder (optional)	Integer	Specifies the order of this part of the procedure. If **Procedure** contains only one step then TheOrder can be omitted. Otherwise TheOrder must be specified for each step of the procedure, and each step must have a different value.	No default value
RegEx (mandatory)	String	Regular expression used to process the data	No default value
ReplacementPattern (optional)	String or K-V list	Replacement pattern applied in data	See embedded table to left

Key	Type	Description	Default Value
		processing. **Key** / **Type** / **Description** / **Default Value** "name" / String / Name of the replacement pattern / Empty string "type" / String / Type of replacement pattern: • "standard"—POSIX type, with named and numbered capturing groups • "genesys"—as specified in the "spec" attribute / "genesys" "spec" / String / Specification of the replacement pattern if "type" = "genesys": • "none"—Replace nothing • "replace-all"—Replace all characters in the found text • "replace-digits- / "replace-digits-0"	

Key	Type	Description	Default Value
		<table><tr><th>Key</th><th>Type</th><th>Description</th><th>Default Value</th></tr><tr><td></td><td></td><td><N>"—Replace only digits in the found text, leaving the <N> rightmost digits, where <N> is a non-negative integer.</td><td></td></tr><tr><td>"repchar"</td><td>String</td><td>Specifies a character used to replace all characters in the found text</td><td>* (asterisk)</td></tr></table>	

Response

The response to the above methods is Event3rdServerResponse, which has the following parameters:

Key	Type	Description	Default Value
OutDataList	List of lists	Key is the reference ID of the original data portion: the interaction ID or the reference ID of the data portion in the DataList parameter of the request.	No default value

<table>
<tr><th>Key</th><th>Type</th><th>Description</th><th>Default Value</th></tr>
<tr><td></td><td></td><td>The value is as follows
<table>
<tr><th>Key</th><th>Type</th><th>Description</th><th>Default Value</th></tr>
<tr><td>"final"</td><td>List of key-value pairs</td><td>The key is "outdata" and the value is a string consisting of the final result of processing this data portion.</td><td>No default value</td></tr>
<tr><td>Reference ID of a step of the procedure (the key in the request's Procedure key-value list).

These data elements are created only when ProcedureOpt = trace.</td><td>List of key-value pairs</td><td><table>
<tr><th>Key</th><th>Value</th></tr>
<tr><td>"out"</td><td>String: the result of processing this step</td></tr>
<tr><td>"posfound"</td><td>List: positions of text found in this processing step; see description below.</td></tr>
<tr><td>"poschanged"</td><td>List: positions of texts inserted in place of</td></tr>
</table></td><td>No default value</td></tr>
</table></td><td></td></tr>
</table>

Key	Type	Description	Default Value
		<table><tr><th>Key</th><th>Type</th><th>Description</th><th>Default Value</th></tr><tr><td></td><td></td><td><table><tr><th>Key</th><th>Value</th></tr><tr><td></td><td>found text; see description below.</td></tr></table></td><td></td></tr></table>	

Values of "posfound" and "poschanged"

Key	Type	Value
"start"	String	Starting position. The first character in a string is numbered 0.
"end"	String	Ending position.