De-identifying Analytics Data with Skyflow

skyflow

Contents

Abstract

Authored by Manish Ahluwalia, Field CTO, Skyflow

Sensitive data often finds its way into analytical data pipelines where it is rarely needed, adding security and compliance risk. However, blindly purging such data from analytical pipelines risks breaking use cases.

Skyflow's data privacy vault and tokenization help you replace sensitive data with placeholders, reducing security and compliance concerns. Skyflow's governance helps you ensure that you can securely support the use cases that need access to sensitive data. Moreover, Skyflow's vault integrations can help your organization achieve zero-touch workflow execution; you can execute workflows on sensitive data without your systems touching it.

Introduction

Skyflow provides a best-of-breed data privacy vault: a secure, isolated, performant, and reliable environment for the governed use of sensitive data. It's the perfect place for PII, PHI, or other data under regulatory restrictions or which might be a target for breaches.

Due to the data privacy vault's enormous flexibility, it can be hard to understand how to put it to use in your architecture. In this paper, we'll take a look at some analytical use cases. You can use Skyflow's vault to de-identify your sensitive data in multiple ways.

Many of these patterns benefit the entire data infrastructure of your company, including the analytic pipelines that feed from them. However, for the purposes of this paper, we'll only consider scenarios where the data entering your analytics pipeline contains sensitive information and the techniques for reducing or eliminating such risk.

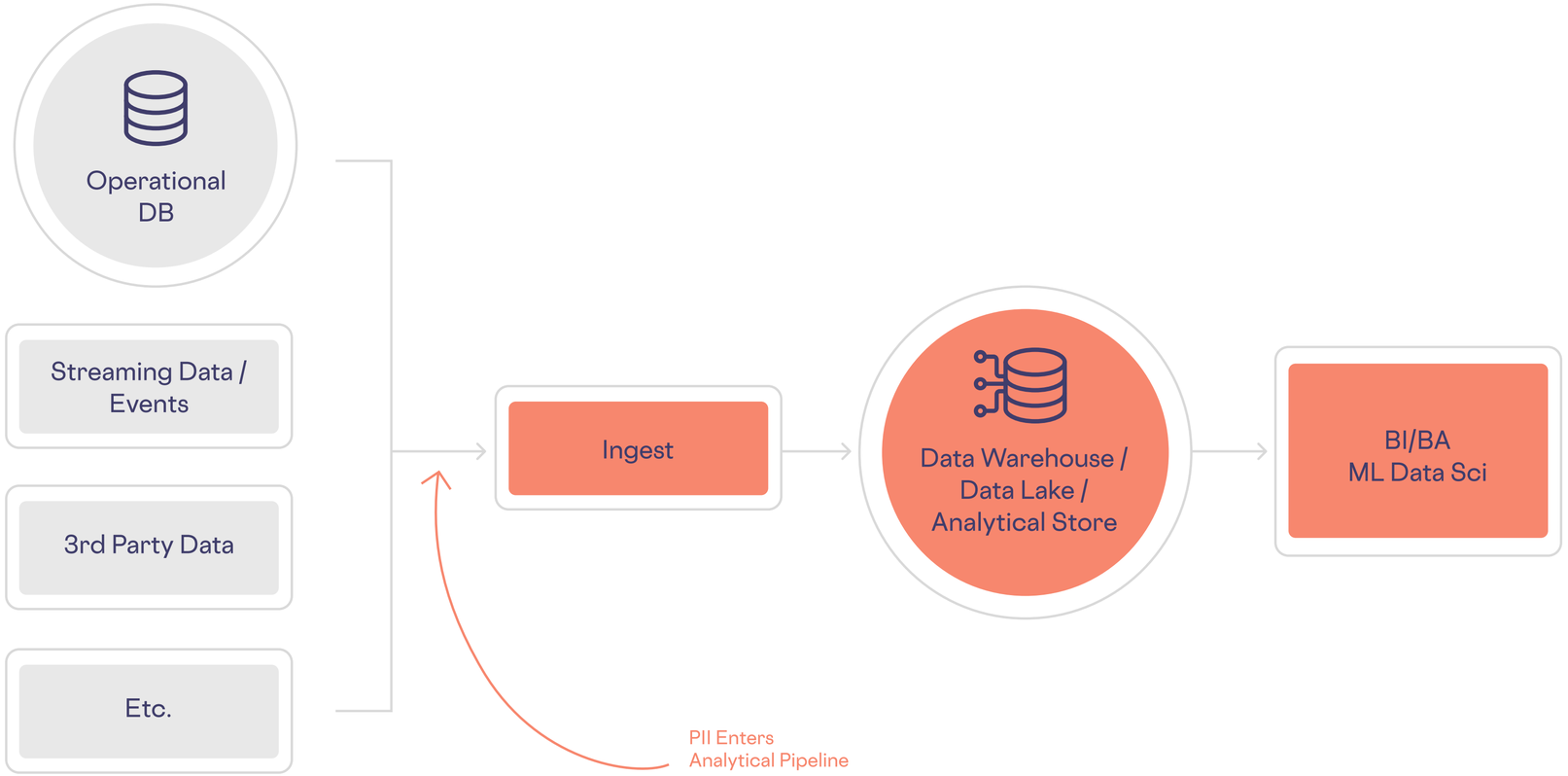
Before we proceed further, here are some key concepts that will help:

