

GDPR - Delete sensitive contact data

Objective:

This document describes how the deletion process occurs in the platform to be compatible with GDPR.

To be compatible with GDPR, this deletion process was designed using a standard methodology that is recognized to support GDPR "Right to be Forgotten" (article 17), and CCPA "Right to Deletion".

When a contact is going to be deleted?

According to our official "Marketing Automation Data Privacy & Legal" documentation, there are 4 triggers that start the execution of this deletion process for one specific customer:

1. Customer request by a ticket on the Service Desk platform
2. The user requests the unsubscribe by manually entering the special keyword "STOP" on the channel (the time period may change according to the deletion policy defined by the customer).
3. The GDPR background worker detects inactive users on a daily basis according to the deletion policy defined by the customer.
4. Customers do it programmatically by executing an API request (not implemented yet)

What happens during the deletion process for a contact?

Hard deletion without causing data corruption is impossible in our database schema. Also, hard deletion will affect negatively all the analytics & statistics useful for our customers. Based on those premises, we defined to use a "Hard Anonymization" process. "Hard Anonymization" means changing (anonymizing) all personal data elements without touching referential fields (to prevent integrity problems). The kind of anonymization we use, was implemented in such a way that the personal data can no longer be attributed to an identified or identifiable natural person because it is not reversible.

The "Hard Anonymization" or deletion process for a contact is executed on a 3 steps process which is executed inside of a unique transaction in a PostgreSQL Store Procedure (`gdpr_delete_contact_by_contact_id`):

1. A new row is created on the GDPR Audit table specific for contacts deleted (datatable: `gdpr_audit_contacts_deleted`)
2. All the personal data associated with the contact is centralized in a datatable, which is deleted and replaced by an anonymized default values which are not reversibles to the original personal data.
It includes the following fields in the `engine_contact` datatable:
 - a. `wid` it stores the contact's phone number. This phone number is deleted and replaced by the first two/three digits which represents the international prefix for phone numbers* because there are customer statistics which uses the prefix to calculate users per country as an example.
Example: before delete = 49123456789 => after delete = 49
* Some countries uses 2 digits as international prefix while others uses 3 digits.
 - b. `name` it stores the WhatsApp setting defined by the user as the name. Technically, the user can enter any string/emoji /phrase as the content for this field using his WhatsApp App personal.
The name field is replaced automatically by the asterisk character (also known as small starlike symbol), which make impossible to reverse it to get the original name that WhatsApp returns.
Example: before delete = 'Marcelo Duarte' => after delete = ''*
Example: before delete = 'Pedro' => after delete = ''*
Example: before delete = '' => after delete = ''*
 - c. `data` during the opt-in process and all the conversations exchanged between the user and the channel, multiple 'tags' are associated to the contact to make it easy apply different filtering criterias during the communication. Some examples of the 'tags' are: preferred language, postal code, preferred shop/market, etc.
All the extra data ("tags") associated with the contact are deleted automatically, replacing them with only two hardcoded tags: 'deleted = true', 'deleted_at : current_timestamp' for statistical purposes.
3. All the messages sent by the user and all the messages sent by the channel to the user, are also affected during the deletion process, because they may also contain some sensitive data entered by the user.
All the historical messages associated with that user, are automatically updated, removing all the data from two fields ('payload', 'api_response') which are the only fields where the message content is stored in our database.
If there is any attachment sent by the user by a message, it will be deleted from our database.
Example: before delete => after delete = {}

```

{
  "type": "template",
  "template": {
    "name": "template_name_easter23",
    "language": {
      "code": "de",
      "policy": "deterministic"
    },
    "namespace": "48af4b5b_1a7d_4394_a901_ab5063c61ab8",
    "components": [
      {
        "type": "header",
        "parameters": [
          {
            "type": "image",
            "image": {
              "id": "4c79799d-50b2-4f0a-bc68-30f309326d5a"
            }
          }
        ]
      },
      {
        "type": "body",
        "parameters": [
          {
            "text": "Hello Marcelo!",
            "type": "text"
          }
        ]
      }
    ]
  }
}

```

GDPR Audit table definition

This table only is going to have the unique contact_id, and the customer it belongs to (account_id) , but no personal sensitive data is stored here.

This table is going to be useful for verification purposes, especially in cases of data subject requests, knowing that the supervisory authority or any legal entity can request for a time period of up to 3 years that we have complied with the request.

```
CREATE TABLE public.gdpr_audit_contacts_deleted (
    deleted_at timestamp with time zone NOT NULL DEFAULT NOW(),
    contact_id bigserial NOT NULL,
    account_id int8 NOT NULL,
    reason varchar(150) NULL
);
```

Table Name:

gdpr_audit_contacts_deleted

Object ID:

0

Tablespace:

pg_default

Owner:

N/A

Partition by:

Has Row-Level Security Partitions

Comment:

Extra Options:

	Column Name	#	Data type	Identity	Comment	Not Null	Default
Columns	<div><div></div>deleted_at</div>		timestamp with			<div>[v]</div>	NOW() AT TIME ZONE "UTC"
Constraints	<div><div>123</div>contact_id</div>		bigserial			<div>[v]</div>	
Foreign Keys	<div><div>123</div>account_id</div>		int8			<div>[v]</div>	
Indexes	<div><div>abc</div>reason</div>		varchar(150)			<div>[]</div>	
Dependencies							
References							

GDPR Delete Contact DB stored procedure definition

This stored procedure is going to be executed by all the triggers explained on this document. It execute all the steps required to delete all contact personal data from our DB.

Stored procedure name: `gdpr_delete_contact_by_contact_id`

Executed using: `call gdpr_delete_contact_by_contact_id(contact_id, reason)`

i For security/privacy reasons the source code of this stored procedure was not included on this document, but it is available internally at [GDPR Delete Contact DB stored procedure definition](#)