EVL Anonymization

EVL Anonymization Microservice enables fast, automated and cost-effective anonymization of data sets. It can be used for pseudonymization and anonymization of the production data according to GDPR requirements as well as for the protection of commercially sensitive data from developers, testers and other outside contractors.

EVL Microservices are built on top of the core EVL software and retain its flexibility, robustness, high productivity, and ability to read data from various sources; including JSON and Excel files, databases—Oracle, Teradata, PostgeSQL, etc.—and streaming data like Kafka.

- Custom functions can be easily designed and embedded into the solution
- High productivity due metadata driven approach

Configuration File

Based on a configuration CSV file, anonymization jobs are generated together with a workflow to manage many sources easily.

Following table shows an example of such configuration file, say `crm.csv`, which would anonymize an Oracle table `accounts` and a file `cust.csv`:

Src	Entity	Field	Data type	Null	Anon type	EVL Value	Description
ORA	accounts	id	int	No	ANON_UNIQ		Unique ID
ORA	accounts	cust_id	int	No	ANON_LOOKUP		Cust. from lookup
ORA	accounts	iban	string		ANON_IBAN		Keep IBAN valid
ORA	accounts	currency	string				Leave as is
ORA	accounts	score	decimal(8,2)		ANON_AMOUNT(0.1)		±10 %
ORA	accounts	valid_from	date		ANON_VAR		Anon. by variance
ORA	accounts	valid_to	date			anonymize(IN, *out->valid_from+1, *out->valid_to+3650)	Must be greater than valid_from
FILE	cust.csv	id	int	No	ANON_UNIQ		Unique ID
FILE	cust.csv	email	string		ANON_EMAIL		
FILE	cust.csv	person_id	string	No		anon_rc(IN)	Sum = 0 mod 11

Where credentials, connection strings, paths, etc., are set in a separate configuration file shared for one such configuration file.

Anon type – this field contains either EVL predefined or custom defined aliases to any EVL functions.

EVL Value – for specific needs, like dependency on other fields (e.g. anonymized `valid_to` must be always greater than `valid_from`), any EVL code can be used. In very specific cases, like Czech and Slovak Personal ID number, which needs to fulfill divisibility by 11, custom C++ function can be used as well.

Then either from EVL Manager (graphical web interface) or by running in the shell:

```
$ evl anon build configs/crm.csv
$ evl run workflow anon/crm.ewf
```

will generate two jobs (one for `accounts` table and one for `cust.csv` file) and a workflow with these two jobs and run the workflow to anonymize the data.

Anonymization Types

Here is the selection of several Anonymization types, i.e. functions to be used to anonymize, randomize or mask the data.

Anon type	Data type	Description	Example
ANON	any	generic anonymization, with min/max range	"A Sample Text" ⇒ "utTfu9h6saPow" 1982-09-28 ⇒ 2007-05-17
ANON_VAR	date/time	within a ± interval	1982-09-28 ⇒ 1983-08-01
ANON_UNIQ	integers	keep the uniqueness	45582 ⇒ 6484
ANON_NAME	string	retain spaces, capitals and numbers	"A Sample Text" ⇒ "E Pottzs Nwxi" "10 Downing St." ⇒ "85 Pottzsq Na."
ANON_EMAIL	string	anonymize emails	"team@evltool.com" ⇒ "ds0@sFux.3t"
ANON_IBAN	string	keep IBAN validity	"NL91 ABNA 0417 1643 00" ⇒ "FR14 2004 1010 0505 0001 3M02 606"
ANON_IBAN _KEEP_COUNTRY	string	keep country and IBAN validity	"NL91 ABNA 0417 1643 00" ⇒ "NL02 BINK 0123 4567 89"
ANON_IBAN _KEEP_BANK	string	keep bank, country and IBAN validity	"NL91 ABNA 0417 1643 00" ⇒ "NL02 ABNA 0123 4567 89"
ANON_AMOUNT(0.1)	numbers	keep the range ±10 %	20.58 ⇒ 21.03
MASK_LEFT(4) MASK_RIGHT(4)	string	mask by '*' from left mask by '*' from right	"1234 5678 9012" ⇒ "**** **** 9012" "1234 5678 9012" ⇒ "1234 **** ****"
RANDOM	any	generic randomization, keep min/max range	"A Sample Text" ⇒ "uisC7dsSacs" 1982-09-28 ⇒ 2001-12-14
ANON_LOOKUP	string	shuffle the data	"Richard" ⇒ "Donald"
ANON_LOOKUP(file)	string	shuffle from file	"Richard" ⇒ "Donald"

All **ANON types** produce for given value and given salt the same output, but might happen that two different values obtain the same anonymized value.

Only **ANON_UNIQ type** produces the output in a unique way, so bijection is guaranteed. Particularly useful for IDs.

Compared to ANON types, **RANDOM types** might return different value each time.

For detailed information see https://docs.evltool.com/evl-anonymization.

Case Study

One bank needed to provide production data for the development team so the data couldn't be re-identified by keeping the entity relationships. The source were 100+ tables stored in csv files, SQL Server, Informix and Oracle. The target for the anonymized development data was Oracle database. Customer filled-in one configuration file containing all data definitions and anonymization types and parameters leading to the source files (directories for csv files and connect strings to databases). The EVL anonymization jobs were created automatically and run in parallel batches with great performance, e.g. the anonymization of one file containing 10 million rows took 50 seconds.