- **Data privacy vault:** a solution that stores your sensitive data in a secure and compliant manner, and lets you manage access to it
- **Tokenization:** the process that replaces your sensitive data with "pointers" to the actual data in the vault
- **Governance:** how you control how vaulted data is used and if and how tokens are detokenized.
- **Vault integrations:** the tools that let you 'do stuff' with your vaulted data without exposing it to your own systems, processes, or people.

Use Case 1:

De-identify data before analytics

Data entering your analytics pipelines can contain PII. This adds privacy, security, and compliance risk to your analytics tools and pipelines, for your data warehouse or lake, and for consumers of analytical output.



CONTAINS PII

Solution

You need to de-identify sensitive data before sending it for analytical processing. You also need to ensure that your analytical consumers, particularly those that need PII, continue to function unimpeded.

CONTAINS PII

PII REPLACED BY TOKENS

NEW

ISOLATE & PROTECT

1 Incoming Data which includes PII is intercepted by the “Tokenize” job

1b (Optional) Tokenize process consults data discovery tool to identify PII

2 Tokenize processes pulls out PII and sends to Skyflow

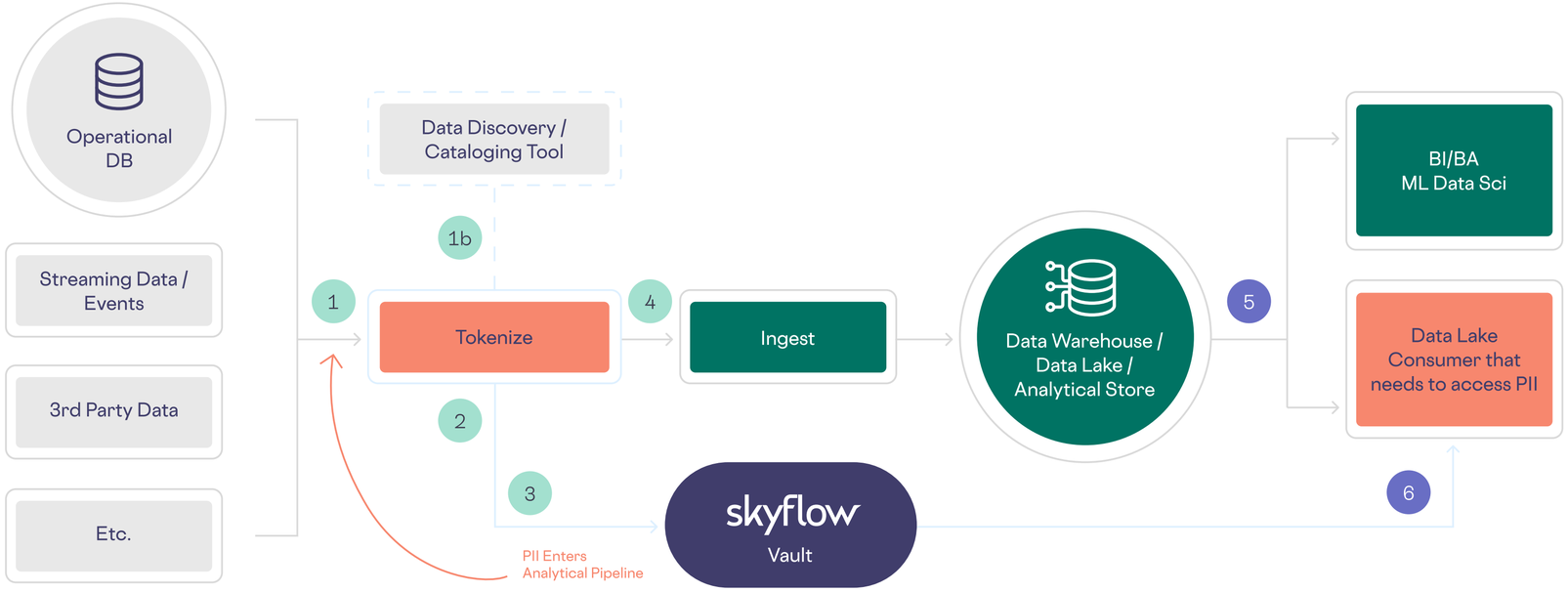
3 Skyflow returns Tokens in exchange for PII

4 Tokenize process replaces PII with tokens and forwards tokenized data

HARNESS

5 Analytical process / systems can store and work with data without risk

6 Authorized Datalake consumers can detokenize the tokens they receive from the Datalake and get policy-gated access to PII



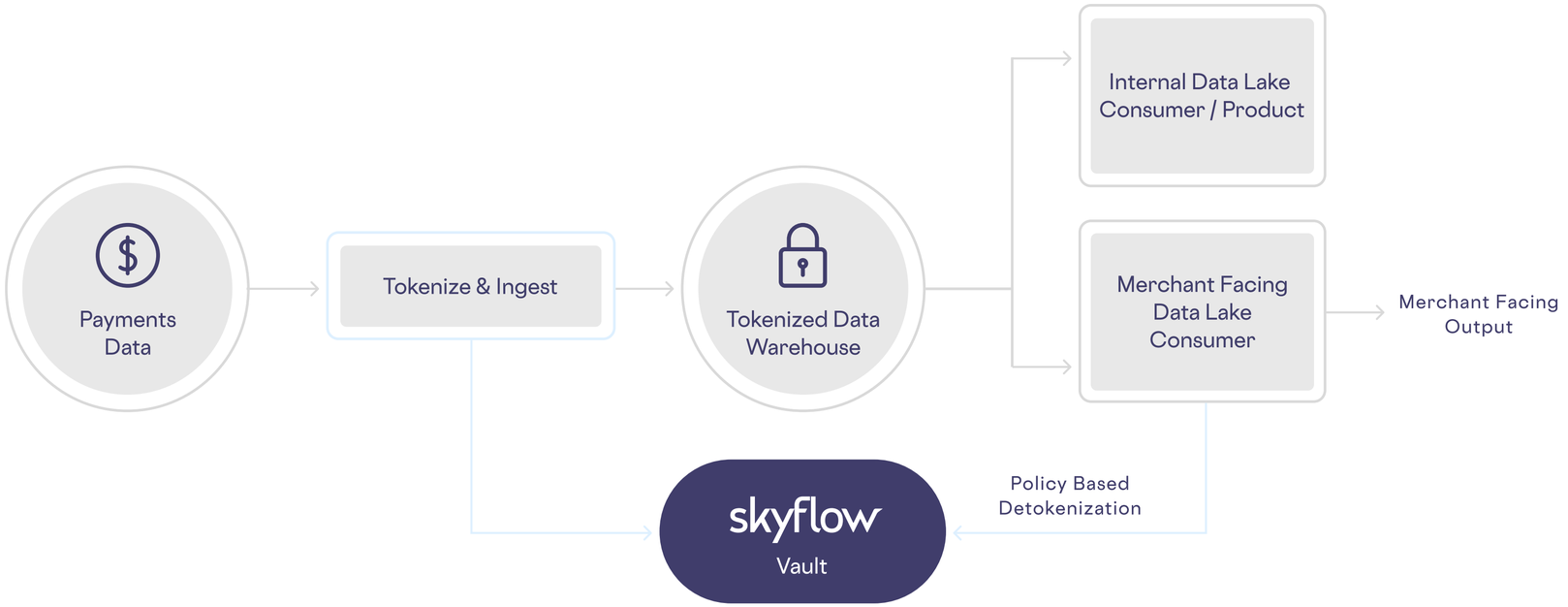
As shown, the only changes to the pipeline are:

