Fundamentals of System Security
(COMSM0122)
Lab 2
(25th October, 2022)

Data Inventory

Objective

In this lab, you will learn how to operationalise data governance by transforming data and applying decisions based on data protection risk. You will crawl a datastore to understand the overall risk to the system, introduce a new data pipeline by creating different datastores, and make decisions about access control, sharing and retention. You will focus on understanding the customer data that PiC stores and how the sensitivity of that data drives critical decisions. Start by reviewing the `customer.json` currently stored by PiC.

Task 1:

To operationalise a tagging strategy write a javascript function that automatically assigns a tag to each field type in `customer.json`. Use data sensitivity labels shown in Table 1 and example tags based on sensitivity levels in Table 2 as a guide. Save the new json object as `taggedcustomer.json`

Table 1: Data Sensitivity Labels

Level	**Tag**	**Description**
Highly Sensitive Data	hsi	Data that can personally identify an individual and is highly sensitive if disclosed, such as government IDs, health data and financial data.
Sensitive Data	si	Data that can personally identify an individual
Quasi Sensitive Data	qsi	Data that can identify an individual when combined with other data
Low Sensitive Data	lsi	Data that is relatively harmless

Table 2: Field tags based onsensitivity levels

Field	**Tag**
name	cust:si-name
chino	cust:hsi-medical-chino
medication	cust:hsi-medical-medication
county	cust:qsi-address
city	cust:qsi-address
ethnicity	cust:qsi-ethnicity
volume	cust:lsi
birthdate	cust:qsi-birthdate
creditcard	cust:hsi-financial-creditcard
sex	cust:qsi-sex
bloodgroup	cust:hsi-medical-bloodgroup
email	cust:si-address
timestamp	cust:lsi

Task 2:

Write a javascript function providing an overview of `taggedcustomer.json`- for each field, showing the associated tag and the count. The result should tell you how many instances of each tag type exist in the datastore.

Task 3:

Write a javascript function to determine the sensitivity profile of taggedcustomer.json. The script should tell you the percentage of records that are in each sensitivity class as well as highly sensitive medical or financial data.
Based on your answer to Task 3

1. What proportion of customer.json is highly sensitive.
2. What proportion of customer.json is sensitive.
3. What proportion of customer.json is quasi sensitive.
4. What proportion of customer.json is highly sensitive financial data.
5. What proportion of customer.json is highly sensitive medical data.
6. Is the customer database appropriate for storage in both operational and analysis datastore.

7. Based on this output, which of the following is true about `customers.json` as it exists now?

   (a) Strict access controls should be applied to this database

   (b) This data should be retained for a very limited period of time, and then ideally deleted

   (c) This data can be shared freely internally and externally, with third-party vendors

   (d) A and B only

   (e) B and C only

Task 4

You will introduce a new data pipeline and building a skeleton data warehouse that enables varying access control and retention policies. Write a script that will generate three new datastores from taggedcustomer.json as follows:

1. A datastore that only contains medical records.

2. A datastore that only contains medical and address records.

3. A datastore that only contains medical, address and demographic records.

Discuss the data protection risk associated with this new data pipline. Including how such risk can be mirigated.

Deliverables

This is a formative assessment. Ensure one of the TAs or the course lecturer has reviewed your solutions before the end of tutorial session.