



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

Privacy Manager Plug-in for GAX

Routing Strategies for Sensitive Data

12/21/2025

Contents

- 1 Routing Strategies for Sensitive Data
 - 1.1 Sample Strategy
 - 1.2 IxnByGroup Parameters
 - 1.3 DataByRegex Parameters
 - 1.4 Response

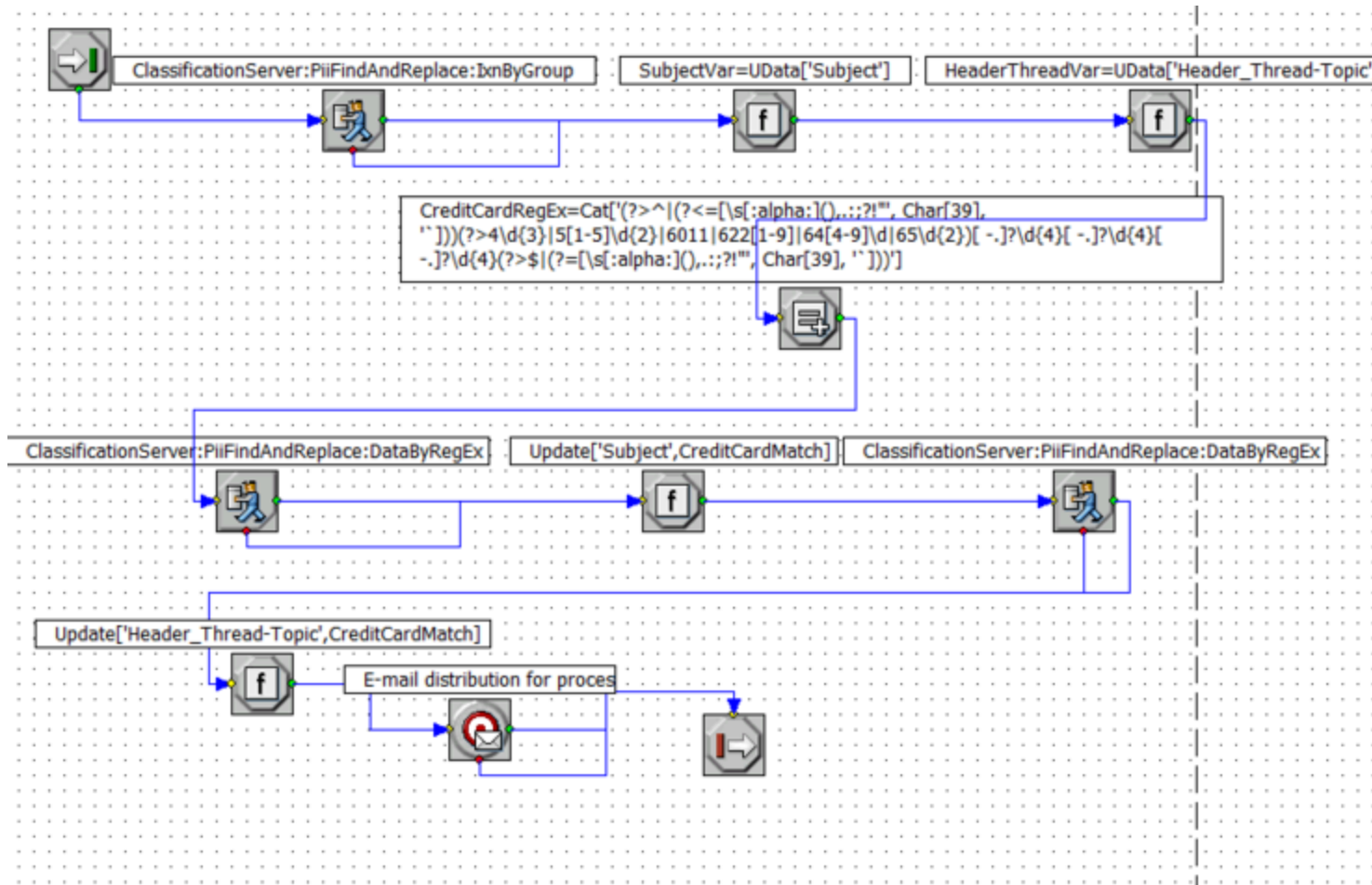
Routing Strategies for Sensitive Data

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**.