- Before data-ingest you insert a “tokenization” process. Alternatively tokenization can be done by the ingest process as part of its operation (not shown)
- The tokenization process can optionally integrate with a data cataloging or discovery tool to find out what part of the incoming data contains PII
- The tokenization process replaces the PII with tokens from Skyflow
- Some analytics consumers that need PII are authorized via Skyflow governance policies to detokenize the PII

Solution

Continued

To make this more concrete, let's take a real-world use case as described in Adyen's blog. In their case, their data lake was storing PII. Most readers from the data lake did not need PII, with the exception of the merchant-facing report builder. They couldn't therefore eliminate PII entirely from their system, but they did not want the risk of storing it in their data lake and with data lake consumers. If we were trying to solve this use case using the above pattern, we would have a architecture that looked like this:



Use Case 2:

Some sensitive data is needed for analytics

The previous example doesn't consider cases where your analytical pipeline needs to process limited PII. For instance, say you need to group customer data by ZIP code before analyzing or you need to use the date of birth information to segment customers by age. You could handle this in the following ways:

OPTION 1 | Extract and transform before load

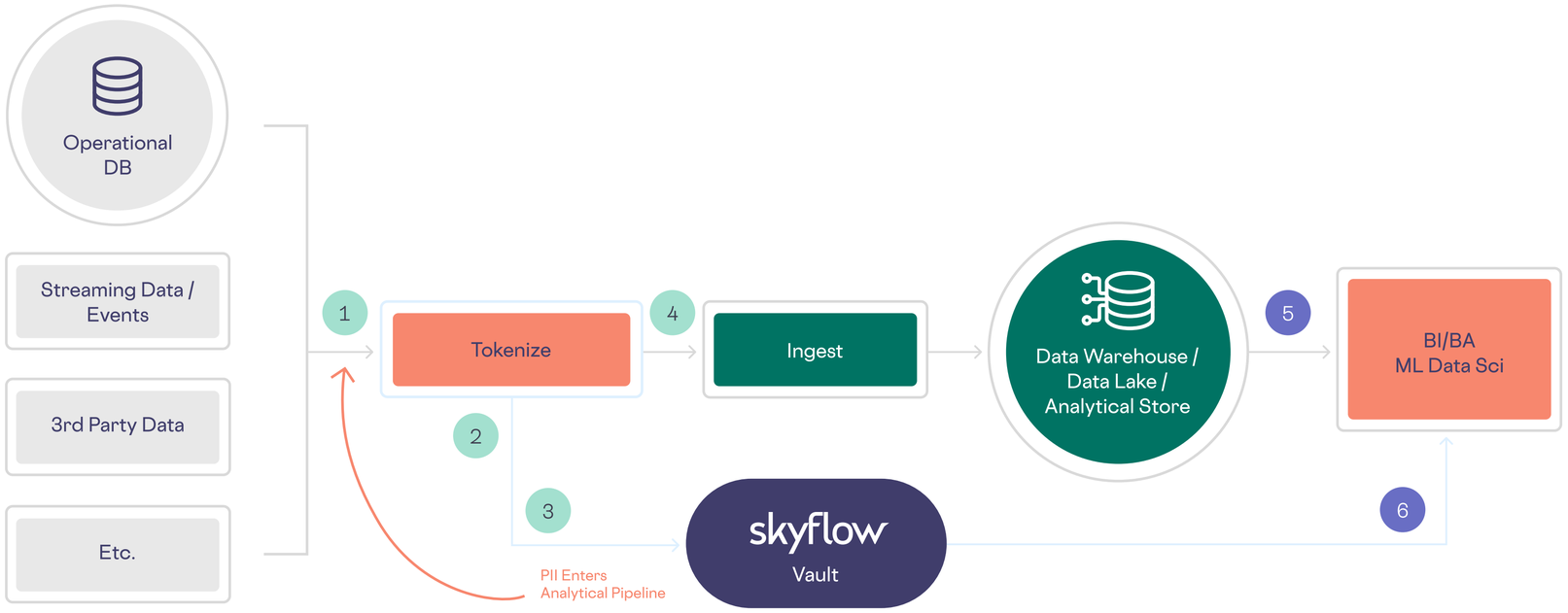
You extract the relevant parts of the PII and separate them from the PII being vaulted. In our example, you would extract the ZIP code (which is not PII) from the address (which is PII). The address would go to the vault to be replaced by a token while the ZIP code would remain available for analytics. Alternatively, you could compute the age from the date of birth during ETL and store the age for processing. The rest of the system would remain as in the previous use case.

