Genesys Security Deployment Guide

Predictive Routing Support for GDPR

5/6/2025

Contents

Predictive Routing Support for GDPR

This page describes product-specific aspects of Predictive Routing support for the European Union's General Data Protection Regulation (GDPR) in premise deployments. For general information about Genesys support for GDPR compliance, see General Data Protection Regulation.

> ## Warning
> Disclaimer: The information contained here is not considered final. This document will be updated with additional technical information.

Genesys Predictive Routing (GPR) provides you the ability to do the following actions, in compliance with GDPR requirements:

- Export Personally Identifiable Information (PII)
- Remove PII ("Forget me")

> ## Important
> GDPR compliance is fully managed by the client. GPR provides the necessary endpoints, but compliance requires you to perform the steps required to locate and remove PII.

- For additional details, refer to the *Predictive Routing API Reference* (access requires a password; contact your Genesys representative for assistance).

Locating and Handling PII via the GPR API

Use the following procedure to read and delete PII:

1. Specify the unique field (Field) that identifies the person making a GDPR request (Person).
2. Find the data structures where the Person's data might be located:
    - Agent Profile schema
    - Customer Profile schema
    - Datasets (there might be multiple datasets)
    - Predictors (there might be multiple predictors)
    - (Optional) Accounts and user-management data might also contain PII. However, note that changes

to these types of data might interfere with GPR operations, because accounts and users are required for administration.

3. Compose an API request directed to each data structure containing PII for the Person.

> ### Important
>
> Refer to Table of API Commands Used to Handle PII in GPR (below) and choose the correct command syntax for each data structure.

4. (Optional) Before removing PII, execute a data export request with the same filter to ensure you are about to remove the right data.

5. Execute the request. In case of removal, the request removes the entire row (document) that matches the filter from the selected data structure.

Example

This example demonstrates how to find and remove information about the customer with the email address johndoe@example.com from the data stored in a predictor.

To start, inspect the data structure to find the relevant field:

```
curl \
    --request GET \
    "https://localhost/api/v2.0/predictors/{id}?token={token}"
```

The result indicates that the email address is stored in a field called customer_email.

Next, export the PII associated with the email address from the specified predictor:

```
curl \
    --request GET \
    "https://localhost/api/v2.0/
predictors/{id}/data?token={token}&filter=%28ctx.customer_email%3Djohndoe%40example.com%29"
```

> ### Important
>
> - Customer fields have the prefix ctx and agent fields have the prefix act. These prefixes exist only in the predictor data structure.
>
> - As shown In the example above, the value of the filter parameter in the query of a GET request (in this example, ctx.customer_email=johndoe@example.com) must be URL (percent) encoded.

To remove the PII from this predictor, use the following command:

```
curl \
    --request POST \
    --header "Content-Type: application/json" \
```

```
--data '{"data_filter": "(ctx.customer_email=johndoe@example.com)"}' \
"https://localhost/api/v2.0/purge/predictors/{id}?token={token}"
```

> **Important**
>
> - PII removal is a permanent (hard) delete operation. You cannot restore deleted data.
>
> - PII removal is an asynchronous job. Check the job status to ensure that the job has removed the PII successfully.
>
> - PII removal deletes the entire row (document) matching the filter. If you filter the data only by the email field, other fields are removed as well.

Table of API Commands Used to Handle PII in GPR

Data	Read	Delete
Agent profiles	GET /agents?ID={id}	POST /purge/agents + body {"data_filter": "({field}={value})"}
Customer profiles	GET /customers?ID={id}	POST /purge/customers + body {"data_filter": "({field}={value})"}
PII in datasets	GET /datasets/{id}/data?filter=({field}={value})	POST /purge/datasets/{id} + body {"data_filter": "({field}={value})"}
PII in predictors	GET /predictors/{id}/data?filter=({field}={value})	POST /purge/predictors/{id} + body {"data_filter": "({field}={value})"}