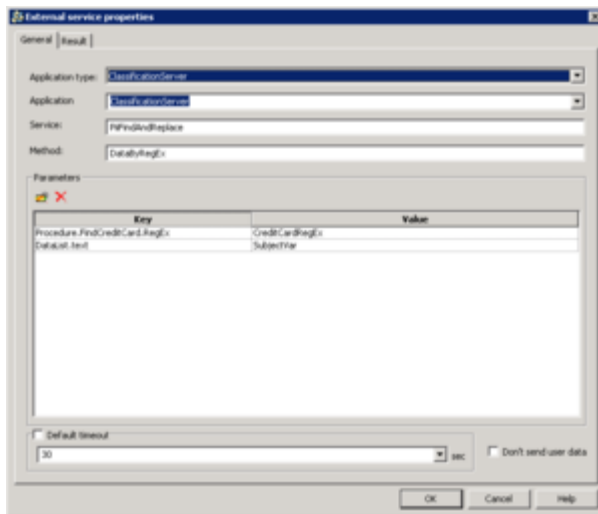
Sample Strategy

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

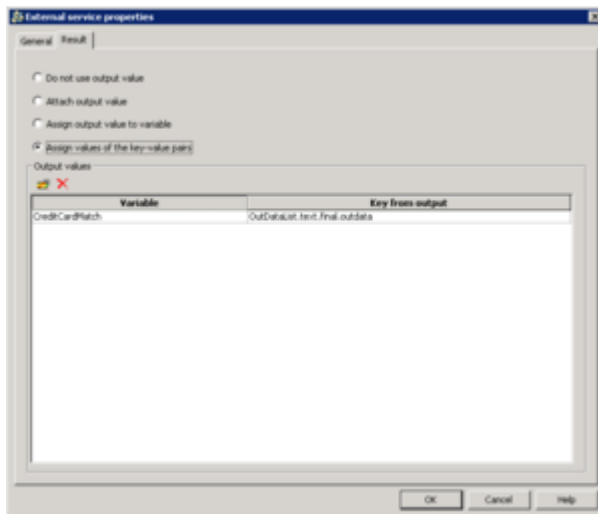


Strategy Using IxnByGroup and DataByRegex

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
		<p>GroupName may be specified, but not both.</p> <p>If both are specified an Error is generated. If neither is specified, the predefined Email group is used.</p>		
GroupName	String	<p>Name of the rule group to be applied. Either Group or GroupName may be specified, but not both.</p> <p>If both are specified an Error is generated. If neither is specified, the predefined Email group is used.</p>	N	Email
IxnAccessSpec	List	<p>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</p> <p><i>part</i> can be Subject, Header, Text (body), StructuredText, Content (MIME content), or _EmailAll (all fields)</p> <p><i>operation</i> 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).</p>	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	<p>Sets the output type of the procedure:</p> <ul style="list-style-type: none"> 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	<p>Describes find and replace procedure by direct explicit specifications of its steps.</p> <ul style="list-style-type: none"> Key (String)—Reference ID of this step of the 	Y	No default value

Parameter	Type	Description	Mandatory?	Default Value
		<p>procedure</p> <ul style="list-style-type: none"> Value: (List of key-value pairs)—Specification of this step of the procedure, as listed in "Values of Procedure" below. 		
ProcedureOpt	String	<p>Sets the output type of the procedure:</p> <ul style="list-style-type: none"> 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.																			
<table><tr><th>Key</th><th>Type</th><th>Description</th><th>Default Value</th></tr><tr><td>"name"</td><td>String</td><td>Name of the replacement pattern</td><td>Empty string</td></tr><tr><td>"type"</td><td>String</td><td>Type of replacement pattern:<ul style="list-style-type: none">"standard"—POSIX type, with named and numbered capturing groups"genesys"—as specified in the "spec" attribute</td><td>"genesys"</td></tr><tr><td>"spec"</td><td>String</td><td>Specification of the replacement pattern if "type" = "genesys":<ul style="list-style-type: none">"none"—Replace nothing"replace-digits-0""replace-all"—Replace all characters in the found text"replace-digits-</td><td></td></tr></table>				Key	Type	Description	Default Value	"name"	String	Name of the replacement pattern	Empty string	"type"	String	Type of replacement pattern: <ul style="list-style-type: none">"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": <ul style="list-style-type: none">"none"—Replace nothing"replace-digits-0""replace-all"—Replace all characters in the found text"replace-digits-	
Key	Type	Description	Default Value																
"name"	String	Name of the replacement pattern	Empty string																
"type"	String	Type of replacement pattern: <ul style="list-style-type: none">"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": <ul style="list-style-type: none">"none"—Replace nothing"replace-digits-0""replace-all"—Replace all characters in the found text"replace-digits-																	

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

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

Key	Type	Description	Default Value
The value is as follows			
		</	

Key	Type	Description				Default Value
		Key	Type	Description	Default Value	
				Key	Value	
				found text; see description below.		

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.