This approach requires:

1. **That you know in advance what fields will be needed in the queries that will be run on the data.**
2. **You run a one-time transform on your data each time the query requirements change. To do this, your one-time transform would have access to the full address from the Skyflow vault, from which it would extract the ZIP code and store that in the data lake for then queries to consume.**

Option 2 | Grant analytics selective access to PII

You would make use of some of Skyflow's capabilities for role-based access control (RBAC) and / or privacy-preserving queries.



CONTAINS PII

PII REPLACED BY TOKENS

NEW

ISOLATE & PROTECT

1 Incoming Data which includes PII is intercepted by the "Tokenize" job

2 Tokenize processes pulls out PII and sends to Skyflow

3 Skyflow returns Tokens in exchange for PII

4 Tokenize process replaces PII with tokens and forwards tokenized data

HARNESS

5 Analytical process / systems can store and work with data without risk

6 Select analytical process can perform analytics on the vaulted data; or, can detokenize selected fields

The only addition is step 5b, which can be implemented in one of two ways:

1. Encrypted Analytics: Skyflow can be configured to give the analytics processes access to the vault to perform encrypted analytics on the tokenized data. For instance, the analytical process can perform an in-vault query (i.e. the data never leaves the vault) to group all customers over age 30 by ZIP code.

2. Fine-grained governance: Skyflow can be configured to give the analytics processes access to selectively detokenize the PII it needs for analytics. For instance, the analytics process can only extract the ZIP code part of the address.

Whichever scheme you use, the extracted ZIP codes could be mounted as an external table depending on the warehousing technology you are using.

Use Case 3:

You want to act on the PII in the analytical output

Say you want to send out emails to all customers over the age of 30 living in selected areas. Your analytical systems have performed these computations and have identified the target customers. You are now in possession of the tokenized email addresses to target. As before, you could have your “analytical consumer” detokenize the email address and then send the email, as shown:

- CONTAINS PII
- PII REPLACED BY TOKENS

ISOLATE & PROTECT

1 Incoming Data which includes PII is intercepted by the “Tokenize” job

2 Tokenize processes pulls out PII and sends to Skyflow

3 Skyflow returns Tokens in exchange for PII

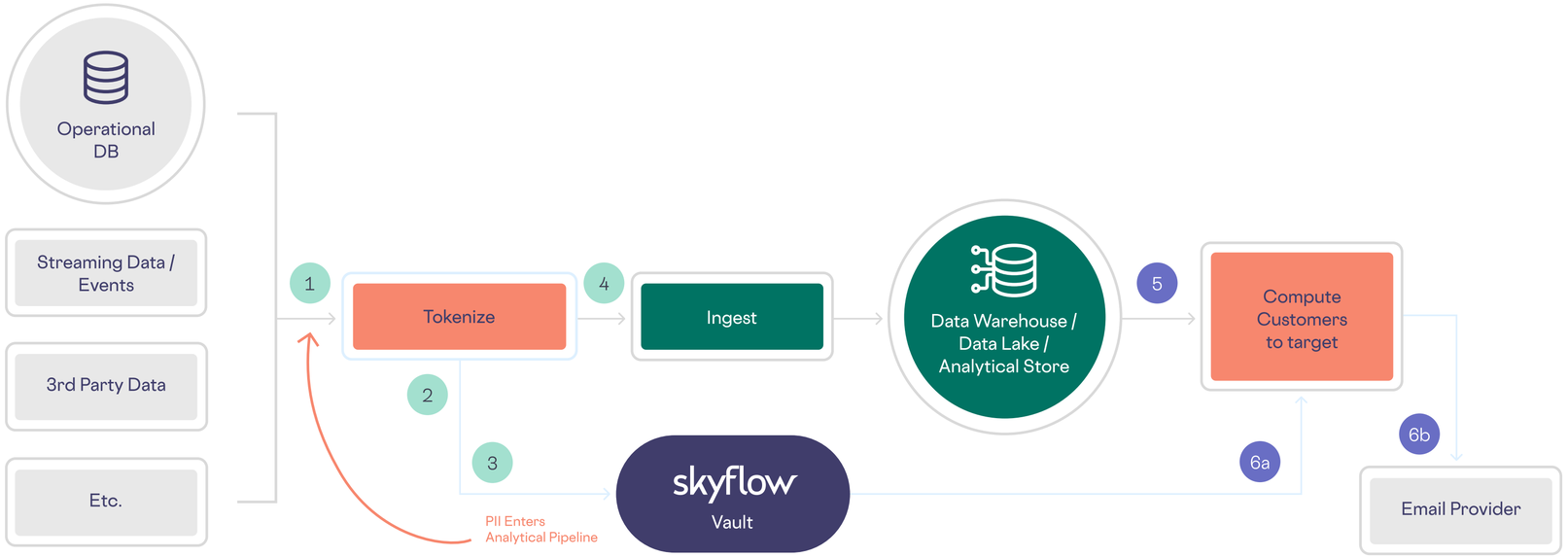
4 Tokenize process replaces PII with tokens and forwards tokenized data

HARNESS

5 Analytical process / systems can store and work with data without risk

6a Analytical process identifies tokenized email addresses to target. Analytics consumer detokenizes the email address via Skyflow, retrieving plain-text email address

6b Analytical consumer uses the plain-text email address to send email



While the privilege to detokenize email addresses can be restricted via Skyflow governance rules, this still exposes the analytics consumer to PII.

Instead, you can use a Skyflow Vault integration (in this case, with email providers). The vault integration works by giving callers the ability to invoke selective integrations (under configurable RBAC rules) that take a token and act directly on the plaintext value without exposing the plaintext value to the caller.

CONTAINS PII

PII REPLACED BY TOKENS

ISOLATE & PROTECT

1 Incoming Data which includes PII is intercepted by the “Tokenize” job

2 Tokenize processes pulls out PII and sends to Skyflow

3 Skyflow returns Tokens in exchange for PII

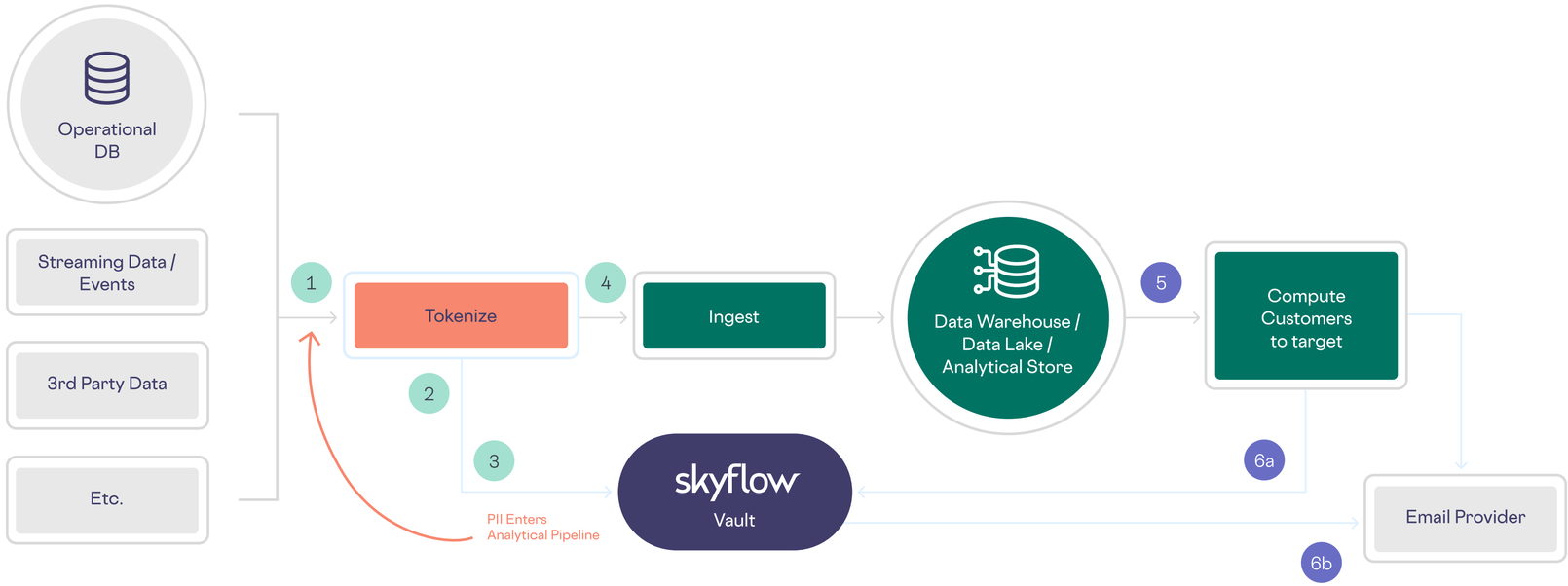
4 Tokenize process replaces PII with tokens and forwards tokenized data

HARNESS

5 Analytical process / systems can store and work with data without risk

6a Analytical process identifies tokenized email addresses to target. Analytics consumer invokes Skyflow vault-integration to send email via email provider

6b Skyflow integration detokenizes the email addresses and sends them to the email provider

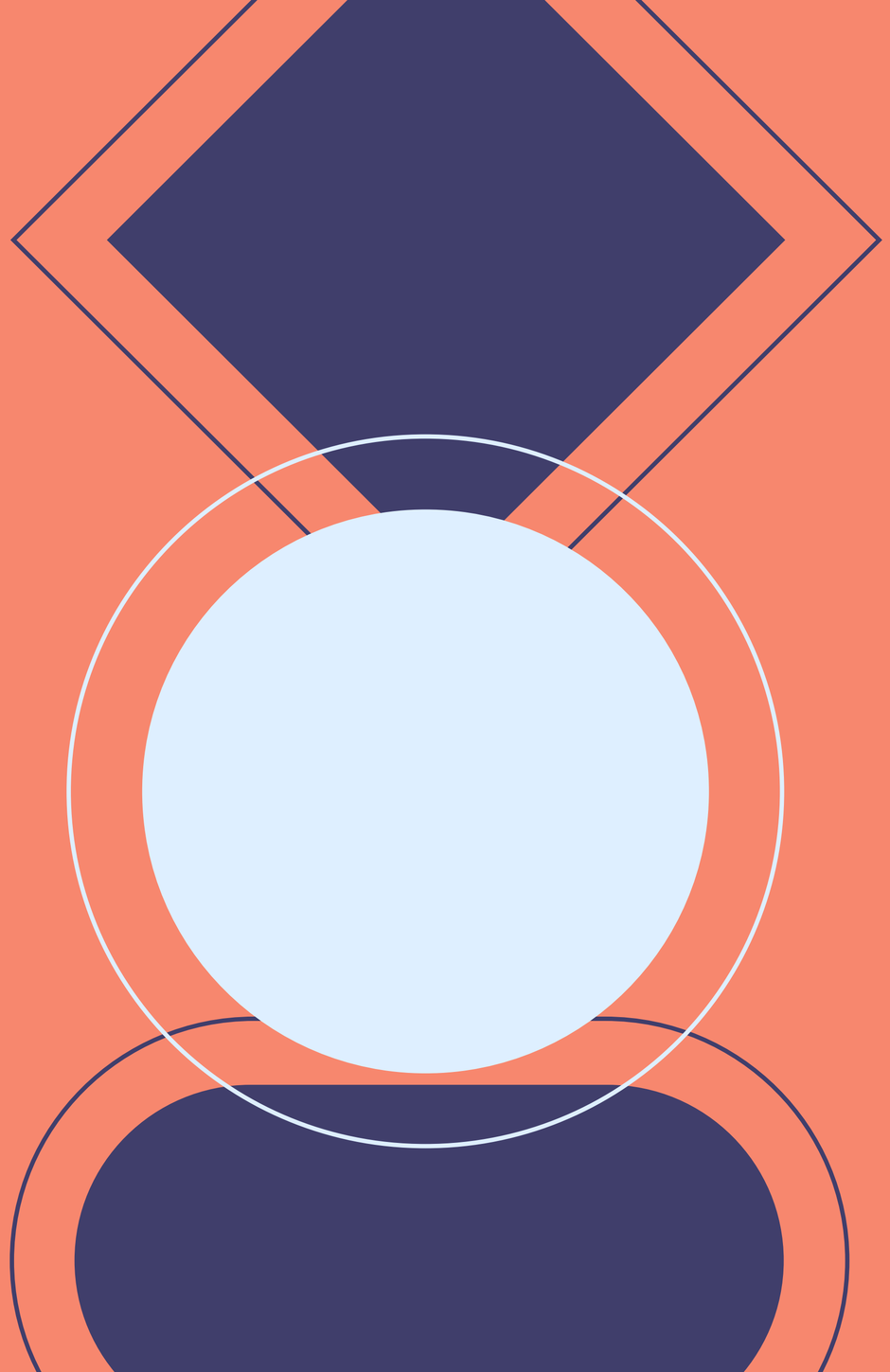


In our example, the “Analytical Biz Ops process” identified users to target with a new email campaign based on desirable business characteristics. Then it invokes a vault integration on the tokenized email addresses. The vault integration detokenizes the email addresses (and other PII needed, like FirstName) and then invokes the “Email Provider” to send the email.

Get in touch

Hopefully this helped understand how Skyflow's data privacy vault can help you remove sensitive information from your analytical pipelines without breaking your use cases. If you're interested in learning more about how we can help you, please sign up for a demo.

Get a Demo



About Skyflow

Founded in 2019, Skyflow is a data privacy vault for sensitive data. The company was founded by former Salesforce executives Anshu Sharma and Prakash Khot to radically transform how businesses handle users' financial, healthcare, and other personal data that powers the digital economy. Skyflow is based in Palo Alto, California, with offices in Bangalore, India. For more information, visit skyflow.com or follow on Twitter and LinkedIn.

About the Author

Manish Ahluwalia has over 2 decades of experience in the software industry, with over 10 years in information-security. Most recently he was running security for NerdWallet. He currently works to help Skyflow's customer's find the right architecture for their data protection needs.

skyflow